



THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par *l'Institut National Polytechnique de Toulouse*
Discipline ou spécialité : *Systèmes Informatiques*

Présentée et soutenue par *XIAO Jing*
Le 4 décembre 2009

Titre : *Gestion des incertitudes dans le processus de développement de systèmes complexes*

JURY

M. CAILLAUD Emmanuel LGECO, Professeur à l'Université de Strasbourg
M. MERLO Christophe, Maître de Conférences à l'ESTIA
M. PINEL Pierre, Maître de Conférences à l'INSA-Toulouse
M. ZOLGHADRI Marc IMS, Maître de Conférences à l'Université de Bordeaux
Mme. BARON Claude LATTIS, Professeur à l'INSA-Toulouse
M. GENESTE Laurent LGP, Professeur à l'ENI-Tarbes

Ecole doctorale : *Ecole doctorale Système*
Unité de recherche : *Laboratoire Toulousain de Technologie et d'Ingénierie des Systèmes*
Directeur(s) de Thèse : *Mme. BARON Claude et M. GENESTE Laurent*
Rapporteurs : *M. CAILLAUD Emmanuel et M. MERLO Christophe*

INSTITUT NATIONAL POLYTECHNIQUE DE TOULOUSE
LABORATOIRE TOULOUSAIN DE TECHNOLOGIE ET
D'INGÉNIERIE DES SYSTÈMES

THÈSE

présentée en première version en vue d'obtenir le grade de Docteur,
spécialité « Informatique Industrielle »

par

XIAO Jing

GESTION DES INCERTITUDES DANS LE PROCESSUS DE DÉVELOPPEMENT DE SYSTÈMES COMPLEXES

Thèse soutenue le 4 décembre 2009 devant le jury composé de :

M.	CAILLAUD EMMANUEL	LGECO, Université de Strasbourg	(Rapporteur)
M.	MERLO CHRISTOPHE	LIPSI, ESTIA	(Rapporteur)
M.	PINEL PIERRE	LATTIS, INSA de Toulouse	(Examineur)
M.	ZOLGHADRI MARC	IMS, Université de Bordeaux	(Examineur)
Mme.	BARON CLAUDE	LATTIS, INSA de Toulouse	(Directrice)
M.	GENESTE LAURENT	LGP, ENI de Tarbes	(Co-directeur)

À mes proches, mes amis et mes professeurs. . .

REMERCIEMENTS

Résumé La gestion des incertitudes constitue un sujet particulièrement important pour le processus de développement des systèmes (DS) complexes. Il faut donc améliorer la flexibilité et la prédictibilité des processus de DS en proposant une méthodologie pour appréhender et maîtriser les incertitudes qu'ils comportent, ce qui passe par la modélisation. Le travail présenté dans ce mémoire se concentre sur deux questions indispensables dans ce but : (1) Comment gérer les incertitudes pendant le processus de DS ? (2) Comment représenter les incertitudes dans la modélisation de processus de DS ? Pour répondre à ces questions, la thèse propose une méthodologie incluant l'identification des incertitudes, l'analyse de leurs effets, leur réduction et/ou leur exploitation, et leur surveillance et leur contrôle ; elle définit également une démarche de modélisation des incertitudes avec le langage UML en proposant un méta-modèle des incertitudes et un nouveau profil UML "Uncertainty-of" qui permet de rendre ce méta-modèle opérable avec les outils UML. La validité de la méthodologie et du profil proposés a été testée par l'étude d'un cas industriel. Cette étude a montré que la définition d'une telle méthodologie était utile pour l'organisation de la gestion des incertitudes dans la conception et le développement de systèmes, et que la création d'un nouveau profil UML apportait une solution pertinente dans la représentation des incertitudes.

Mots-clés Processus de développement de systèmes, modélisation des incertitudes, méthodologie de la gestion des incertitudes, méta-modèle des incertitudes, nouveau profil UML "Uncertainty-of"

Abstract Uncertainty management is a particularly important issue for complex systems development (SD) processes. The overall objective to be attained in this context is to improve the flexibility and predictability of SD process by proposing a methodology to understand and master the uncertainties. We believe that modeling uncertainties is an indispensable step in this direction. For this purpose, the presented work in this thesis focuses on two critical issues : (1) How should we manage uncertainties in SD processes? (2) How can we represent uncertainties in modeling such processes? In order to answer to these questions, this thesis proposes a methodology including identification, analysis of impact, diminution and / or exploitation, and monitoring and control of uncertainties ; and it also defines an approach to modeling uncertainties with UML by proposing a meta-model of uncertainties and a new UML profile named "Uncertainty-of", which allows to make this meta-model operable with UML tools. The validity of our proposals was tested by an industrial case study, which showed that the definition of such a methodology is helpful to manage the uncertainties in systems design and development, and that the creation of a new UML profile provides a valid solution in the representation of uncertainties.

Keywords System development processes, uncertainties modeling, methodology of uncertainty management, meta-model of uncertainties, new UML profile "Uncertainty-of"

TABLE DES MATIÈRES

TABLE DES MATIÈRES	ix
LISTE DES FIGURES	xiii
LISTE DES TABLEAUX	xv
1 INTRODUCTION	1
1.1 MOTIVATIONS	1
1.2 CADRE GÉNÉRAL DE NOS RECHERCHES	3
1.3 LES QUESTIONS À RÉSOUDRE	4
1.3.1 Méta-modélisation des incertitudes	4
1.3.2 Gestion des incertitudes au plus tôt dans le processus de développement de systèmes	5
1.4 PLAN DU MÉMOIRE	7
2 CADRE GÉNÉRAL ET PROBLÉMATIQUE	9
2.1 INGÉNIERIE DES SYSTÈMES COMPLEXES ET INCERTITUDES AS- SOCIÉES	10
2.1.1 Qu'est-ce qu'un système?	10
2.1.2 Ingénierie des Systèmes : définition et objectifs	12
2.1.3 Les projets en tant que systèmes	13
2.1.4 Les processus dans le développement de systèmes	14
2.1.5 Incertitudes dans le développement de systèmes complexes	17
2.1.5.1 Incertitudes et imprécisions	17
2.1.5.2 Sources des incertitudes	18
2.1.5.3 Prévisibilité des incertitudes	20
2.2 MODÉLISATION DES INCERTITUDES DANS LA MODÉLISATION DE PROCESSUS DE DS	21
2.2.1 Modèles de processus et leurs éléments fondamentaux	22

2.2.2	Modélisation des incertitudes pour l'analyse des caractéristiques des processus	25
2.2.2.1	PERT / Critical Path Method / Gantt / Design Structure Matrix	25
2.2.2.2	Limites et conclusions	28
2.2.3	Modélisation des incertitudes par langages formels	29
2.2.3.1	Le langage UML	30
2.2.3.2	Le langage SPEM	33
2.2.3.3	Limites et conclusions	34
2.3	STRATÉGIES DE TRAITEMENT DES INCERTITUDES ET DE LEURS EFFETS	35
2.3.1	Risques, incertitudes et effets des incertitudes	35
2.3.2	Principe général de traitement des incertitudes et de leurs effets	36
2.3.3	Analyse des risques	40
2.3.3.1	Outils et techniques pour l'analyse qualitative des risques	41
2.3.3.2	Outils et techniques pour l'analyse quantitative des risques	44
2.3.3.3	Limites et conclusions	47
2.3.4	Gestion des inconnues inconnues	48
2.3.4.1	Principes de l'approche par apprentissage	48
2.3.4.2	Principes de l'approche par sélection	51
2.3.4.3	Conclusions	53
2.4	NOTRE PROBLÉMATIQUE	54
2.5	CONCLUSIONS DU CHAPITRE	57
3	MÉTHODOLOGIE DE GESTION ET MÉTA-MODÉLISATION DES INCERTITUDES	59
3.1	MÉTHODOLOGIE DE GESTION DES INCERTITUDES	60
3.1.1	Vue d'ensemble de la méthodologie	60
3.1.2	Etape 1 : identification et sélection des incertitudes	62
3.1.2.1	Objectifs	62
3.1.2.2	Entrées / Activités recommandées / Outils / Sorties	63
3.1.3	Etape 2 : analyse des effets des incertitudes	66

3.1.3.1	Objectifs	66
3.1.3.2	Entrées / Activités recommandées / Outils / Sorties	66
3.1.4	Etape 3 : réduction et exploitation des incertitudes	67
3.1.4.1	Objectifs	67
3.1.4.2	Entrées / Activités recommandées / Outils / Sorties	68
3.1.5	Etape 4 : validation des solutions et contrôle des incertitudes	70
3.1.5.1	Objectifs	70
3.1.5.2	Entrées / Activités recommandées / Outils / Sorties	70
3.1.6	Activité auxiliaire : représentation des incertitudes	72
3.1.6.1	Objectifs	72
3.1.6.2	Entrées / Activités recommandées / Outils / Sorties	72
3.1.7	Conclusion	73
3.2	INTÉGRATION DES TÂCHES DE LA MÉTHODOLOGIE DANS UN PROJET	74
3.2.1	Phasage d'un projet	74
3.2.2	Intégration des activités de la méthodologie dans un projet	75
3.3	CATÉGORISATION DES INCERTITUDES DANS LE DOMAINE DES PROCESSUS DE DÉVELOPPEMENT	79
3.4	MÉTA-MODÉLISATION DES INCERTITUDES	80
3.4.1	Choix du langage de modélisation	81
3.4.2	Workflow de la modélisation	82
3.4.3	Méta-modèle de représentation des incertitudes "Uncertainty-of"	84
3.4.3.1	Paquetage des incertitudes	84
3.4.3.2	Paquetage des effets des incertitudes	88
3.4.3.3	Paquetage des traitements des incertitudes	91
3.4.3.4	Conclusion du méta-modèle "Uncertainty-of"	93
3.4.4	Définition technique du profil "Uncertainty-of"	94
3.4.4.1	Stéréotypes, valeurs marquées, contraintes structurelles	95
3.5	CONCLUSION DU CHAPITRE	97

4	CAS D'ÉTUDE	101
4.1	CONTEXTE	101
4.2	TRAITEMENT DES INCERTITUDES SUR L'ARCHITECTURE DU SYSTÈME	103
4.2.1	Introduction : l'architecture du système	104
4.2.2	Identification et sélection des incertitudes	105
4.2.3	Analyse des effets des incertitudes	107
4.2.4	Réduction / exploitation des incertitudes	108
4.2.4.1	Identification des solutions envisageables	108
4.2.4.2	Identification des critères de sélection de solutions	110
4.2.4.3	Détermination du choix de solution	111
4.2.5	Représentation des incertitudes	114
4.3	TRAITEMENT DES INCERTITUDES SUR LA DURÉE ET LE COÛT DES ACTIVITÉS LIÉES AU PROCESSUS DE DÉFINITION DU SYSTÈME	117
4.3.1	Introduction : les activités liées au processus de définition préliminaire du système	118
4.3.2	Analyse des effets des incertitudes	119
4.4	CONCLUSIONS DU CHAPITRE	121
5	CONCLUSIONS ET PERSPECTIVES	125
5.1	RÉSUMÉ SUR LES QUESTIONS DE RECHERCHE ET NOS CONTRIBUTIONS	125
5.1.1	Gestion des incertitudes dans le processus de développement de systèmes complexes	125
5.1.2	Méta-modélisation des incertitudes	129
5.1.3	Résumé de nos contributions du point de vue des processus de développement de systèmes complexes	131
5.2	PROPOSITIONS DE DIRECTIONS DE RECHERCHES FUTURES	132
	BIBLIOGRAPHIE	135
	NOTATIONS	143

LISTE DES FIGURES

1.1	La procédure de la gestion des incertitudes	5
2.1	Modèle ZOPH (Fricke et al. 2000)	15
2.2	Exemple du diagramme PERT (Marcel 2002)	26
2.3	Configurations de base des diagrammes de DSM	27
2.4	Exemple de DSM (Qi 2000)	28
2.5	Exemple de diagramme de séquence	31
2.6	Méta-modèle UML : Mécanisme d'extension	32
2.7	Modèle conceptuel de SPEM	34
2.8	Cadre de traitement des incertitudes et de leurs effets dans les systèmes complexes (McManus et Hastings 2005)	37
2.9	L'évaluation des impacts d'un risque (PMI 2000)	42
2.10	Matrice de Vraisemblance-Impacts (NASA 2007)	43
2.11	Estimation de coûts obtenue par l'entrevue de risque	45
2.12	Exemple d'analyse d'arbre de décision	46
2.13	Exemple de la simulation de coût d'un risque	47
2.14	Modèle d'apprentissage (Smith et al. 2004)	49
2.15	Cycle Plan-Do-Check-Act (PDCA)	51
2.16	Trois approches fondamentales à la gestion des inconnues	55
3.1	La procédure de la gestion des incertitudes	61
3.2	Catégorisation des incertitudes dans le processus de déve- loppement	79
3.3	Workflow de la modélisation	83
3.4	Paquetage des incertitudes	85
3.5	Paquetage des effets des incertitudes	89
3.6	Paquetage des traitements des incertitudes	92
3.7	Représentation graphique du profil "Uncertainty-of" dans le diagramme de classes (a)	96

3.8	Représentation graphique du profil "Uncertainty-of" dans le diagramme de classes (b)	97
3.9	Représentation graphique du profil "Uncertainty-of" dans le diagramme de classes (c)	97
4.1	Le schéma de synthèse de la mise en oeuvre de la méthodologie dans ce cas d'étude	103
4.2	Le principe de la TMR avec détection des erreurs	105
4.3	Première architecture proposée par les experts	106
4.4	Feuille d'identification et analyse de l'effet	107
4.5	Solution 1 : Tripler les acquisitions d'entrée critiques	109
4.6	Solution 2 : Echanger les acquisitions via les réseaux redondants	110
4.7	Diagramme de Fiabilité de l'architecture 2003 (CEI61508 2003)	111
4.8	Représentation des incertitudes liée à l'architecture préliminaire avec diagramme de classe UML	115
4.9	Représentation de l'analyse des effets de l'incertitude liée à l'architecture préliminaire avec diagramme de classe UML .	116
4.10	Représentation des traitements de l'incertitude liée à l'architecture préliminaire avec diagramme de classe UML . . .	117
4.11	Diagramme DSM des activités du processus de définition préliminaire du système	119
4.12	Algorithme du modèle de simulation de durée du processus de définition	122
4.13	Algorithme du modèle de simulation de durée du processus de définition	122

LISTE DES TABLEAUX

2.1	Catégories d'informations imparfaites	18
2.2	Briques fondamentales de modèles de processus de DS, et certains de leurs attributs (Browning et al. 2006)	23
2.3	Comparaison des propriétés des deux langages de description de processus	34
3.1	Le phasage d'un projet de développement de systèmes complexes (Blanchon et Durand 2004)	76
3.2	L'intégration des activités de la méthodologie dans un projet (a)	77
3.3	L'intégration des activités de la méthodologie dans un projet (b)	78
3.4	Description technique du profil "Uncertainty-of" dans le diagramme de classes (a)	98
3.5	Description technique du profil "Uncertainty-of" dans le diagramme de classes (b)	99
4.1	Les scénarios de tolérance aux fautes de l'architecture 2003	106
4.2	Une échelle de gravité personnalisée (en terme de risque) .	108
4.3	Résultats de l'analyse quantitative de la fiabilité et de l'indisponibilité de la première architecture proposée	113
4.4	Résultats de l'analyse quantitative de la fiabilité et de l'indisponibilité des trois architectures	113
4.5	Durées des activités de définition préliminaire du système .	120

INTRODUCTION



1.1 MOTIVATIONS

Les travaux présentés dans cette thèse, menés au sein d'une école d'ingénieurs, s'inscrivent dans le cadre général de l'ingénierie système (IS) : ils envisagent comment l'être humain pourrait mieux concevoir et développer les systèmes complexes par la prévision, la prise en compte explicite et la maîtrise des incertitudes concernant les systèmes et leur développement.

La gestion des incertitudes constitue un sujet fondamental et particulièrement important pour l'IS. L'existence d'incertitudes est une caractéristique universelle, touchant tous les domaines scientifiques et pratiques : l'économie, la physique, la géographie, la théorie de la décision et l'évaluation des risques (Antunes et Dias 2007). Les incertitudes sont devenues un objet d'étude à part entière. Elles doivent être traitées comme un élément de base au cours de développement des systèmes (DS) (Ulrich et Eppinger 2000), comme le montrent de nombreuses études et les expériences industrielles. Compte tenu de leur impact potentiel important sur la société, elles doivent bénéficier d'une approche multi-aspects (politique, économie, technique) qui nous ramène à l'ingénierie système (De Rocquigny et al. 2008).

L'accent mis sur la prise en compte explicite des incertitudes dans l'IS implique un changement culturel majeur dans la réflexion sur le paradigme de l'ingénierie. L'approche traditionnelle de l'ingénierie consiste à concevoir des cahiers des charges fixés à l'extérieur du processus d'ingénierie, dictés par exemple par le désir du client ou les réglementations

gouvernementales (De Neufville 2004). L'existence d'incertitudes est pourtant un fait incontournable, qui oblige à changer cette approche.

Il est bien sûr impossible d'identifier et de modéliser toutes les incertitudes au début d'un processus de DS, à cause des propriétés auto-émergentes des systèmes et du changement constant des environnements naturel et social. Lorsque le projet évolue, les incertitudes se réduisent en général, mais de nouvelles peuvent aussi apparaître. Il est dans tous les cas utile de gérer les incertitudes, car les managers et les ingénieurs sont obligés de prendre des décisions au début de la phase du processus de DS sans disposer d'informations complètes.

Prendre en compte les incertitudes au plus tôt est important pour la gestion du projet et permet d'examiner des scénarios dans lesquels les forces de la concurrence, les changements dans les préférences de clients, les événements politiques et économiques peuvent influencer la définition des systèmes (Smith et Merritt 2006).

Les études sur les incertitudes s'orientent en général selon deux axes : l'étude des incertitudes relatives au système complexe lui-même et l'étude des incertitudes relatives au processus global de son développement. La première s'intéresse par exemple à l'architecture, la complexité, les techniques et les critères de qualité, l'interopérabilité, la sûreté du système, etc. ; la deuxième met l'accent sur les incertitudes liées au projet, concernant l'être humain, les activités, les livrables, la planification, etc. La thèse se consacre essentiellement à ce second point : la gestion des incertitudes dans le processus de développement de systèmes complexes, même si, dans la réalité, les incertitudes liées au système et à son développement sont difficilement dissociables.

La complexité croissante des systèmes impose en effet une complexité croissante du processus de leur développement ; l'existence de multiples parties prenantes, aux objectifs différents et parfois contradictoires, accroît encore cette complexité. Il devient stratégique d'étudier les incertitudes qui en résultent et leurs effets dans le développement de systèmes complexes, afin que nous puissions découvrir les moyens de les surmonter,

voire d'en profiter lorsque de nouvelles opportunités se font jour (Pich et al. 2002).

Le travail présenté dans ce mémoire a donc pour objectif plus précis une pré-étude des moyens envisageables pour la gestion des incertitudes dans le développement de systèmes complexes.

1.2 CADRE GÉNÉRAL DE NOS RECHERCHES

Selon nous, l'amélioration du processus de développement de systèmes complexes passe par la modélisation des incertitudes inhérentes à cette activité. Bien sûr, le problème est différent si le système est déjà décrit de façon à peu près fixe (par exemple lors de l'évolution d'un système existant) ou s'il s'agit de concevoir un système innovant. Mais, dans tous les cas, il est nécessaire de penser au couplage de la conception du système et du processus de son développement.

Un concept central dans cette perspective est celui de processus ; notre modélisation va donc s'organiser autour des processus, le système et ses composants étant considérés comme des livrables du processus. Cette modélisation permettra de faciliter la communication entre les différentes parties prenantes dans le projet ; elle aidera également les individus à décrire et analyser les incertitudes attachées aux parties interdépendantes d'un système et à son processus de développement avec une sémantique précise.

Deux grandes questions se posent dans ce cadre :

1. Comment représenter des incertitudes dans la modélisation de processus de développement de systèmes complexes ?
2. Comment gérer les incertitudes pendant le processus de développement de systèmes complexes ?

L'objectif global de notre travail, en définitive, est d'améliorer la flexibilité et la prédictibilité du processus de développement de systèmes en proposant une méthodologie pour appréhender les incertitudes.

1.3 LES QUESTIONS À RÉSOUDRE

Deux niveaux doivent être distingués dans la modélisation des incertitudes dans le processus de développement de systèmes (DS) complexes.

Le premier niveau est destiné à définir un cadre de référence et un langage pour la modélisation des incertitudes : il s'agit donc d'une méta-modélisation de l'incertitude.

Le second niveau correspond à une méthodologie de modélisation des incertitudes pour un projet précis, qui doit s'inscrire parmi les processus amont de la conception et du développement de systèmes complexes.

Les questions qui se posent alors sont déclinées dans les paragraphes qui suivent ; leur résolution détaillée sera traitée dans le corps du mémoire.

1.3.1 Méta-modélisation des incertitudes

Deux étapes essentielles sont nécessaires dans ce cadre :

1. L'identification des catégories d'incertitudes qui peuvent être associées aux éléments du processus de DS. Elle est destinée à définir les abstractions nécessaires, leurs attributs, et leurs relations.
2. La méta-modélisation des incertitudes identifiées, qui permettra de produire un méta-modèle destiné à guider l'acquisition des incertitudes dans les cas concrets. Un candidat naturel pour cette méta-modélisation est le langage UML, cadre standard structuré de modélisation largement employé en recherche et en ingénierie.

D'où nos trois premières interrogations :

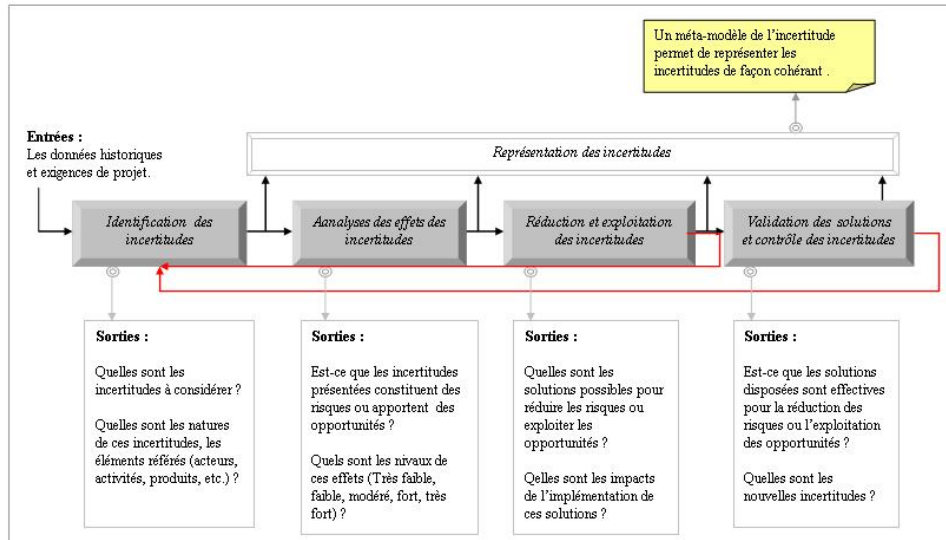


FIGURE 1.1 – La procédure de la gestion des incertitudes

- **Q1 : Quelles sont les incertitudes associées au processus de DS et leurs catégorisations ?**
- **Q2 : Est-ce qu'il est possible de modéliser les incertitudes avec UML ? Quels sont les avantages et les inconvénients ?**
- **Q3 : Si la réponse de Q2 est oui, comment modéliser les incertitudes avec UML ? La création d'un nouveau profil UML apporte-t-elle une solution ? Quelles sont les limites auxquelles on se heurtera ?**

1.3.2 Gestion des incertitudes au plus tôt dans le processus de développement de systèmes

En supposant que la décomposition initiale du système (à concevoir ou à produire en sous-systèmes) est donnée, il s'agit d'offrir un cadre aux acteurs du projet pour qu'ils puissent reconnaître et intégrer les incertitudes tout au long du processus de développement du système.

Nous cherchons une méthodologie pour gérer les incertitudes en considérant les types des incertitudes pour un projet précis. La thèse emploie le terme de "gestion" pour désigner la prévision et la maîtrise des différentes incertitudes. En plus des préoccupations traditionnelles visant

la diminution des coûts et des risques, la gestion des incertitudes met l'accent sur le développement des opportunités associées aux incertitudes. C'est le principe essentiel de la gestion des incertitudes et ce qui fait sa différence avec la gestion des risques (De Neufville 2004).

Les questions liées à la gestion des incertitudes peuvent être résumées dans la liste ci-dessous :

- **Q4 : Quelles sont les méthodes qui ont été proposées pour la gestion des incertitudes ? Quelles sont leurs forces et leurs faiblesses ?**
- **Q5 : Comment gérer les incertitudes dès le début du processus de DS ?**

Cinq actions essentielles paraissent a priori nécessaires (cf. Figure 1.1) :

1. L'identification des incertitudes. Cette action consiste à identifier les incertitudes qui se présentent durant tout le cycle de développement du système et à déterminer celles qui vont être traitées en priorité. Le niveau, le type, la prévisibilité des incertitudes sont les éléments qui pourront caractériser ces incertitudes.
2. L'analyse des effets des incertitudes. Cette action comprend l'analyse qualitative des incertitudes et des risques. Elle permet de hiérarchiser les effets des incertitudes sur l'objectif du projet et de mesurer les effets potentiels des incertitudes qui constituent des risques.
3. La réduction et l'exploitation des incertitudes. Cette action consiste à définir l'ensemble des dispositions envisageables pour réduire les risques provoqués par ces incertitudes, ainsi que pour exploiter les opportunités apportées. Elle comprend implicitement la comparaison d'alternatives et le choix d'une solution.
4. La validation des solutions et le contrôle des incertitudes. Cette action consiste à s'assurer que les choix nécessaires ont été correctement faits (par exemple, que les incertitudes susceptibles d'avoir

une influence négative sur le projet ont été effectivement réduites ou éliminées), et que les incertitudes nouvellement introduites sont prises en compte pour une nouvelle itération.

5. La représentation des incertitudes. Cette action consiste à représenter et intégrer les données incertaines ou imprécises dans la base de connaissances du projet, le méta-modèle de l'incertitude permettra de réaliser cette étape de façon cohérente.

Notre dernière question nous permettra de nous prononcer sur la validité de ce schéma :

- **Q6 : La procédure de gestion des incertitudes proposée ci-dessus est-elle pertinente ? Quelles sont ses qualités et ses faiblesses ?**

1.4 PLAN DU MÉMOIRE

Le mémoire cherche à répondre aux questions posées ci-dessus ; il est organisé en 5 parties (y compris celle-ci).

Tout d'abord, le chapitre 2 présente l'état de l'art en IS et situe notre approche par rapport au cadre de référence que nous proposons. Dans ce chapitre, nous clarifions notre vision sur l'IS et la modélisation de processus et de produit ; nous étudions les méthodes existantes concernant la modélisation et la gestion des incertitudes, et discutons leurs avantages et limites.

Le chapitre 3 détaille les contributions de notre recherche. Il est organisé en quatre parties principales. La première propose une méthodologie de la gestion des incertitudes en amont des processus de DS, et décrit les activités et les outils appropriés pour chaque étape de cette méthodologie. La seconde partie expose comment intégrer cette méthodologie dans les différentes phases d'un projet. La troisième concerne la catégorisation des incertitudes du point de vue des processus, et la quatrième partie décrit la méta-modèle des incertitudes et la construction d'un nouveau profil UML

pour les représenter.

Le chapitre 4 détaille un cas d'étude conduit dans l'industrie. Afin de tester la validité de nos propositions, nous les avons appliquées pour identifier, modéliser et analyser les incertitudes existantes dans un processus de conception.

Le chapitre 5 consacré à la conclusion résume les approches proposées dans ce mémoire, et les enseignements du cas d'étude. Les six questions posées dans le chapitre 1 sont revues. Ce que nous avons appris du cas d'étude sur chaque question de recherche est listé.

Grâce à une meilleure compréhension des incertitudes et leurs natures, nous avons trouvé les réponses à certaines des questions et nous pouvons poser de meilleures questions pour les recherches futures.

CADRE GÉNÉRAL ET PROBLÉMATIQUE

Nos travaux ont été développés dans le cadre de l'Ingénierie des Systèmes ; ils abordent plus précisément le domaine de la conception et du processus de développement de systèmes (DS) complexes, où on utilise des modèles pour décrire le produit et le processus de son développement.

Dans ce contexte, de nombreuses informations associées aux éléments (activités, livrables, rôles...) du processus sont incomplètes, vagues et pas entièrement déterminées en amont du projet et au cours de son évolution (O'Donovan et al. 2005) : il est donc souhaitable que la méthodologie de gestion du projet intègre cette dimension, aussi bien au niveau de la modélisation que dans les opérations qui s'appuient sur le modèle du processus.

Pour mieux poser notre démarche, nous présenterons dans la première section les notions de système, ingénierie des systèmes, projet et processus, puis nous détaillerons la notion d'incertitude.

La seconde section sera dédiée à la modélisation et à une revue de méthodes et outils de modélisation de processus qui prennent en compte (au moins partiellement) la représentation des incertitudes. Nous nous intéresserons notamment à la modélisation des incertitudes pour l'analyse des caractéristiques de processus et aux langages formels de modélisation de processus. A ce stade, nous allons répondre aux questions :

- Comment représenter les incertitudes dans les modèles de processus avec ces méthodes et langages de modélisation ?
- Quelles sont leurs limites et avantages ?

La troisième section sera consacrée au traitement des incertitudes et de leurs effets. Elle abordera la distinction entre risques et opportunités, et s'appuiera, à titre d'exemples, sur la gestion des risques et celle des "inconnues inconnues". Nous chercherons à répondre aux questions :

- Quels sont les effets des incertitudes ?
- Quelles sont les méthodes existant pour les analyser et gérer ?
- Quelles sont les limites et avantages de ces méthodes ?

Enfin, dans la quatrième section, nous repositionnerons la problématique de la thèse dans ce contexte du processus de DS complexes, en termes de modélisation des incertitudes comme de suivi de leur évolution tout au long du projet.

2.1 INGÉNIERIE DES SYSTÈMES COMPLEXES ET INCERTITUDES ASSOCIÉES

2.1.1 Qu'est-ce qu'un système ?

Définition 01 : Un système est un ensemble d'éléments interdépendants qui interagissent entre eux de façon organisée et forment un ensemble unique (INCOSE 2004).

Bien que l'on puisse aussi considérer des systèmes politiques, sociaux, économiques ou des écosystèmes, nous nous restreindrons à des systèmes en lien avec l'organisation matérielle de la vie humaine. Les éléments d'un tel système peuvent être divers et se composer de personnes, d'organismes, de procédures, de logiciels, d'équipements, ou d'installations.

Citons quelques exemples de systèmes complexes illustrant différents

de points de vue possibles autour de la notion de système :

- Un système de contrôle du trafic aérien international est un système organisé autour de réglementations et de procédures, dont les composants élémentaires sont de niveau relativement élevé : avions, systèmes terrestres et embarqués de transmission, centres de contrôle et de régulation du trafic, etc. Les buts essentiels de ce système sont la sécurité et la fluidité du trafic aérien, dans le cadre de la mise en oeuvre de normes internationales.
- Le système d'exploration lunaire Apollo de la NASA était composé d'un véhicule de lancement et de différents modules en étage supérieur pour accomplir le rendez-vous en orbite lunaire, la descente vers la surface lunaire et le redécollage, le retour sur Terre et la récupération. Le système comprenait également les équipes de la mission et le soutien, des véhicules d'installations de lancement, etc. Certains de ces éléments pouvaient être partagés avec d'autres systèmes ou réutilisables, comme les équipements de suivi et les stations de relais, ou encore le contrôle de mission. Il s'agit là un système d'essence technique, de grande ampleur, organisé autour de la réalisation d'une mission précise.
- La climatisation d'une voiture est un exemple de système technique de taille plus réduite, qui peut être considéré comme un élément (un sous-système) d'un système plus important, l'automobile elle-même.

Il devrait être évident que, dans le cas de grands systèmes complexes, des méthodologies et des techniques sont nécessaires pour faciliter la collaboration entre l'ensemble des sous-systèmes.

L'Ingénierie des Systèmes (IS) est une discipline de recherche, relativement récente, dont l'objet est la conception et le développement de type de systèmes complexes. Nous allons maintenant détailler les définitions et objectifs de cette discipline.

2.1.2 Ingénierie des Systèmes : définition et objectifs

Selon la définition de l'International Council on Systems Engineering (INCOSE 2004),

Définition 02 : L'Ingénierie des Systèmes est une approche et des moyens interdisciplinaires permettant la réalisation et le déploiement de systèmes réussis.

L'IS a évolué comme une branche de l'ingénierie à partir des années 1940. Pendant et après la seconde guerre mondiale, ses applications sont peu à peu rentrées dans le domaine militaire et civil (Hughes et Thomas) (Honour 2004). L'influence de l'IS a non seulement touché les scientifiques et les ingénieurs, mais également les spécialistes de la gestion et des sciences sociales.

On trouvera ci-après un tour d'horizon des points de vue qui ont été portés par les chercheurs sur l'IS, pour tenter de donner un premier contour de la discipline ; ces points de vue sont présentés par ordre chronologique :

1. Systems engineering is to recognize that the system being designed and developed is a whole, not just parts (chestnut 1965).
2. Systems engineering must understand the requirements on the system (chestnut 1965).
3. System engineering must also address the integration, verification and validation of the systems (Martin et Thomas 1997).
4. Systems engineering is the management of the technical development of a system. Systems engineering must communicate across many groups involved in the development. System engineering is a teamwork effort (Martin et Thomas 1997).
5. Systems engineering is about tradeoffs of different objectives. One must optimize the overall system rather than parts (Stevens et al.

1998).

6. Systems engineering must pay attention to the dynamics and life-cycle of system. This includes considering the "-ilities" of the system in addition to the function of the system during design. Examples of "-ilities" are reliability, maintainability, etc. (Blanchard 1965).

Nous pouvons voir que l'IS est une approche interdisciplinaire concernant : le système, les parties prenantes, le projet, les exigences des utilisateurs, le processus de développement de systèmes, et la synergie du travail en équipe.

L'IS a débouché, compte tenu de la complexité croissante des systèmes et exigences, sur des outils et techniques destinés au développement des systèmes et à l'amélioration de leurs performances ainsi qu'à la gestion de projets (leurs performances techniques et leur qualité, les délais de livraison, et les coûts).

Au total, l'objectif de l'IS est la conception et la réalisation de systèmes complexes. Elle peut être vue comme l'application de techniques à l'ingénierie, aussi bien que comme l'application d'une approche systématisée aux efforts d'ingénierie, pour guider et coordonner au mieux l'effort de toutes les disciplines concernées par la conception et la réalisation de systèmes complexes.

2.1.3 Les projets en tant que systèmes

Définition 03 : Un projet est un processus d'action construit à partir d'un diagnostic de situation, du choix d'objectifs cohérents par rapport à cette situation et de différents moyens à mettre en oeuvre pour répondre à une problématique précise. Les actions, articulées entre elles, visent à transformer la "situation initiale insatisfaisante" en "situation résultante satisfaisante". Les résultats sont mesurables et programmés dans le temps. L'élaboration est collective tout comme son objet (Insi-

tute 2003).

La littérature en IS met surtout l'accent sur les caractéristiques des produits et des services; mais un projet peut également être considéré comme système.

C'est ce que font Fricke et al. (2000) avec leur modèle ZOPH¹(Figure 2.1), qui précise les rôles de différents sous-systèmes : le système de produits, le système de processus, le système d'agents et le système d'objectifs, dans un projet de développement de systèmes complexes.

Chaque système est lié aux autres :

- le système de produits comprend les produits intermédiaires et le résultat attendu du projet,
- le système de processus comprend les activités à accomplir pour fabriquer le produit,
- le système d'agents comprend les ressources et méthodes nécessaires pour la réalisation des objectifs. Il comprend l'organisation², les technologies, outils, etc.
- le système d'objectifs garantit la cohérence des trois autres systèmes entre eux, et avec les exigences extérieures et l'environnement du projet.

2.1.4 Les processus dans le développement de systèmes

Définition 04 : Un processus d'IS est un ensemble prédéfini d'activités utilisées pour accomplir des tâches d'Ingénierie Système (INCOSE

1. ZOPH est l'acronyme allemand pour Zielsystem (système d'objectifs), Objektsystem (système de produits), Prozeßsystem (système de processus), et Handlungssystem (système d'agents).

2. L'organisation se compose des structures et des personnes travaillant sur le produit, c'est-à-dire des individus, des groupes, des équipes, ou d'autres unités d'organisation, liés les uns aux autres par la communication, les rapports, etc.

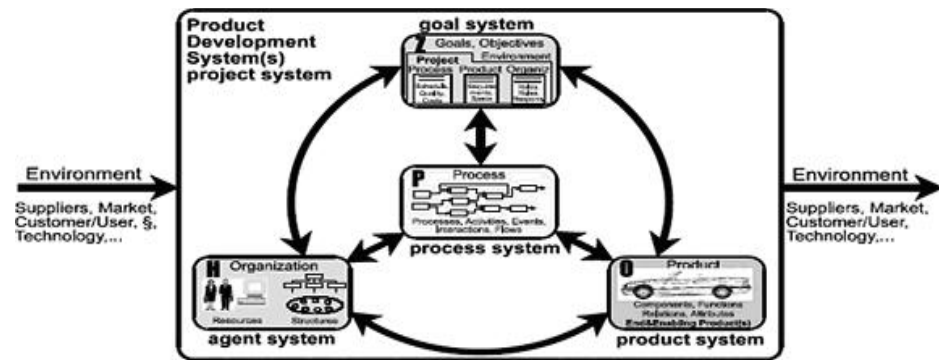


FIGURE 2.1 – Modèle ZOPH (Fricke et al. 2000)

2004).

Cette définition implique que l'on peut considérer les travaux effectués dans un projet comme un grand processus. Dans cette thèse, nous étudierons le développement de systèmes complexes comme un processus dont l'objectif est de créer une "recette" pour développer un système nouveau ou améliorer un système existant, conformément aux exigences du client ou aux besoins du marché.

Les processus représentent un aspect essentiel dans la théorie et les pratiques de l'IS. Un processus est également une sorte de système qui a reçu beaucoup moins d'attention en tant que tel (Browning 2001).

Le modèle de processus est la base et la clé de l'intégration efficace des modèles de systèmes de projet, et de la gestion efficace des projets (Browning 2001). Par conséquent, un bon modèle de processus doit accorder une attention particulière aux interfaces entre les activités, et doit prendre en compte également les incertitudes sur chaque élément de processus et les relations entre eux.

En ce qui concerne le développement de systèmes, nous choisissons la définition suivante :

Définition 05 : Le développement de systèmes (DS) est constitué de la multitude d'activités multifonctionnelles qui prennent place entre la définition d'une technologie ou d'une opportunité de marché et le début de la production (Browning et al. 2006).

Un processus de DS est donc "un ensemble d'activités inter-reliées et menées dans le but de produire un produit" (Hammer 2001).

Selon l'Electronic Industries Alliance, il s'agit d'une approche systématique, itérative, qui procède étape par étape et qui comprend la définition du système, la spécification, la conception, la réalisation, la validation, l'intégration, la modification et la planification du projet (EIA 1999). On peut au contraire considérer, comme par exemple (Fricke et al. 2000), que les processus de DS s'organisent naturellement en réseaux plutôt qu'en chaînes d'activités séquentielles.

Les processus de DS diffèrent des autres processus d'entreprise par un nombre important d'aspects. Le processus d'entreprise conventionnel a tendance à être plus séquentiel, alimenté par des dépendances fermes sur les matériels et les données, tandis que le processus de DS a tendance à fonctionner plus en parallèle (ne serait-ce que par l'intégration de l'ingénierie concurrente), car les dépendances sur les informations sont plus "souples".

Ensuite, les dépendances rencontrées dans des processus d'entreprise sont plus claires que celles des processus de DS. Par exemple, la communication dans un processus d'entreprise se déroule d'une manière connue par les clients et les fournisseurs. Cela est moins vrai dans celui de DS, car un certain nombre d'hypothèses et d'interactions peuvent ne pas être documentés, en raison d'un plus grand flou et de davantage d'ambiguïté dans les actions et interactions.

De ce fait, le processus de DS doit être relativement plus souple et plus agile que celui d'entreprise. Comme la créativité et l'innovation ne sont pas linéaires, le processus de DS est en évolution constante en fonction de l'état du projet par l'ajout ou l'élimination d'activités, d'interactions et d'itérations. Le processus d'entreprise, par contre, est moins ambigu (plus facile à définir). L'incertitude, l'ambiguïté, et le risque sont plus élevés dans les DS (Schrader et al. 1993) (Pich et al. 2002) et la plupart des organisations tendent à tolérer un risque plus élevé dans le DS qu'elles ne

le font dans les opérations courantes.

Toutes ces différences entre les processus ordinaires d'entreprise et les processus de DS ont des implications importantes pour le langage de modélisation de processus et l'approche adoptée pour la construction du modèle.

2.1.5 Incertitudes dans le développement de systèmes complexes

Dans cette section, nous cernons d'abord la notion d'incertitude, en la comparant à la notion d'imprécision; nous considérons ensuite les sources potentielles d'incertitudes; enfin, nous distinguons les incertitudes du point de vue de leur prévisibilité.

2.1.5.1 Incertitudes et imprécisions

De façon générale, on utilise le terme "incertitude" pour faire référence à des doutes sur la validité et la vérité d'une information, alors que le terme "imprécision" se réfère à des difficultés dans l'expression de l'information, par exemple, si elle n'est pas complètement connue ou si elle ne décrit pas exactement les caractéristiques ciblées (Smets 1999) (Ben Amor 2004). La notion d'"imperfection" réunit les deux notions d'incertitude et d'imprécision (Smets 1999).

Les auteurs ont défini des catégories pour ces deux termes (Bonissone et Tong 1985) (Bouchon et Meunier 1995) (Bosc et al.) :

- trois catégories pour l'incertitude : la probabilité, la crédibilité, la possibilité ;
- quatre catégories pour l'imprécision : l'appartenance à un ensemble, l'appartenance à un intervalle, l'incomplétude et le flou.

Nous illustrons la catégorisation des imperfections que nous avons extraite de leurs études par le Tableau 2.1 ci-dessous.

Cette classification est utile pour l'analyse de l'imperfection de l'infor-

mation (par la logique floue, la théorie des probabilités, etc.), mais elle ne suffit pas pour représenter sa structure dans les processus de développement de systèmes. Nous allons donc proposer une autre catégorisation dans le chapitre suivant, en nous limitant au cas le plus fréquent, où l'imprécision peut être considérée comme une source de l'incertitude.

Catégorie	Sous-catégorie	Exemple
Incertitudes	Probabilité	Quelles sont mes chances de gagner au casino ?
	Crédibilité	Je suppose que les participants au sondage n'ont pas menti.
	Possibilité	La possibilité que l'équipe A marque 3 buts.
Imprécisions	Ensemble	J'ai fait un voyage en Europe. La Chine n'appartient pas à l'ensemble des pays possibles.
	Intervalle	La durée de vie habituelle d'une voiture est entre 8 et 20 ans.
	Incomplétude	« La couleur de la voiture est rouge » : l'attribut couleur n'est pas suffisant pour décrire une voiture.
	Flou	Une femme jeune : quel âge a-t-elle ?

TABLE 2.1 – Catégories d'informations imparfaites

2.1.5.2 Sources des incertitudes

De toute évidence, prendre une décision dans la conception et le développement d'un système d'ingénierie ou dans la planification, peut poser des difficultés importantes. L'incertitude peut intervenir sur des quantités ou sur les modèles destinés à représenter le comportement du système et/ou le processus de développement de systèmes.

Fondamentalement, la littérature cite cinq sources d'incertitude différentes (Smets 1999) (Ben Amor 2004) (McManus et Hastings 2005) :

1. Les informations incomplètes
2. Les aléas
3. L'imprécision linguistique
4. Le changement
5. La complexité

– **Source 1. Informations incomplètes.** Elles sont présentes quand un facteur (ou une donnée, un paramètre) dans une décision ou dans un modèle n'est tout simplement pas connu. Parfois, cela peut être

résolu (par la recherche, une enquête, etc.) mais pas toujours : certains facteurs sont parfois nécessairement incertains (par exemple, le budget de la recherche pour l'année 2060). Parfois encore, bien que techniquement déterminé, un facteur peut ne pas être pratiquement mesurable (par exemple, le nombre de personnes assises en Europe en ce moment).

- **Source 2. Aléas.** L'incertitude peut aussi résulter d'une variation inhérente associée à un système physique ou dans un environnement à l'examen. Ces incertitudes peuvent généralement être mesurées par leur quantité de distribution qui peuvent prendre des valeurs dans une gamme connue, mais dont la valeur exacte est différente selon le hasard de l'unité ou le temps.
- **Source 3. Imprécision linguistique.** L'imprécision linguistique est extrêmement commune. Parce que peu de précision est requise pour la communication générale, les gens ont souvent l'habitude d'utiliser des termes et expressions imprécis. Malheureusement, dans un contexte où l'exactitude est importante, cette imprécision peut engendrer des incertitudes.
- **Source 4. Changement.** La quatrième source d'incertitudes se réfère simplement au changement. Le phénomène de changement (quelle qu'en soit la raison) peut donner lieu à des incertitudes. Notons que certains changements peuvent être provoqués par des causes physiques (par exemple, sur la base de la mécanique quantique) ou venir de l'influence de l'homme (les changements du budget de développement de systèmes spatiaux en raison de l'évolution des forces politiques) ou du marché économique (la crise financière internationale à partir de 2008).
- **Source 5. Complexité.** La dernière source d'incertitudes est la complexité du système ou du projet. Les incertitudes sont dues dans ce cas aux interconnexions et aux interactions des éléments du système. Dans un projet, l'incertitude concerne les activités et les

parties prenantes.

Ces cinq types de sources d'incertitudes peuvent également s'associer dans certaines situations. C'est le cas, par exemple, lorsqu'une modification sur une activité en amont provoque des modifications sur l'aval : les incertitudes sont alors à la fois de sources 4 et 5. Les sources d'une incertitude se trouvent ainsi souvent mélangées dans un système complexe. De plus, les incertitudes évoluent, et une incertitude peut changer de nature au cours des évolutions et des itérations du processus.

Cela signifie que l'identification d'incertitudes dans les processus de DS doit croiser les regards et les synthétiser de façon évolutive et itérative. C'est pourquoi établir un modèle pertinent est important pour décrire les incertitudes dans ce contexte.

2.1.5.3 Prévisibilité des incertitudes

Certains chercheurs De Rocquigny et al. (2008) McManus et Hastings (2005) catégorisent les incertitudes par leur prévisibilité dans le cadre de développements de systèmes complexes : ils distinguent entre "inconnues connues" et "inconnues inconnues".

- **Inconnues connues** : des choses que l'on sait ne pas connaître. Leurs valeurs sont au mieux encadrées, et peuvent être totalement inconnues. On les associe en général à des risques ; par exemple, la fiabilité d'un équipement peut être caractérisée par son temps moyen de bon fonctionnement (MTBF).
- **Inconnues inconnues** : des choses qui restent imprévisibles quelle que soit la quantité de données disponibles. Par définition, "on ne les connaît pas" et elles peuvent représenter un risque ou une opportunité ; par exemple, il peut s'agir d'un changement du marché suite à l'apparition ou à la chute d'un compétiteur.

Les approches de la gestion des incertitudes doivent être différentes selon la prévisibilité des incertitudes De Rocquigny et al. (2008). La plu-

part des méthodes de la gestion des incertitudes sont orientées vers les "inconnues connues", parce que plus de 90% des activités et des réalisations d'un projet peuvent être anticipés à partir d'un projet antérieur (Austin et al. 2000).

La présence d'inconnues inconnues peut concerner l'ensemble des éléments d'un processus de développement ; la modélisation du processus devrait donc explicitement prévoir leur possibilité d'apparition et leurs conséquences négatives ou positives.

Nous allons maintenant voir quelques méthodes et langages de modélisation de processus qui prennent en compte (au moins partiellement) la représentation et l'intégration des incertitudes.

2.2 MODÉLISATION DES INCERTITUDES DANS LA MODÉLISATION DE PROCESSUS DE DS

La modélisation de processus est un domaine très actif actuellement. La littérature sur les langages de modélisation et la multitude d'outils existant est extensive, cependant seule une petite partie de cette littérature aborde spécifiquement les processus de DS. La plus grande partie de la littérature est axée sur la modélisation des processus d'entreprise (Business Process Modeling), qui est différente de celle utilisée pour les processus de DS, ce qui est logique compte-tenu des différences de nature entre ces processus expliquées dans la section 2.1.4. Un panorama complet des langages de modélisation de processus de DS est présenté dans (O'Donovan et al. 2005) (Smith et Morrow 1999).

Dans cette section, nous présentons d'abord les éléments fondamentaux des modèles de processus ; puis nous évoquerons des méthodes de modélisation des incertitudes dont le but est l'analyse des caractéristiques des processus, et enfin certains langages formels de modélisation des processus. Dans chaque cas, les limites et avantages de ces représentations

seront examinés.

2.2.1 Modèles de processus et leurs éléments fondamentaux

Définition 05 : Un modèle est une représentation abstraite de la réalité qui est construite, vérifiée, analysée, et manipulée pour mieux comprendre cette réalité (Browning et al. 2006).

En effet, la modélisation favorise la prévision et le test des hypothèses afin d'évaluer les effets de certaines des actions considérées dans le monde réel quand ces actions sont trop difficiles ou coûteuses à essayer dans la pratique. Par exemple, selon (Ralyté 1999), du point de vue de la conception, un modèle utile doit être capable de permettre la détermination (avec une bonne résolution) du meilleur parmi les candidats en compétition pour une conception en présence d'incertitudes.

Définition 06 : Un modèle de processus prescrit une manière de faire, une démarche méthodologique pour atteindre la cible souhaitée. Il décrit à un niveau abstrait et idéal, la façon d'organiser la production du produit : les étapes, les activités qu'elles comprennent, leur ordonnancement, et parfois les critères pour passer d'une étape à une autre. Il joue le rôle de 'moule' des processus d'ingénierie. Le produit est donc le résultat d'application de cette démarche (Ralyté 1999).

Un modèle de processus peut être descriptif, normatif, ou les deux (Hazelrigg 1999). Un modèle de processus descriptif tente de saisir les connaissances sur la façon dont le travail est fait. Il essaie de décrire les caractéristiques principales de "comment est" la réalité. Il est construit par induction. A l'opposé, un modèle de processus normatif indique quel travail faire et parfois aussi comment le faire. Il est construit par déduction, par exemple à partir d'une norme à laquelle il se réfère.

Dans cette thèse, nous nous intéressons à la fois à des modèles de processus descriptif qui aident les ingénieurs et chefs de projets à représenter et comprendre les incertitudes, et à des modèles de processus normatif qui aident à mieux gérer ces incertitudes pour améliorer les processus de DS.

Après avoir introduit les modèles de processus, nous allons à présent considérer dans la suite les éléments fondamentaux de ces modèles.

Définition 07 : Un élément de processus est une référence générique à une unité de travail. Il peut être tout processus, une activité, une tâche, etc. Et il peut faire partie d'un élément de processus plus large Browning (2001).

Parmi les éléments fondamentaux utilisés pour la modélisation des processus, on trouve l'activité et le livrable. Le Tableau 2.2 illustre de façon détaillée (mais non exhaustive) les attributs possibles des activités et des livrables, qui peuvent être utilisés pour "marquer" les informations pertinentes (ou les liens d'information) à ces deux éléments.



 Process/Activity Object Attributes	 Deliverable Object Attributes
<ul style="list-style-type: none"> • Name • Description • Mode • Inputs (and sources, timing, sensitivity) • Outputs (and sinks, timing) • Parent process • Constituent activities ("children") • Metrics (duration, cost, uncertainty, risk, criticality, switching cost, budget, deadline, capability, robustness, flexibility, productivity, percent complete, improvement curve, etc.) • Resources (tools, facilities, etc.)—pointers to tool system • Roles (training, skills, assignments, skill gaps)—pointers to organization system • Organizational owner; assigned resources • Entry criteria • Exit criteria • Work policies, business rules, design rules [Baldwin and Clark 2000] • etc. 	<ul style="list-style-type: none"> • Name • Description • Mode • Supplier(s), with agreement and commitment from each • Consumer(s), with agreement and commitment from each • Parent deliverable • Constituent deliverables ("children") • Requirements (what is agreed and committed to) • Constraints (regulations, policies, etc.) • Verification and Validation criteria • Metrics (status, confidence, maturity, quality, value added, due date, time until ready, lateness penalty, complexity, time last modified, percent complete, etc.) • Format • Medium of transmission • Pointer to actual artifact/data (perhaps in product system) • etc.

TABLE 2.2 – Briques fondamentales de modèles de processus de DS, et certains de leurs attributs (Browning et al. 2006)

Définition 08 : Une activité est une unité de travail définie par ses entrées, ses sorties, les ressources utilisées, et potentiellement par d'autres attributs (nom, description, coût, délais, etc.) comme indiqué dans le Tableau 2 ci-dessus (Fricke et al. 2000).

Définition 09 : Un délivrable est l'entrée ou la sortie d'un élément de processus. Un délivrable peut être un produit, un service, une autorisation, une information, un logiciel, un équipement, un plan, le résultat d'une décision, etc...(Browning et al. 2006).

Définition 10 : Une entrée est un délivrable en provenance d'un élément de processus fournisseur, utilisé par un élément de processus consommateur (Browning et al. 2006).

Définition 11 : Une sortie est un délivrable issu du travail d'un élément de processus fournisseur, en principe utile à l'élément consommateur (Browning et al. 2006).

Dans cette modélisation, les activités sont donc les éléments constitutifs d'un processus. Elles sont des 'paquets de travail' à faire pour produire les résultats escomptés du projet, et elles sont représentées par des 'boîtes' dans le modèle de processus. Elles ont besoin de produits délivrables comme entrées; elles produisent des délivrables en sortie. Elles consomment ou utilisent des ressources comme le temps, l'argent, la main-d'oeuvre, des outils, etc. Les délivrables sont donc les connecteurs du modèle de processus; ils représentent toutes les informations, les données, les matériels, etc., requis ou produits par une activité.

Au-delà de ces deux types d'éléments, un modèle de processus peut inclure d'autres éléments tels que :

1. Les unités de l'organisation (personne, équipe, entreprise, etc.).
2. Les outils (modèles, système informatique, logiciels d'application, etc.).
3. Les éléments du produit (sous-systèmes, composants, etc.).
4. Les objectifs (exigences, budgets, etc.).
5. Les évènements planifiés ou non prévus.

2.2.2 Modélisation des incertitudes pour l'analyse des caractéristiques des processus

Les méthodes qui suivent sont des techniques de modélisation des incertitudes (comme le budget, le délai, les risques associés) qui permettent d'analyser certaines des caractéristiques des processus. Elles permettent de maîtriser la complexité des processus, de visualiser les dépendances entre activités et de procéder à leur ordonnancement.

2.2.2.1 PERT / Critical Path Method / Gantt / Design Structure Matrix

PERT, Critical Path Method (CPM) et les diagrammes de Gantt sont des techniques répandues pour le management des projets (Steward 2001). Les détails concernant chaque technique peut être retrouvés dans la plupart des ouvrages de gestion de projets, par exemple (Moder et Phillips 1995) (Steward 2001).

PERT est l'abréviation de Program Evaluation and Review Technique. On l'utilise essentiellement pour prévoir la date de fin d'un projet en prenant en compte des incertitudes portant sur la durée des tâches, leur début et leur fin. Cette méthode utilise la distribution de probabilités sur le temps de terminaison de chaque tâche pour mieux prédire la fin du projet. Dans la Figure 2.2, à chaque tâche sont associés :

- Un Début : Date de début de la tâche
- Une Fin : Date de fin de la tâche
- Une Durée : Généralement en jours, c'est la différence entre les dates de début et de fin de la tâche
- Des Liens : Ce sont les liens entre les tâches permettant de déterminer l'enchaînement des tâches.

Le diagramme PERT permet d'identifier le chemin critique d'un projet et de concentrer les efforts sur les tâches le composant, le chemin critique étant l'enchaînement de tâches dont la durée totale est la plus longue du projet. Souvent représenté en rouge, il est facilement déterminable grâce

au diagramme PERT.

CPM est une technique développée par Remington Rand et Dupont. Elle propose un mécanisme d'évaluation des actions alternatives et consécutives pour mieux les choisir et les réaliser. Elle aide à programmer des tâches en parallèle. Elle représente un moyen d'analyser la durée d'un projet à partir du temps dépensé pour chaque activité.

Un diagramme de Gantt présente le programme des tâches (scheduling) sur l'axe du temps. Il peut montrer les activités sur les chemins critiques ou pas. Pour celles qui ne sont pas sur les chemins critiques, il peut indiquer leurs temps de terminaison au plus tôt et au plus tard. Mais s'il permet de visualiser les résultats d'une étude basée sur PERT ou sur CPM, il n'est pas par lui-même un outil d'analyse.

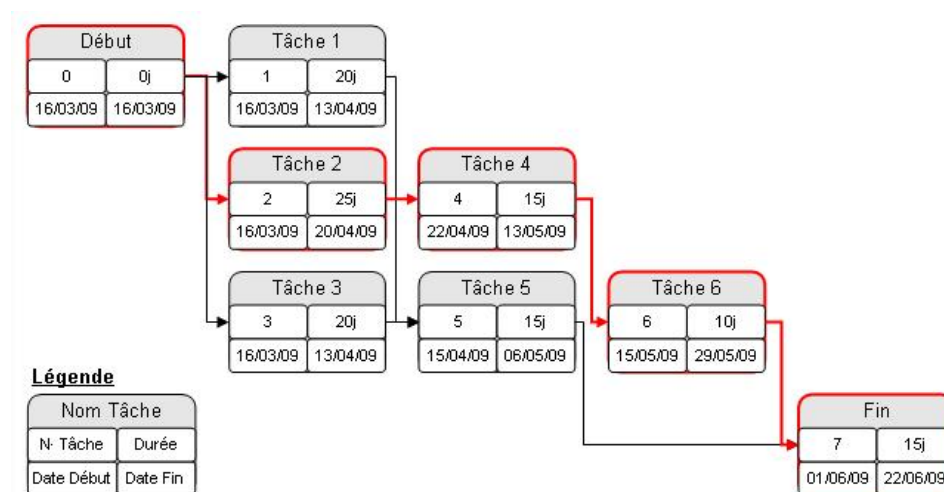


FIGURE 2.2 – Exemple du diagramme PERT (Marcel 2002)

La méthode Design structure Matrix (DSM) a une longue histoire. L'utilisation de matrices dans la modélisation de systèmes peut être retracée à Warfield dans les années 70 et Steward dans les années 80. Toutefois, ce n'est pas avant les années 1990 que cette méthode a reçu de l'attention et a été largement diffusée. Grâce à la recherche du MIT dans le domaine de la modélisation du processus de conception, cette méthode est de plus

en plus répandue.

Dans DSM, il y a trois configurations de base (cf. Figure 2.3) pour décrire la relation entre les éléments du système : parallélisme (ou simultanéité), séquentialité (ou dépendance) et couplage (ou interdépendance).

Si le système est un projet, alors les éléments du système seront les tâches à accomplir.

Dans une relation de parallélisme, les éléments du système n'ont pas d'interaction les uns avec les autres et l'activité B est indépendante de l'activité A : aucun échange d'informations n'est nécessaire entre les deux activités.

Dans une relation séquentielle, un élément influence le comportement ou la décision d'un autre élément, et l'activité A doit être effectuée avant que l'activité B puisse commencer.

Enfin, dans une relation couplée, le flux d'information est étroitement lié : l'activité A influence B et l'activité B influence A. Les itérations qui résultent de ces couplages doivent être éliminées ou réduites, si possible, au maximum. Si c'est impossible, les activités peuvent être regroupées en sous-cycles itératifs. Par exemple, dans la Figure 2.4, les activités (1, 2 et 3) et les activités (6, 7, 8, 9 et 10) sont regroupées en deux sous-cycles itératifs.

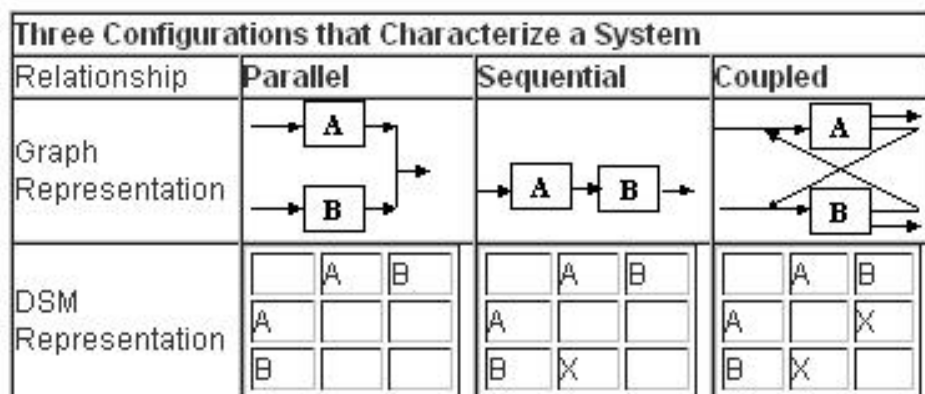


FIGURE 2.3 – Configurations de base des diagrammes de DSM

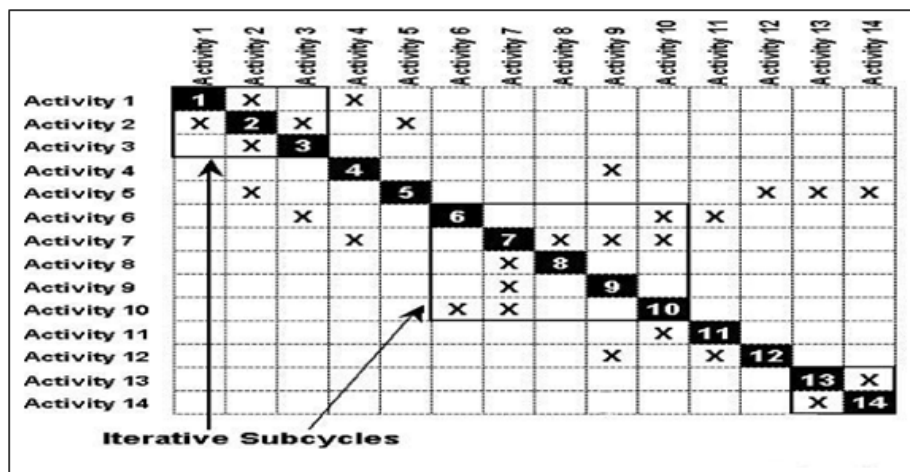


FIGURE 2.4 – Exemple de DSM (Qi 2000)

2.2.2.2 Limites et conclusions

Les trois premières méthodes citées (PERT, CPM, Gantt) permettent de capturer et analyser efficacement les processus de management d'un projet. Mais elles ne sont pas applicables lorsqu'il existe des itérations entre activités d'un processus complexe, ou lorsqu'on est confronté à des reprises de tâches.

Par rapport à ces trois techniques, la méthode DSM a deux principaux avantages. Tout d'abord, elle peut représenter un grand nombre d'éléments du système et leurs relations d'une façon compacte et faire ainsi ressortir les caractéristiques importantes des données (telles que des boucles de feed-back). Deuxièmement, l'analyse par DSM peut être utilisée pour gérer les effets du changement. Par exemple, si le cahier des charges d'un élément a dû être changé, il est possible, sur la base des informations actualisées, d'identifier rapidement tous les processus ou activités qui ont été tributaires de cette spécification, en réduisant les risques intervenus dans le travail ultérieur.

En conclusion, les méthodes de modélisation des incertitudes pour l'analyse des caractéristiques de processus permettent de mieux gérer la

complexité des systèmes ou des projets, en mettant en évidence les flux d'informations, les séquences et les itérations de tâches. Elles peuvent aider les équipes à rationaliser les processus de DS en optimisant les flux d'informations entre les différentes activités interdépendantes. Elles sont très utilisées pour la planification de projets et fournissent un outillage intéressant pour la gestion des connaissances, l'apprentissage organisationnel, etc.

Mais ces méthodes ont deux principales limites face au problème de l'incertitude. Premièrement, leurs modèles sont très tournés vers la modélisation des incertitudes liées aux niveaux de coût et délais, mais ils ignorent l'existence des incertitudes liées aux autres éléments de processus, comme les livrables, l'humain, etc. Ces incertitudes qui ne sont en effet pas exprimées ou prises en compte explicitement par ces techniques. Deuxièmement, du point de vue de la prévisibilité des incertitudes, ces modèles sont plutôt applicables à l'analyse des "inconnues connues", parce qu'ils supposent que l'ensemble des activités des processus sont prévisibles, ce qui n'est pas toujours vrai dans la réalité.

2.2.3 Modélisation des incertitudes par langages formels

Un langage formel est un langage qui utilise un ensemble de termes et de règles syntaxiques pour permettre de communiquer sans aucune ambiguïté (par opposition à un langage naturel) (OMG 2003b). Nous allons discuter les éléments principaux de deux langages qui autorisent la représentation et l'intégration des incertitudes aux modèles de processus : UML (Unified Modeling Language) et SPEM (Software Process Engineering Metamodel).

2.2.3.1 Le langage UML

UML est né de la fusion des trois méthodes (OMT, Booch et OOSE)³ qui ont le plus influencé la modélisation objet au milieu des années 1990. UML est un cadre standard structuré de modélisation, qui est largement appliqué et combiné avec les méthodes de l'entreprise pour le développement de systèmes industriels (Sigel 2007). UML n'est pas seulement utile pour le Génie Logiciel, mais aussi pour l'Ingénierie des Systèmes (OMG 2005). L'utilisation d'UML est possible pour la modélisation de processus de DS (Rochet 2008).

Des 13 diagrammes qu'UML offre, cinq sont particulièrement utiles pour décrire des processus de DS (Unhelkar 2003) (Chantal et al. 2005) :

- le diagramme de communication : il met en évidence les interactions entre objets ;
- le diagramme de séquence : il est une variante du diagramme de communication. Il permet de visualiser la séquence des messages entre objets ;
- le diagramme d'états : il associe des états aux objets et fait apparaître un ordonnancement entre ces états ;
- le diagramme d'activité : il constitue une variante du diagramme d'états où les états sont des activités ;
- le diagramme de cas d'utilisation : il représente des interactions entre les acteurs et le système au travers de cas identifiés.

A titre d'exemple, nous donnons un diagramme de séquence dans la Figure 2.5, pour montrer comment on peut utiliser UML pour décrire les activités qui concourent à un processus. Les différentes périodes d'activité d'un objet *y* sont représentées au moyen d'une bande rectangulaire superposée à la ligne de vie de l'objet. On peut aussi représenter des opérations

3. La méthode OMT (Object Modeling Technique, créée par James Rumbaugh), la méthode OOSE (Object Oriented Software Engineering, créée par Ivar Jacobson) et la méthode Booch (du nom de son concepteur, Grady Booch) sont toutes orientées objet. Leur fusion donnera UML en 1995.

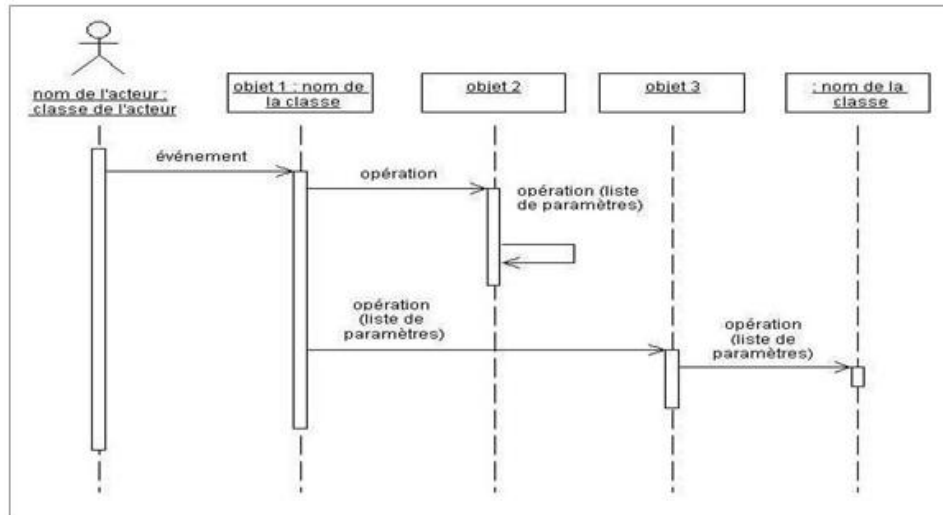


FIGURE 2.5 – Exemple de diagramme de séquence

récurrentes (cf. l'objet 2 de la figure).

Si un des intérêts principaux du langage UML est de gagner en précision sur la description d'un objet et de faciliter la compréhension de représentations abstraites complexes, un autre intérêt de ce langage est de permettre la création de nouveaux concepts grâce à ses mécanismes d'extension, qui lui permettent de s'adapter aux besoins descriptifs de l'utilisateur.

Mécanismes d'extension d'UML

Trois mécanismes d'extension permettent d'étendre UML au niveau du méta-modèle pour inclure de nouveaux concepts : les stéréotypes, les valeurs marquées (TaggedValues) et les contraintes. La Figure 2.6 illustre le méta-modèle du mécanisme d'extension d'UML.

1. Un stéréotype, noté «...», est utilisé pour introduire une nouvelle classe dans le méta-modèle par dérivation d'une classe existante. Les utilisateurs peuvent ainsi ajouter de nouvelles classes d'éléments de modélisation au méta-modèle, en plus du noyau prédéfini par UML (Muller 2000). Nous introduirons ainsi une méta-classe supérieure, «Uncertainty-of», qui représentera la famille des incertitudes asso-

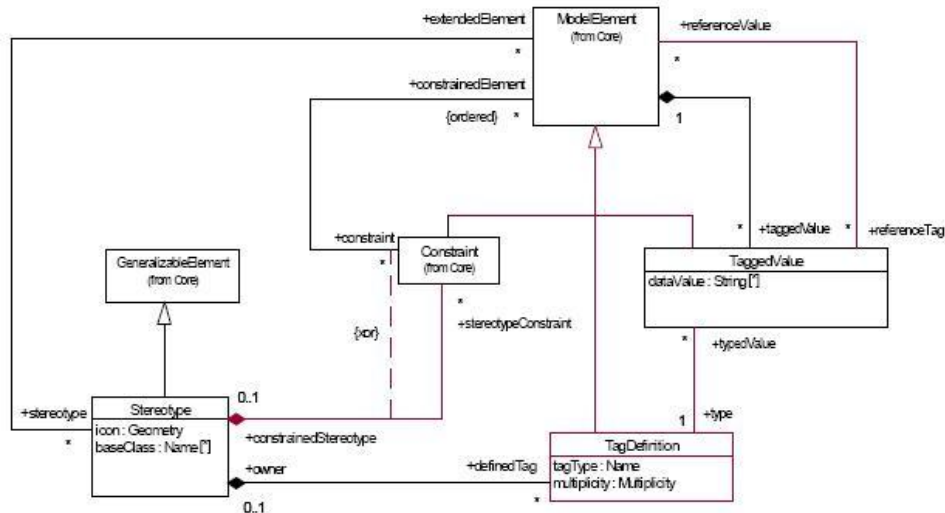


FIGURE 2.6 – Méta-modèle UML : Mécanisme d'extension

ciées aux différents éléments du processus.

2. Une valeur marquée (tagged value), indiquée entre accolades ..., est une paire (nom, valeur) qui ajoute une nouvelle propriété à un élément de modélisation (Muller 2000). Elle peut être appliquée à un stéréotype préalablement défini. Dans notre cas, nous utiliserons un attribut {Is_uncertain = vrai} de type booléen pour "Uncertainty-of" pour indiquer l'état d'une incertitude portant sur un attribut d'un élément.
3. Une contrainte, exprimée en langage naturel ou en OCL (Object Constraint Language), est une relation sémantique quelconque entre éléments de modélisation qui définit des propositions devant rester vraies pour que la validité du système modélisé soit garantie (Muller 2000). Par exemple, nous définirons une contrainte sur un attribut "value" par {1 < value < 10}. L'utilisation d'une valeur permet aussi de décrire une contrainte.

Un ensemble défini par ces trois concepts est appelé un profil (OMG 2003b). Les chercheurs et les ingénieurs définissent et utilisent des profils UML pour leur domaine particulier. C'est ce que nous ferons dans le chapitre 3 dans le domaine des processus de DS.

2.2.3.2 Le langage SPEM

SPEM (Software Process Engineering Metamodel) est une extension d'UML, proposée par l'OMG (SPEM 2005), qui fournit un méta-modèle du processus de développement logiciel.

SPEM fournit une définition des termes usuels et une description de la notion de composant de processus. SPEM a la particularité d'être défini à la fois comme un méta-modèle à part entière et comme un profil UML.

De manière simplifiée, SPEM décrit un processus comme étant composé (cf. la Figure 2.7) :

- d'activités : les unités de travail. Elles précisent les opérations à appliquer sur les produits de travail ;
- de produits de travail : les résultats ou les entrées des activités ;
- de rôles : les responsabilités par rapport aux produits de travail et à la réalisation des activités.

Les activités sont réalisées par un rôle unique et peuvent produire ou utiliser un nombre quelconque de produits de travail. Un même rôle peut réaliser plusieurs activités et être responsable de plusieurs produits de travail.

Ce méta-modèle a connu deux versions : la version 1.1 (SPEM 2005) de janvier 2005 et la version 2.0 (SPEM 2007) adoptée le 3 mars 2007. Il semble pour le moment encore peu employé dans les projets d'IS.

Comme SPEM est un profil UML, les mécanismes d'extension d'UML sont théoriquement applicables, et peuvent donc répondre à nos besoins ; malheureusement, il n'existe pas à l'heure actuelle, à notre connaissance, d'outil spécifique permettant de réaliser facilement cette extension.

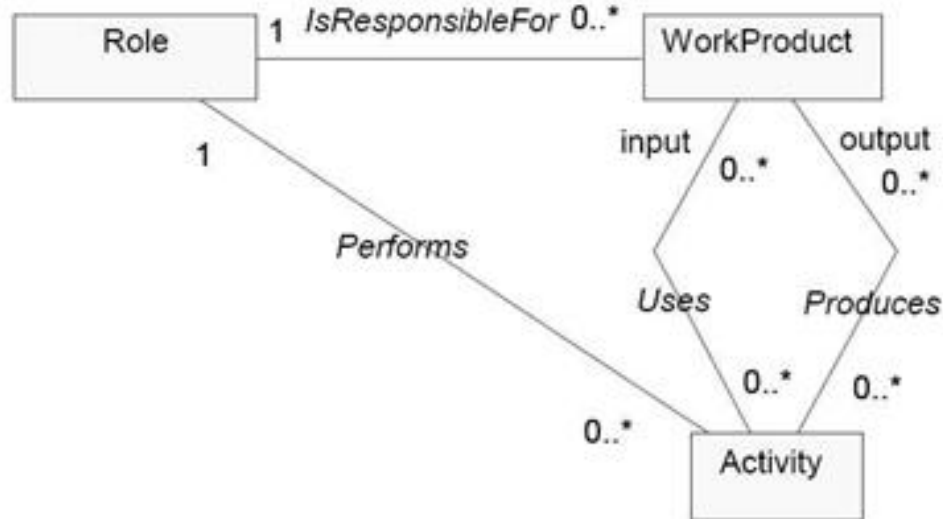


FIGURE 2.7 – Modèle conceptuel de SPEM

2.2.3.3 Limites et conclusions

Les deux langages formels évoqués ci-dessus sont des supports de communication performants qui concentrent l'analyse sur les éléments de processus, et facilitent la compréhension de représentations abstraites complexes. Les principales propriétés de ces deux langages de description de processus sont répertoriées sur le Tableau 2.3.

	UML	SPEM
Meta-modèle	Oui	Oui
Mécanisme d'extension	Oui	En tant que profil UML
Popularité industrielle	Très populaire	Peu employé
Supporté par des outils pour la création de profils	Oui	En tant que profil UML
Facilité d'intégration des modèles dans le projet	Facile	Difficile

TABLE 2.3 – Comparaison des propriétés des deux langages de description de processus

Il n'y a pas de concepts prédéfinis dans UML ou dans SPEM pour décrire les incertitudes associées aux éléments de processus, mais ces deux langages fournissent la possibilité d'intégrer la représentation des incertitudes aux modèles de processus grâce aux mécanismes d'extension. Nous choisirons entre ces deux langages au chapitre suivant.

Après avoir vu les méthodes et les langages de modélisation des incertitudes dans la modélisation de processus, nous allons présenter dans la

suite les stratégies possibles et certains des outils qui peuvent être utilisés dans le traitement des incertitudes et de leurs effets.

2.3 STRATÉGIES DE TRAITEMENT DES INCERTITUDES ET DE LEURS EFFETS

Dans cette section, nous distinguons d'abord entre incertitudes et risques, et nous examinons les effets possibles des incertitudes sur lesquelles notre recherche se penche ; puis nous présentons le principe de traitement des incertitudes ; nous discutons ensuite les outils existants qui permettent de mettre en oeuvre ce principe dans des cas spécifiques, lorsqu'il s'agit d'analyser des risques ou de gérer des inconnues inconnues. Nous concluons sur la problématique que nous adoptons dans la suite de ce mémoire.

2.3.1 Risques, incertitudes et effets des incertitudes

Il est important de noter que risques et incertitudes ne sont pas synonymes.

L'incertitude peut produire un risque ou une opportunité ; le risque est toujours associé à des aspects négatifs, des dangers (?). La notion de risque est liée à deux concepts : la probabilité et la gravité des conséquences (INCOSE 2004) :

- Comment le risque peut-il se concrétiser ? (probabilité d'occurrence)
- Quels effets indésirables se produiraient-ils dans ce cas ? (gravité des conséquences)

Nous pouvons aussi distinguer les incertitudes et les risques de point de vue de l'"objectif" :

Hillson et al. (2004) indique qu'il faut comprendre **qu'une incertitude existe quel que soit le contexte, mais que le risque ne peut être défini que par rapport à des objectifs clairs**. Il donne ensuite une définition très

simple du risque serait : " une incertitude qui a de l'importance ", et cette incertitude a de l'importance si elle peut avoir un effet sur un ou plusieurs objectifs. Le risque ne peut pas exister dans un vide ; il nous faut définir ce qui est " à risque " - c'est-à-dire quels seraient les effets sur les objectifs si l'incertitude se concrétisait.

Une définition plus complète serait donc " **une incertitude qui, si elle arrivait, pourrait avoir un effet sur un ou plusieurs objectifs** "(Pich et al. 2002). Dans cette définition, nous reconnaissons qu'il y a peut-être bien d'autres incertitudes qui sont sans effet sur les objectifs et que nous ne devons donc pas prendre en compte dans le processus de gestion des risques. Il est par exemple probable, mais incertain, qu'il pleuve à un moment donné à Londres ; par rapport à un projet informatique aux Indes, cet évènement est sans intérêt, mais par rapport à un projet de garden-party pour la reine Elizabeth II, il faudrait certainement le prendre en compte. L'incertitude est identique dans les deux cas, mais pour l'un des projets elle est sans effet, et pour l'autre elle présente un risque considérable.

Une autre question est soulevée par la phrase " une incertitude qui pourrait avoir un effet sur un ou plusieurs objectifs " : quel est donc cet effet ? Il y a deux catégories d'effets : ceux qui vont rendre les objectifs plus difficiles à atteindre (connus sous le nom de risques) et ceux qui vont faciliter la réalisation d'une tâche ou l'atteinte de l'objectif (les opportunités). Le calcul des opportunités peut adopter la même logique que celui des risques : (probabilité d'un évènement) x (valeur de l'évènement).

Il est donc important d'établir un principe général de traitement des incertitudes et de leurs effets, et les méthodes et outils pour les analyser et les gérer. C'est ce que nous allons discuter maintenant.

2.3.2 Principe général de traitement des incertitudes et de leurs effets

Le problème du traitement des incertitudes peut s'organiser selon quatre axes : les incertitudes elles-mêmes, les effets des incertitudes, les réponses techniques et les résultats à évaluer. Ce cadre s'inspire de celui

proposé dans (McManus et Hastings 2005), qui est illustré dans la Figure 2.8.

Les incertitudes "**Uncertainties**" se rangent donc selon cinq catégories :

- "**Lack of knowledge**" signifie que des connaissances qui ne sont pas connues, ou ne sont connues que de façon imprécise, sont nécessaires pour compléter l'architecture du système de façon rationnelle. Ces connaissances peuvent tout simplement avoir besoin d'être collectées, ou elles doivent être créées. Elles peuvent même être inconnues, ou seulement connues à un moment quelconque dans l'avenir.



FIGURE 2.8 – Cadre de traitement des incertitudes et de leurs effets dans les systèmes complexes (McManus et Hastings 2005)

- "**Lack of definition**" signifie que des éléments du système n'ont pas été décidés ou bien spécifiés. Ces incertitudes sont souvent présentes au début d'un projet de développement d'un produit innovant.
- "**Statistically Characterized Variables**" signifie qu'une variable peut ne pas toujours être connue avec précision, mais peut être caractérisée statistiquement, au moins dans certaines limites. Une bonne mesure serait de connaître la distribution statistique des valeurs possibles; une mesure moins bonne serait de connaître au moins les limites des valeurs possibles. Ce type d'incertitude peut être traité par des techniques d'analyse puissantes. En effet, une grande partie de la science de l'analyse des risques est consacrée à

la caractérisation statistique des incertitudes. Nous allons voir dans la section suivante des méthodes qui rentrent dans ce cadre.

- "**Known Unknowns** " : désigne des choses que l'on sait ne pas connaître. Leurs valeurs sont au mieux encadrées, et peuvent être totalement inconnues. On les associe en général à des risques; par exemple, la fiabilité d'un équipement peut être caractérisée par son temps moyen de bon fonctionnement (MTBF).
- "**Unknown Unknowns** " : désigne des choses qui restent imprévisibles quelle que soit la quantité de données disponibles. Par définition, "on ne les connaît pas" et elles peuvent représenter un risque ou une opportunité; par exemple, il peut s'agir d'un changement du marché suite à l'apparition ou à la chute d'un compétiteur.

Parmi les risques et opportunités "**Risks / Opportunities**", on trouve par exemple :

- "**Failure**" : lorsque le système ne fonctionne pas.
- "**Emergent capabilities** " : lorsque le système fonctionne bien de façon inattendue, et / ou à des fins non prévues à l'origine.
- "**Market shifts (+/-)** " : lorsque le système fonctionne, mais que le besoin a changé. Cela peut être bon ou mauvais.

"**Mitigations**" indique des approches techniques de minimisation des risques, par exemple :

- "**Redundancy** " : lorsqu'on prévoit des copies multiples de sous-systèmes (ou de plusieurs copies d'ensembles de systèmes) pour s'assurer d'un résultat correct.
- "**Design Choices** " : lorsqu'on choisit des stratégies de conception, des technologies et / ou des sous-systèmes qui ne sont pas vulné-

rables à un risque connu.

" **Exploitations** " : indique des approches techniques d'amélioration de résultats attendus, par exemple :

- " **Trade Space Exploration** " : lorsqu'on analyse ou on simule plusieurs solutions envisageables dans de nombreuses conditions possibles. En général, l'utilisation de la simulation nécessite une puissance de calcul massive et donne un aperçu de la façon (favorable ou non) dont le système répond aux variations selon les choix de conception (McManus et Schuman 2003).
- " **Portfolios and Real Options** " : il s'agit de techniques en provenance du monde financier, qui permettent de simuler plusieurs options de conception en amont et d'améliorer les options d'une manière rationnelle quand plus d'informations sont disponibles, et / ou lors d'un changement des conditions du marché. Elles peuvent également être utiles pour la planification (Walton 2002).

" **Outcomes** " indique les résultats attendus qui sont souvent des attributs du système, en particulier ceux qui peuvent quantifier ou au moins qualifier leurs rapports avec les incertitudes. Par exemple,

- " **Reliability** " : c'est la probabilité que le système fonctionne correctement durant un temps déterminé. C'est l'objectif de la plupart des méthodes de gestion des risques actuellement utilisées (Villemeur 1998).
- " **Flexibility** " : c'est la souplesse du système, son aptitude à être modifié pour étendre ou enrichir ses fonctions ou services originaux (Villemeur 1998).

Le principe général de traitement des incertitudes est donc, simplement dit : *les incertitudes conduisent à des risques ou des opportunités, qui sont gérés techniquement par des approches de réduction de risques ou d'exploitation d'opportunités, qui espèrent aboutir à des résultats*

souhaités.

Notons que le cadre présenté ci-dessus a été proposé par McManus et Hastings (2005) dans le contexte de la conception de systèmes complexes, et que la typologie des incertitudes, les catégories des risques et opportunités, ainsi que les résultats ont été définis du point de vue de l'architecte ou du concepteur du système. Les réponses techniques proposées sont donc adaptées au traitement des incertitudes liées aux produits.

Le détail des catégories utilisées par McManus et Hastings (2005) ne s'applique donc pas forcément dans notre cadre de processus. Mais les principes qui sont sous-jacents à son analyse sont manifestement applicables dans notre cadre.

Il est donc nécessaire de savoir quelles sont les méthodes courantes qui permettront de mettre en oeuvre ces principes dans le cadre de processus de développement de systèmes. L'approche dépend de la prévisibilité des incertitudes, quelles que soient leurs sources ; nous allons voir séparément des méthodes pour l'analyse des "Inconnues connues ", considérées sous l'angle du risque, et des méthodes pour gérer les "Inconnues inconnues ".

2.3.3 Analyse des risques

Le management du risque est un processus de découverte des incertitudes qui pourraient constituer une menace pour un projet, d'évaluation des risques correspondants, et d'application des moyens de minimisation des risques pour garantir la fiabilité des produits et la robustesse des processus. Il intervient donc forcément dans nos études.

Toutefois, comme il est appliqué en général au début du projet, il est souvent de nature qualitative ou simplement basé sur des expériences antérieures. L'analyse des risques est donc habituellement utilisée pour évaluer les " inconnues connues " telles que le risque d'un accident nucléaire ou de la chute d'un avion sur une maison en caractérisant statistiquement les variables qui sont de type de risques "catastrophe". Il est

rarement utilisé pour cibler les opportunités apportées.

Les détails concernant chaque technique peut être retrouvée dans la plupart des ouvrages de la gestion de projets (Moder et Phillips 1995) (Steward 2001) (PMI 2000) (IEEE 2003).

2.3.3.1 Outils et techniques pour l'analyse qualitative des risques

L'analyse qualitative des risques est le processus d'évaluation de l'impact et la vraisemblance des risques identifiés. Ce processus privilégie les risques en fonction de leur effet potentiel sur les objectifs du projet. Il demande que la probabilité et les conséquences des risques soient évaluées en utilisant des méthodes et des outils établis. Nous allons illustrer ci-dessous les quatre moyens courants appliqués au cours de cycle de vie de projet.

1. *Evaluation de la vraisemblance des risques et impacts.*

La vraisemblance du risque est la probabilité qu'un risque se produise ; l'impact du risque est l'effet sur les objectifs du projet si le risque se produit (IEEE 2003).

La vraisemblance des risques et les impacts des risques peuvent être décrits en termes qualitatifs tels que : très forte, forte, modérée, faible et très faible.

L'analyse des risques en utilisant la vraisemblance et les conséquences aide à identifier les risques qui doivent être gérés proactivement (IEEE 2003).

2. *Matrice de probabilité de l'évaluation des risques.*

Une matrice peut être construite pour attribuer un niveau à des risques (très faible, faible, modéré, élevé et très élevé) sur la base de la combinaison de leur vraisemblance et de leur impact. La vraisemblance et l'impact d'un risque peuvent être mesurés par des nombres compris entre 0,0 (aucune probabilité) et 1,0 (certitude)(Grey 1995). Leur évaluation peut être difficile, car un jugement d'expert est

utilisé, souvent sans bénéficié de données historiques suffisantes.

Une échelle ordinale peut aussi être utilisée, en utilisant une échelle allant de "très peu de chances" à "à peu près certain", de même qu'une échelle cardinale (linéaire comme 0,1 / 0,3 / 0,5 / 0,7 / 0,9, ou non linéaire comme 0,05 / 0,1 / 0,2 / 0,4 / 0,8, selon les besoins).

Le choix de l'échelle se fait en général en fonction de la culture de l'organisation qui conduit l'analyse. L'utilisation d'une échelle qualitative ou non linéaire reflète souvent le désir de particulariser les risques dont la vraisemblance ou l'impact exigent des précautions particulières.

L'utilisation de ces échelles nécessite que les définitions qui sont à leur base soient établies avant le début du projet, bien connues et partagées par l'organisation. Ces définitions permettent d'améliorer la qualité des données et de rendre le processus plus reproductible (IEEE 2003).

Evaluating Impact of a Risk on Major Project Objectives (ordinal scale or cardinal, non-linear scale)					
Project Objective	Very Low .05	Low .1	Moderate .2	High .4	Very High .8
Cost	Insignificant Cost Increase	<5% Cost Increase	5-10% Cost Increase	10-20% Cost Increase	>20% Cost Increase
Schedule	Insignificant Schedule Slippage	Schedule Slippage <5%	Overall Project Slippage 5-10%	Overall Project Slippage 10-20%	Overall Project Schedule Slips >20%
Scope	Scope Decrease Barely Noticeable	Minor Areas of Scope Are Affected	Major Areas of Scope Are Affected	Scope Reduction Unacceptable to the Client	Project End Item Is Effectively Useless
Quality	Quality Degradation Barely Noticeable	Only Very Demanding Applications Are Affected	Quality Reduction Requires Client Approval	Quality Reduction Unacceptable to the Client	Project End Item Is Effectively Unusable

Chaque risque est évalué en fonction de sa probabilité d'occurrence et de son impact s'il se produit. Les risques sont répartis en faibles (gris foncé), moyens (gris clair) ou élevés (noir) comme indiqué dans le tableau.

FIGURE 2.9 – L'évaluation des impacts d'un risque (PMI 2000)

La Figure 2.9 est un exemple d'évaluation d'impacts des risques

selon l'objectif du projet. Il illustre son utilisation avec une approche ordinale ou cardinale.

Risk Score for a Specific Risk					
Probability	Risk Score = P × I				
0.9	0.05	0.09	0.18	0.36	0.72
0.7	0.04	0.07	0.14	0.28	0.56
0.5	0.03	0.05	0.10	0.20	0.40
0.3	0.02	0.03	0.06	0.12	0.24
0.1	0.01	0.01	0.02	0.04	0.08
	0.05	0.10	0.20	0.40	0.80
	Impact on an Objective (e.g., cost, time, or scope) (Ratio Scale)				

Les impacts sur les objectifs du projet peuvent être évalués sur une échelle allant de très faible à très haute ou sur une échelle numérique. L'échelle numérique (cardinale) présentée ici est non-linéaire, ce qui indique que l'organisation souhaite éviter des risques élevés et à très fort impact.

FIGURE 2.10 – Matrice de Vraisemblance-Impacts (NASA 2007)

La Figure 2.10 est une matrice de Vraisemblance-Impacts (V-I) qui illustre une façon de combiner ces deux dimensions. Elle multiplie simplement les valeurs attribuées aux estimations de la vraisemblance et de l'impact pour déterminer si un risque est considéré comme faible, modéré ou élevé. Le score (V*I) du risque permet de mettre le risque dans une catégorie qui doit guider les actions de réponse aux risques.

3. Tests d'hypothèses de projet.

Les hypothèses identifiées doivent être testées selon deux critères : la stabilité et la prise en charge des conséquences sur le projet. Les hypothèses alternatives qui peuvent être vraies doivent être identifiées et leurs conséquences sur les objectifs du projet doivent être testées dans le processus d'analyse qualitative des risques (INCOSE 2004).

4. *L'évaluation de la précision des données.*

L'analyse qualitative des risques nécessite des données exactes et impartiales pour être utile à la gestion de projet. L'évaluation de la précision des données est une technique permettant de mesurer le degré auquel les données sur les risques sont utiles pour la gestion des risques (INCOSE 2004). Il s'agit d'examiner :

- les données disponibles sur le risque
- la qualité des données
- la fiabilité et l'intégrité des données.

L'utilisation des données de précision faible, par exemple, si le risque n'est pas bien compris peut conduire à une analyse qualitative des risques peu utile pour le chef de projet.

2.3.3.2 **Outils et techniques pour l'analyse quantitative des risques**

L'analyse quantitative des risques suit généralement l'analyse qualitative des risques. Elle vise à analyser numériquement la probabilité de chaque risque et ses conséquences sur les objectifs du projet, ainsi que l'étendue de l'ensemble des risques du projet. Ce processus utilise des techniques telles que la simulation de Monte Carlo et analyse de la décision pour (PMI 2000) :

- déterminer la probabilité d'atteindre un objectif du projet ;
- quantifier le risque pour le projet, et déterminer le coût et les délais qui peuvent être nécessaires ;
- identifier les risques nécessitant le plus d'attention.

Quatre moyens d'analyse quantitative des risques sont présentés dans les paragraphes suivants.

1. *Interview.*

Les techniques d'interview sont utilisées pour quantifier la probabi-

lité et les conséquences des risques sur les objectifs du projet.

Une revue de risque avec les parties prenantes du projet et des experts sera la première étape dans la quantification des risques.

Les informations nécessaires dépendent du type de la distribution de probabilité qui sera utilisée. Par exemple, les informations devraient être rassemblées d'un point de vue optimiste, neutre et pessimiste, et les scénarios estimés en supposant une distribution triangulaire des probabilités, ou une distribution normale ou log-normale. Un exemple d'estimation des coûts est présenté dans la figure suivante 2.11 (PMI 2000).

Project Cost Estimates and Ranges			
WBS Element	Low	Most Likely	High
Design	4	6	10
Build	16	20	35
Test	11	15	23
Total Project		41	

L'interview de risque détermine l'estimation de coût de trois éléments d'un WBS (Work Breakdown Structure). L'estimation traditionnelle de \$ 41, qui est la somme des coûts les plus probables, est en fait relativement peu probable, comme le montre la troisième figure 2.13.

FIGURE 2.11 – Estimation de coûts obtenue par l'entrevue de risque

2. Analyse de sensibilité.

L'analyse de sensibilité permet de déterminer les risques qui ont le plus l'impact potentiel sur le projet. Elle examine dans quelle mesure l'incertitude de chaque élément de projet affecte l'objectif (Saltelli et al. 2000).

3. Analyse d'arbre de décision.

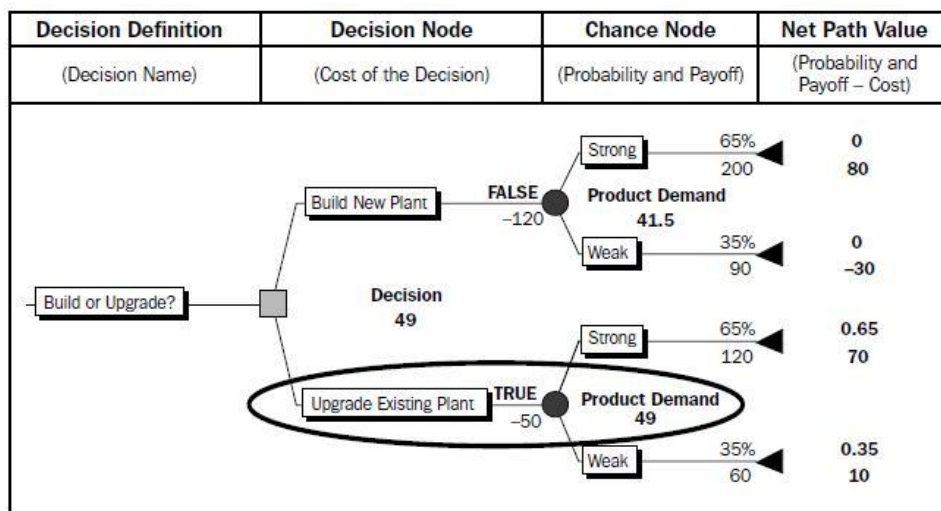
Une analyse de la décision est souvent structurée comme un arbre de décision. L'arbre de décision est un diagramme qui permet de

considérer les solutions alternatives pour arriver à une décision Il intègre les probabilités et les coûts des risques ou les avantages de tous les chemins pouvant être suivis selon les événements et les décisions futures.

La solution de l'arbre de décision indique la décision qui permet au décideur de maximiser une valeur, en supposant que toutes les implications des incertitudes, les coûts, les avantages, et les décisions ultérieures sont quantifiés. Un arbre de décision est montré dans la Figure 2.12 (Grey 1995).

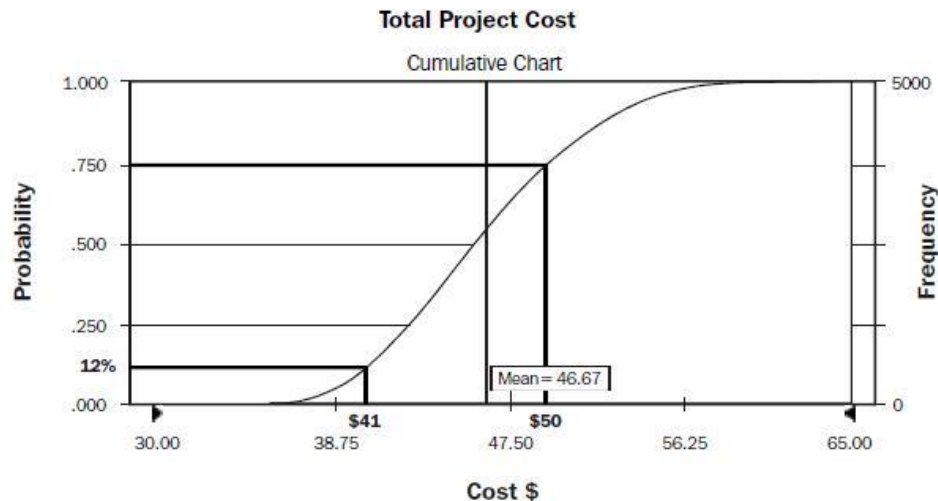
4. Simulation.

Une simulation de projet utilise un modèle qui relie les incertitudes spécifiées à leur impact potentiel sur les objectifs du projet. Ce type de simulation est généralement réalisé en utilisant la technique de Monte Carlo. Un exemple d'un résultat de la simulation du coût d'un risque est montré dans la Figure 2.13 (PMI 2000).



Vaut-il mieux construire une nouvelle usine ou moderniser l'ancienne ? Avec les valeurs données pour les probabilités et les avantages des différentes hypothèses alternatives, l'espérance de gain est de $0,65 \times 80 + 0,35 \times (-30) = 41,50$ dans le premier cas, et de $0,65 \times 70 + 0,35 \times 10 = 49$ dans le second cas.

FIGURE 2.12 – Exemple d'analyse d'arbre de décision



Coût probable cumulé du WBS de la Figure 2.11. Estimation de coûts obtenue par l'entrevue de risque en supposant des distributions triangulaires pour les données. La simulation montre que le projet possède seulement 12% de chances de coûter 41 unités ou moins. Une estimation conservatrice, sûre à 75%, est un budget de 50 unités.

FIGURE 2.13 – Exemple de la simulation de coût d'un risque

2.3.3.3 Limites et conclusions

Au regard de la bibliographie, nous avons noté que les approches présentées, dans cette section, pour l'évaluation des risques, plus précisément pour l'analyse (qualitative et quantitative) des risques, reposent sur une hypothèse fondamentale, à savoir que nous travaillons essentiellement sur le "terrain du savoir", où les événements, l'espace des solutions et les résultats des actions sont en principe connus, où la complexité reste modérée, où une action n'a pas d'effet inattendu dans une autre partie du projet, et où l'on peut choisir entre différents plans d'action.

En d'autres termes, nous connaissons la gamme de risques qui peuvent se produire et leurs causes, même si nous ne pouvons pas être en mesure de les prédire avec certitude ou d'avoir de bonnes estimations de leurs probabilités.

Cependant, tous les projets ne répondent pas à ces hypothèses. Au contraire, les projets qui sont nouveaux en termes de technologie, et / ou les projets de longue durée, sont souvent perturbés par des événements imprévisibles, et / ou des interactions inconnues entre différentes actions

et des parties de projet. Ils comportent donc des incertitudes de type " Inconnues inconnues ". Nous allons examiner dans la section suivante comment on peut réagir face à ce type d'incertitudes.

2.3.4 Gestion des inconnues inconnues

Comme nous venons de le voir à propos des limites de la gestion des risques, la gestion des inconnues inconnues est nécessaire dans le processus de développement de systèmes complexes.

Selon les observations de Smith et al. (2004), qui sont aujourd'hui largement reconnues, l'apprentissage et la sélection sont les deux principales approches envisageables pour la gestion de ces inconnues dans le contexte des processus de DS. Nous allons maintenant présenter les principes de ces approches.

2.3.4.1 Principes de l'approche par apprentissage

Cette approche a reçu des noms différents selon les chercheurs en gestion de projets. Par exemple, Chew et al. (1991) ont examiné les inconnues inconnues dans le contexte de l'introduction des nouvelles technologies de fabrication de produits et en ont conclu à l'existence de quatre étapes : l'itération, l'apprentissage, la définition des nouveaux problèmes à résoudre, et l'ajustement. Dans le cadre du développement de nouveaux produits, Leonard-Barton (1995) appelle cette approche "Product morphing" (au sens où des modifications répétées sont effectuées pendant la phase de la conception de produit). Le nom le plus récent, donné par Loch et al. (2005), est "Learning". L'itération et l'expérimentation sont un élément fondamental de la résolution de problèmes en matière d'innovation et de l'ingénierie de projet.

Les typologies classiques de l'approche d'apprentissage organisationnel distinguent deux niveaux d'apprentissage, comme le montre dans la Figure 2.14 : (1) " single loop ", et (2) " double loop ". Argyris et Schön (1978) font la distinction entre simple et double boucle d'apprentissage, en

lien avec les concepts de premier et de deuxième ordre d'apprentissage. Lors d'un apprentissage à boucle simple, les entités (individus, groupes ou organisations) modifient leurs actions en fonction de la différence entre les résultats attendus et obtenus. Ce type d'apprentissage met l'accent sur la détection et la correction des erreurs sur un ensemble de variables liées à des changements dans les organisations.

En apprentissage à double boucle, les entités se posent des questions sur les valeurs, les hypothèses et les politiques qui ont conduit à des actions; si elles sont en mesure de visualiser et de modifier celles-ci, un apprentissage de deuxième ordre, ou de double boucle, a lieu pour remettre en cause et optimiser les stratégies.

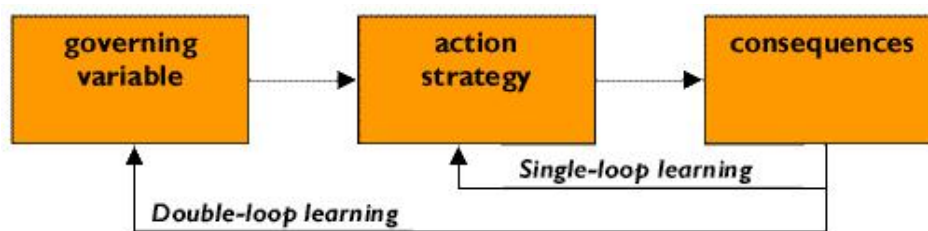


FIGURE 2.14 – Modèle d'apprentissage (Smith et al. 2004)

Comment les organisations peuvent-elles réaliser un apprentissage à double boucle? En d'autres mots, comment vont-elles opérer par rapport à l'acquisition de nouvelles informations et comment vont-elles réagir à cette information? Il existe deux types d'apprentissage à double boucle : l'apprentissage improvisé (l'apprentissage en temps réel des changements de l'action, peu adapté aux projets complexes) et l'apprentissage expérimental (qui s'appuie sur des mécanismes d'essais et erreurs). Nous nous intéresserons seulement à l'apprentissage expérimental.

Le principe de base de l'apprentissage expérimental est le cycle PDCA, représenté sur la figure 2.15.

Ce cycle matérialise une méthode d'apprentissage et de conduite de l'amélioration d'un produit ou d'un processus. Il comporte quatre étapes

dont les noms peuvent être traduits par : Préparer, Développer, Comprendre, Agir (Thomke et Reinertsen 1998).

1. Préparer. Cette étape, la plus importante, consiste à préparer un changement ou une amélioration en établissant les objectifs et les processus nécessaires pour fournir des résultats conformément aux sorties attendues. Lorsque quelqu'un a une idée pour améliorer un produit ou un processus, il faut préparer un essai comparatif ou une expérience. Le cycle complet repose sur cette première étape, bien qu'elle soit souvent abrégée dans la pratique parce que l'on est pressé de passer à l'action, peut-être pour ne pas paraître inactif.
2. Développer. Cette étape consiste à réaliser l'essai comparatif ou l'expérience, de préférence à petite échelle, en respectant bien les dispositions définies à la première étape.
3. Comprendre. Il s'agit d'étudier les résultats. Qu'avons-nous appris ? Les résultats répondent-ils à nos attentes ? Dans le cas contraire, pourquoi ?
4. Agir. Trois cas peuvent se présenter. Soit on décide d'adopter le changement, ou de généraliser l'expérience. Soit on décide d'abandonner l'étude, donc de continuer à travailler dans les mêmes conditions. Soit on décide de recommencer le cycle en changeant certaines conditions initiales.

Tout comme un cercle sans fin, le cycle PDCA doit être répété encore et encore pour l'amélioration continue. Loch et al. (2005) indique que le facteur clé de succès pour l'apprentissage est de maintenir le cycle PDCA petit et rapide. Autrement dit, les "échecs" devraient venir tôt et souvent, avant qu'ils ne deviennent catastrophiques ; les changements au début du projet sont toujours moins coûteux que ceux qui arrivent plus tard dans le projet.

L'utilisation du cycle PDCA est une méthode qui peut obtenir d'excellents résultats à condition d'accorder à la phase de préparation toute



FIGURE 2.15 – Cycle Plan-Do-Check-Act (PDCA)

l'importance nécessaire (De Rocquigny et al. 2008).

2.3.4.2 Principes de l'approche par sélection

Cette approche, dite aussi par "essais parallèles", a été identifiée (en traitant des problèmes pratiques très complexes) avant que la recherche se penche sur les problèmes d'ingénierie. Leonard-Barton (1995), qui prend pour exemple l'introduction simultanée de modèles concurrents d'automobiles sur les marchés, a parlé à ce sujet de "sélection darwinienne". Pich et al. (2002) reprennent le qualificatif de "selectionist", parce que la caractéristique fondamentale de cette approche est qu'un seul des nombreux scénarios est retenu (que les essais aient été exécutés en parallèle ou les uns après les autres).

L'idée d'essayer plusieurs solutions rappelle l'envoi de plusieurs équipes à la recherche d'une cité inconnue dans la jungle : il s'agit d'explorer un espace de solutions sur les paramètres de conception (d'un processus, un produit ou un projet) pour trouver celui qui a la meilleure performance. Le problème est souvent de trouver au moins une solution avec une performance suffisamment "bonne". Inversement, il peut y avoir trop de "bonnes" solutions, ce qui rend l'optimisation difficile, du moins par l'analyse et la recherche incrémentale.

En définitive, l'approche par sélection repose sur l'équilibre entre les aspirations individuelles (pour les équipes, qui prennent des risques lors

des essais en vue de réussir) et l'objectif du groupe (trouver la meilleure solution globale pour l'organisation)(Sommer et al. 2009).

Sobek II (Sobek II et al. 1999) décrit par exemple comment un constructeur japonais fabrique de nombreux prototypes d'une nouvelle voiture : un large ensemble de solutions possibles à un problème de conception est examiné, et on réduit progressivement l'ensemble des possibilités pour converger sur une solution définitive. Dès le début, il y a un large éventail de possibilités, suivies par des développements et des tests, et confrontées aux exigences clientèles et aux performances des autres prototypes. Progressivement, les idées convergentes, et les différents participants s'engagent à rester au sein d'une enveloppe de conception fixe.

Selon l'auteur, cette forme de sélection, conjuguée à un affinage progressif de l'espace des solutions, permet de trouver la meilleure solution avec une grande probabilité. La définition initiale l'espace des solutions est coûteuse en temps, mais une fois les différents essais effectués, la convergence et le passage à la production sont plus rapides que lorsqu'on utilise des méthodes d'ingénierie plus traditionnelles.

Trois conditions de succès doivent être assurées pour que cette méthode fonctionne :

1. La définition de l'espace des solutions acceptables, c'est-à-dire des contraintes et des critères qui caractérisent cet ensemble acceptable.
2. L'intégration des solutions qui se sont développées en parallèle par la recherche de l'intersection des ensembles réalistes. Dans cet exercice, il est extrêmement important de rechercher une solution robuste, c'est-à-dire de sélectionner les solutions qui peuvent être appliquées avec succès dans de nombreuses combinaisons de contraintes différentes.
3. La validation des solutions possibles avant d'aller plus loin.

Dans la pratique, on peut exprimer les principes de la sélection par

trois questions fondamentales :

1. Dans quel espace allons-nous former des alternatives ? Quel est de l'espace des solutions réalisables ?
2. Combien d'options, au total, ou d'expériences peut-on se permettre de gérer simultanément ?
3. Quand est-ce que nous nous arrêtons de développer des options ?

Autour de ces questions, de nombreux outils sont apparus. Par exemple, l'application de la théorie de l'utilité (Thurston 1990) est l'un des plus puissants outils mathématiques pour l'évaluation et la comparaison des alternatives dans le cadre de l'aide à la décision en présence d'alternatives complexes dans le projet.

2.3.4.3 Conclusions

La Figure 2.16 résume la logique des trois approches fondamentales en face des différents types d'"Inconnues" dans le contexte de la gestion de projet.

La première approche (notée a dans la figure) décrit le Management du Risque évoqué dans la section précédente. Avec une information adéquate sur l'impact des variables, les effets des actions, et le calcul des performances, nous pouvons choisir un plan de projet et escompter un résultat. Nous pouvons éventuellement re-planifier le projet, si de nouvelles informations sur un risque majeur apparaissent, et "improviser" autour de l'objectif en vue de répondre aux risques résiduels. Mais fondamentalement, cette approche est une approche planifiée, qui ne convient pas à la gestion des incertitudes imprévues.

Les approches notées b et c montrent deux approches fondamentales que nous pouvons utiliser pour répondre à la présence d'incertitudes de type "Inconnues inconnues".

Si nous admettons que nous en savons trop peu sur l'univers des résultats possibles du projet (et comment y arriver), nous ne nous focaliserons pas sur le choix d'un seul objectif dès le départ. Nous ne choisirons pas non plus un seul critère d'évaluation, puisque l'objectif précis n'est pas connu.

Dans l'approche par apprentissage (notée b sur la figure), nous commencerons par aller vers un résultat (le meilleur que nous pouvons identifier à ce stade), en étant prêts à reprendre le projet et/ou à effectuer des changements fondamentaux au moment où de nouvelles informations deviennent disponibles. En d'autres termes, nous répétons le cycle (examen de l'état du projet, acquisition des nouvelles informations, etc.) pendant toute la durée du projet.

Dans l'approche sélectionniste (notée c sur la figure), l'espace des solutions est a priori inconnu, ce qui n'était pas le cas dans les deux approches précédentes. La solution est alors de lancer plusieurs essais et de voir lequel aboutit aux meilleurs résultats. Le cas échéant, cette démarche pourra être itérée jusqu'à arriver à un résultat considéré comme définitif.

En général, l'application de ces trois approches est coûteuse. Dans l'évaluation des risques et l'approche par apprentissage, les itérations répétées peuvent prendre un temps considérable ; dans l'approche sélectionniste, c'est l'exécution parallèle de plusieurs essais qui est coûteuse.

Pour revenir à notre problème, nous pensons que les deux approches les plus prometteuses sont les approches par sélection et par apprentissage, lorsqu'on a affaire à des incertitudes appartenant à la catégorie des inconnues inconnues.

2.4 NOTRE PROBLÉMATIQUE

La gestion des incertitudes est un exercice difficile et pourtant essentiel pour le développement de systèmes complexes à long-terme. De nom-

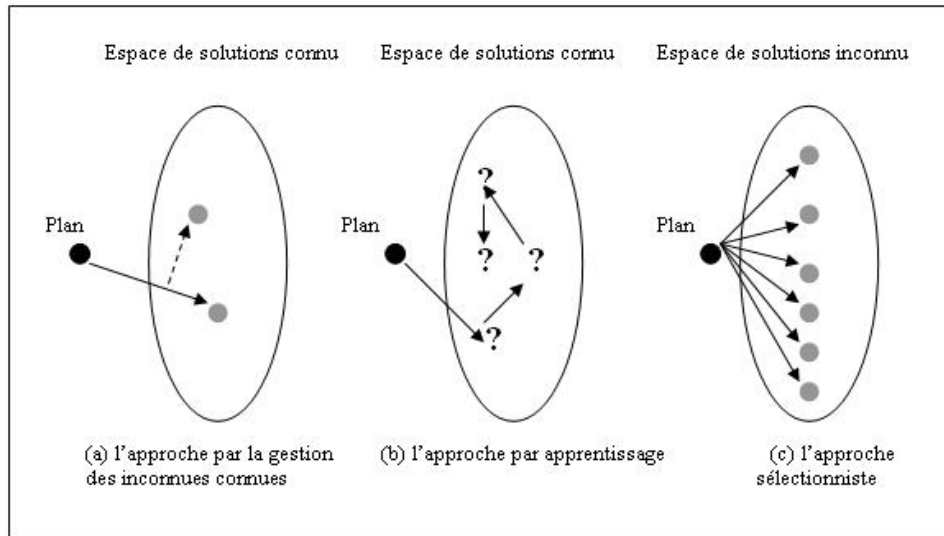


FIGURE 2.16 – Trois approches fondamentales à la gestion des inconnues

breuses études rétrospectives montrent que les incertitudes constituent souvent une considération majeure dans la performance et la qualité de l'ingénierie des systèmes. L'incertitude doit donc être traitée comme un élément essentiel pour la conception et le développement de systèmes complexes.

Pour cela, l'idéal serait de disposer d'un cadre systématique de gestion des incertitudes dans la conception de produit et la conception de projet, qui respecterait les principes de traitement des incertitudes évoqués ci-dessus en recueillant les connaissances de toutes les incertitudes liées aux processus de DS complexes, en calculant tous les risques et toutes les opportunités potentielles découlant de ces incertitudes, en modélisant tous les effets de réductions de risques et d'exploitations d'opportunités, et en évaluant tous les résultats souhaitables.

Les difficultés sur cette voie sont nombreuses et diverses : il faut cogérer des incertitudes du point de vue du produit (matériaux divers, fonctions multiples et interactives, disciplines différentes, etc.), et du point de vue du projet (ensemble de processus de cycle de vie, de métiers divers, etc.)... Il faut penser à l'évolution des incertitudes et à la réutilisation des modèles. Il faut être sûr de faire des choix techniques de gestion des incertitudes pouvant être facilement implémentés dans la pratique et de définir une méthodologie de gestion des incertitudes pouvant être faci-

lement intégrée aux autres méthodologies de l'entreprise. Il faut penser également au coût et au temps nécessaire à la réalisation des activités de gestion des incertitudes.

Dans tous les cas, nous pensons que deux étapes préalables sont nécessaires pour répondre à ces problématiques :

1. Comme l'utilisation de modèles est à la base de l'ingénierie, la représentation des informations imparfaites dans les modèles de processus est indispensable afin de fournir une base solide pour évaluer l'incertitude et suivre son évolution dans un projet.
2. Les processus de gestion des incertitudes doivent être définis et intégrés dans la gestion des projets. Ceci permettra aux activités de gestion des incertitudes de traverser tout le cycle de vie du produit et/ou du projet.

Dans cette thèse, nous voulons montrer que la représentation explicite des incertitudes dans les modèles de processus peut faciliter leur compréhension et être un support de description de processus complexes, et qu'une méthodologie de gestion des incertitudes peut répondre aux questions pratiques de réalisation des activités de gestion des incertitudes.

Cette problématique répond à deux types d'enjeux :

- D'une part, à des **enjeux industriels** liés à la maîtrise et à la mise en oeuvre d'une gestion des incertitudes qui conduise à mieux maîtriser les objectifs des projets (coût, délais, qualité...).
- D'autre part, à des **enjeux académiques** autour de l'Ingénierie Système. Cela explique l'engagement, depuis plusieurs années, du laboratoire LATTIS (Laboratoire Toulousain de Technologie et d'Ingénierie des Systèmes) et du LGP de Tarbes (Laboratoire de Génie de

Production) dans l'approfondissement d'une démarche de conception étendue de projet.

2.5 CONCLUSIONS DU CHAPITRE

Dans ce chapitre, nous nous sommes attachés à définir les concepts principaux pour la modélisation de processus de développement de systèmes complexes et à préciser la nature, le statut, et la prévisibilité des incertitudes dans ce cadre.

Les sections 2.2 et 2.3 ont passé en revue plusieurs des méthodes utilisées pour aborder les problèmes de la modélisation, évaluation et gestion des incertitudes⁴.

Nous avons notamment retenu que :

1. Les méthodes existantes de modélisation des incertitudes dans la modélisation de processus s'adressent souvent aux incertitudes liées aux activités, mais ignorent les existences des incertitudes liées aux autres éléments de processus, comme les livrables, les humains, etc.
2. Les modèles de processus actuels sont plutôt adaptés à l'analyse des "inconnues connues" qui représentent seulement une partie des incertitudes.
3. Les méta-modèles UML ou SPEM ne comportent pas de moyens d'expression standard pour décrire l'incertitude dans les processus, mais ces deux langages fournissent un moyen pour pallier cette absence grâce aux mécanismes d'extension.
4. Les incertitudes ne sont pas toujours synonymes de risques, et les méthodologies de gestion des risques ne sont pas suffisantes pour gérer toutes les incertitudes dans la gestion de projet.

4. Nous n'avons pas cru utile de présenter des théories fondamentales comme la logique floue, la théorie des probabilités, etc., sur lesquelles sont basées la plupart des méthodes introduites.

A partir de ce que nous venons de retenir, nous avons dégagé deux problèmes concrets qui feront l'objet du chapitre 3 de la thèse :

- Représenter des informations imparfaites dans les modèles de processus.
- Intégrer des processus de gestion des incertitudes dans la gestion de projet.

MÉTHODOLOGIE DE GESTION ET MÉTA-MODÉLISATION DES INCERTITUDES

Le chapitre précédent a défini les concepts principaux à propos de la modélisation des incertitudes associées aux processus de développement de systèmes, et a évoqué certaines des stratégies de traitement envisageables dans ce cadre.

Il a également souligné l'intérêt, voire la nécessité, de s'appuyer pour cela sur des formalismes adaptés aux problèmes de la modélisation, de l'évaluation et de la gestion des incertitudes.

Le présent chapitre va chercher à répondre de façon plus précise au problème de la prise en compte et de la gestion des incertitudes dans le développement de systèmes, en se concentrant sur trois questions majeures :

- *Quelle méthodologie suivre pour gérer les incertitudes qui peuvent se présenter lors du développement d'un système complexe ?*
- *Comment catégoriser ces incertitudes ?*
- *Comment les représenter, en lien avec la modélisation des processus ?*

Les première et deuxième sections de ce chapitre vont tâcher de répondre à la première question, en définissant d'abord une méthodologie

de gestion des incertitudes précisant l'ensemble des activités appropriées pour concrétiser cette méthodologie, ainsi que leurs entrées et leurs sorties, puis en examinant comment intégrer ces activités dans les différentes phases d'un projet.

La troisième section reviendra sur les incertitudes qui peuvent se présenter concrètement dans le domaine que nous étudions, et cherchera à les ordonner et à les catégoriser : c'est un préalable nécessaire à la quatrième section, qui sera dédiée à la méta-modélisation de nos incertitudes, et à la construction d'un nouveau profil UML pour leur représentation.

Nous discuterons enfin, en conclusion, les avantages et les limites des solutions que nous proposons.

3.1 MÉTHODOLOGIE DE GESTION DES INCERTITUDES

Dans cette section, nous présenterons d'abord une vue d'ensemble de notre méthodologie et des étapes principales de son déroulement ; nous préciserons ses objectifs, et indiquerons sous quelle hypothèse elle peut être considérée comme valable, ainsi que les niveaux de décomposition de système et les moments possibles pour son application ; ensuite nous définirons plus en détail les constituants de cette méthodologie.

Plus précisément, nous allons définir l'objectif, les éléments en entrée, les activités principales recommandées, certains outils applicables, et les éléments en sortie de chaque étape de la méthodologie.

3.1.1 Vue d'ensemble de la méthodologie

La méthodologie de gestion des incertitudes que nous proposons combine quatre étapes séquentielles avec une activité en parallèle, comme le montre la Figure 3.1.

- Etape 1 : identification des incertitudes
- Etape 2 : analyse de leurs effets

- Etape 3 : réduction et /ou exploitation des incertitudes
- Etape 4 : validation de solution et contrôle des incertitudes
- Activité en parallèle : représentation des incertitudes

Les objectifs de cette méthodologie sont les suivants :

- (Comprendre) : comprendre (ou prévoir) l'influence ou l'importance des incertitudes au cours de développement, et donc guider les efforts des ingénieurs ;
- (Démontrer) : démontrer la conformité du système avec des critères explicites ou des seuils réglementaires (par exemple des normes, une licence Safety Integrity Level, ou la certification du système pour une utilisation critique) ;
- (Gérer) : gérer les données concernant les incertitudes de façon cohérente ;
- (Sélectionner) : comparer les performances des différentes solutions pour le meilleur choix.

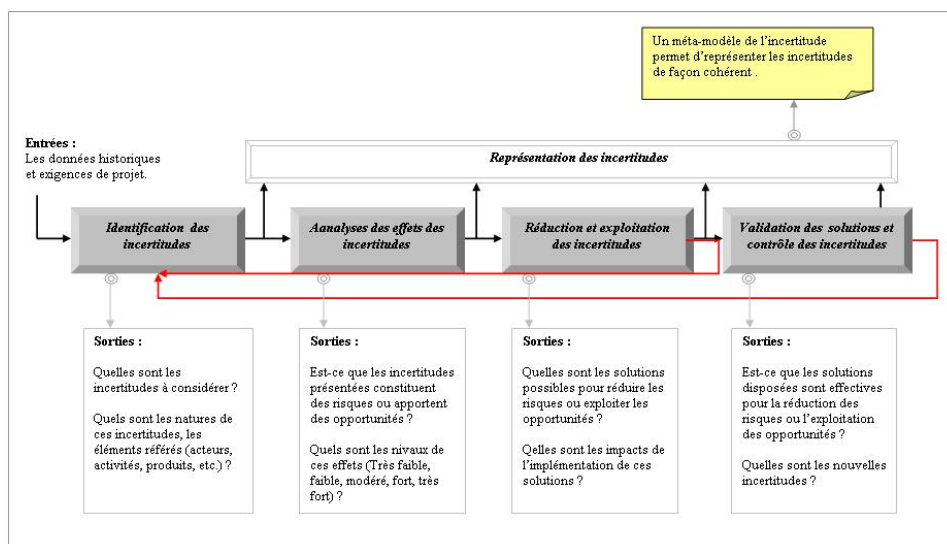


FIGURE 3.1 – La procédure de la gestion des incertitudes

L'hypothèse de l'application de cette méthodologie réside dans le fait que l'organisation possède des données suffisantes pour que l'identification des incertitudes soit raisonnablement possible. La plupart du

temps, il s'agira de données historiques sur un système existant dont le fonctionnement est similaire à celui qui va être développé.

Ces données historiques peuvent notamment concerner la définition du système existant et le processus de son développement. Elles peuvent provenir des documents techniques et méthodologiques relatifs aux anciens projets de l'organisation, par exemple, des documents du cycle de vie de développement du système relatifs à sa définition, sa conception, sa réalisation, sa validation et sa vérification, ou à l'exploitation et à la mise en service du système.

Cette méthodologie peut s'appliquer à chaque niveau de décomposition du système, les réponses aux incertitudes données pour un niveau de décomposition (n) devenant les référentiels pour le niveau inférieur (n-1) et ne devant plus être remises en cause, sauf exception. L'application de la méthodologie peut ou doit donc se faire successivement pour les différents niveaux de décomposition du système : système, sous-système, composant, unité, etc. En fonction des besoins, des ressources et des habitudes de l'entreprise, elle pourra être mise en oeuvre en totalité ou seulement en partie.

Ceci dit, le meilleur moment pour commencer à dérouler cette méthodologie est lors d'une phase de définition suffisamment précoce (par exemple, la phase d'étude de faisabilité) ; des améliorations sont en effet encore possibles sans remettre en question les éléments déjà définis.

Nous allons maintenant décrire les cinq constituants de la méthodologie en détail dans la suite.

3.1.2 Etape 1 : identification et sélection des incertitudes

3.1.2.1 Objectifs

L'objectif de cette étape est d'identifier les incertitudes qui se présentent (ou se présenteront) au cours de développement du système, en fonction des données de base (en général historiques) et de l'analyse de la

conformité du système aux exigences des clients et/ou aux prescriptions réglementaires et normatives à respecter.

3.1.2.2 Entrées / Activités recommandées / Outils / Sorties

Entrées

Les éléments d'entrées de l'étape d'identification se composent :

1. des exigences des clients et des normes à respecter ;
2. des données de base (historique, etc.) d'un système existant ;
3. des résultats obtenus à partir des études sur les incertitudes déjà réalisées pour les itérations précédentes.

Note : les deux premiers éléments sont indispensables pour démarrer la méthodologie de gestion ; le troisième élément est nécessaire pour réaliser une nouvelle itération de la méthodologie.

Activités recommandées

L'ensemble des activités principales que nous définissons pour l'étape d'identification. Il s'agit de :

1. recueillir les données de base relatives à un système existant proche du système en cours de développement et au processus de son développement ;
2. identifier les incertitudes liées aux éléments du processus ;
3. sélectionner et la hiérarchiser les incertitudes ;

Au bout de cette étape, nous aurons donc identifié les incertitudes considérées comme importantes ; il s'agira en général d'incertitudes sur les activités, sur les livrables et sur le comportement de processus identifiées à partir des données historiques. La nature, la prévisibilité, l'importance et le point d'impact de ces incertitudes sont des éléments qui

pourront les caractériser.

On aura aussi sélectionné les incertitudes qui doivent être traitées en priorité, compte tenu de leurs effets sur l'objectif du projet, ainsi bien sûr que des moyens alloués à la gestion des incertitudes.

En général, les analyses menées auront été de type qualitatif pour chaque incertitude susceptible d'avoir une influence sur l'objectif. Elles seront souvent guidées par l'analyse de la conformité du système existant aux exigences et normes à respecter dans le projet, et écarteront certaines des incertitudes qui n'ont pas beaucoup d'intérêt selon l'expérience des ingénieurs.

Enfin, il sera en général nécessaire de hiérarchiser les incertitudes en leur attribuant une priorité, pour arriver à mieux organiser les activités de gestion. Cette hiérarchisation peut être par exemple déterminée en fonction des délais alloués aux ingénieurs et aux managers pour le traitement des incertitudes.

Certaines des incertitudes concernent en effet des préoccupations immédiates, par exemple, la façon dont on va maintenir les équipements d'un système dans un temps proche (ex : la semaine prochaine). Il s'agira par exemple d'incertitudes sur le temps moyen de réparation, les horaires précis, la répartition des ressources et la logistique, etc.

D'autres incertitudes concernent des préoccupations à plus long terme. Evoquons par exemple les incertitudes sur l'opportunité de l'évolution d'un système existant par l'amélioration des moyens de détection des fautes (ex : de redondance passive à redondance active). Dans ce cas, les incertitudes porteront sur la durée de la conception, la mise en exploitation, la maintenance, etc.

Une stratégie raisonnable pourrait par exemple attribuer une priorité d'autant plus haute à une incertitude que le problème correspondant doit être réglé à plus brève échéance.

Outils méthodologiques

1. *Interviews et revues*

Les techniques d'interview peuvent être utilisées, par exemple pour déterminer la priorité des incertitudes. Une revue des incertitudes avec les parties prenantes du projet et des experts sera souvent la première étape dans l'identification et la hiérarchisation des incertitudes.

2. *Audits*

Les techniques d'audit peuvent également être utilisées, par exemple pour déterminer si les résultats du projet sont conformes aux exigences. Ils peuvent être menés à tous les niveaux (par exemple, les sorties d'une seule activité, les livrables d'un processus ou le produit final du projet).

Les audits et/ou les interviews doivent être réalisés par plusieurs analystes possédant des bonnes connaissances du type de système qui va être développé. Une contrainte stricte sur ces analystes est qu'ils ne doivent pas être les mêmes personnes que celles qui ont participé au développement du système existant, conformément aux normes de sûreté destinées à minimiser l'impact des fautes humaines.

Sorties

Les éléments de sortie de cette étape sont les incertitudes retenues après sélection, avec leur priorité.

3.1.3 Etape 2 : analyse des effets des incertitudes

3.1.3.1 Objectifs

L'objectif de cette étape est d'analyser les effets des incertitudes retenues par l'étape précédente.

3.1.3.2 Entrées / Activités recommandées / Outils / Sorties

Entrées

Les éléments d'entrée de cette étape sont les éléments de sortie de l'étape d'identification.

Activités recommandées

L'ensemble des activités principales que nous définissons pour l'étape d'identification. Il s'agit de :

1. l'identification des effets des incertitudes ;
2. l'analyse de ces effets.

La première de ces activités consiste à identifier les effets des incertitudes sur le système et le projet. Comme nous l'avons discuté dans la section 3.1 du chapitre 2, il y a deux catégories d'effets : ceux qui vont rendre les objectifs plus difficiles à atteindre (connus sous le nom de risques) et ceux qui vont faciliter la réalisation d'une tâche ou l'atteinte de l'objectif (les opportunités). Pour chaque effet d'une incertitude, on doit en particulier identifier :

- la ou les phases du projet dans lesquelles elle se trouve,
- l'événement ou l'opération associé(e) à cet effet,
- le ou les systèmes, sous-systèmes ou composants affectés par cet effet,
- le ou les facteurs causant cet effet,

- la ou les conséquences de cet effet.

La seconde activité permet d'analyser, en termes qualitatifs ou quantitatifs, les effets identifiés au cours de la première activité, et donc estimer plus précisément la probabilité de survenance de l'effet et les conséquences de celui-ci : risques et conséquences de pannes éventuelles, s'il s'agit d'un effet négatif ; opportunités et valeur ajoutée, s'il s'agit d'un effet positif qui facilite l'atteinte d'un objectif du projet, l'amélioration de la qualité du produit, la baisse d'un coût, un progrès écologique, etc.

Remarquons qu'il peut arriver que cette deuxième activité se fasse en même temps que la première, ou soit omise, selon les caractéristiques de l'incertitude et les objectifs visés.

Outils

Tous les outils d'analyse de risque, qualitatifs ou quantitatifs, peuvent se révéler utiles dans ce cadre (cf. Chapitre 2, §2.3.3).

Sorties

Les éléments de sortie sont les effets des incertitudes (probabilité, gravité du risque ou valeur ajoutée, etc.).

En fonction des résultats obtenus lors de cette deuxième étape, la hiérarchisation des incertitudes résultant de la première étape peut éventuellement être revue (par exemple si on découvre, après une analyse détaillée, qu'un risque est plus grave que ce qu'une première impression laissait supposer)

3.1.4 Etape 3 : réduction et exploitation des incertitudes

3.1.4.1 Objectifs

L'objectif de cette étape est de réduire les incertitudes qui constituent des risques et/ou d'exploiter les incertitudes qui apportent des opportu-

nités.

3.1.4.2 Entrées / Activités recommandées / Outils / Sorties

Entrées

Les éléments d'entrée de cette étape sont les éléments de sortie de l'étape précédente.

Activités recommandées

Les activités principales permettant de réduire ou d'exploiter des incertitudes incluent :

1. l'identification des solutions envisageables pour réduire un risque et/ou exploiter une opportunité ;
2. l'identification des critères de sélection de solutions ;
3. la détermination de la solution choisie.

1. L'identification des solutions envisageables pour réduire un risque et/ou exploiter une opportunité est la première chose à faire. On peut distinguer deux grandes catégories de réponse : améliorer la robustesse du processus pour le renforcer contre un événement incertain ; améliorer sa flexibilité afin qu'il puisse s'adapter à un événement incertain.

Pour le premier type de réponse, il s'agit de traiter la réponse pour que le processus reste opérationnel et efficace dans des circonstances différentes. Par exemple, prévoir une solution de repli en cas de sinistre (locaux, matériels, ...) est un moyen d'améliorer la robustesse d'un processus de conception ou de fabrication.

Pour le deuxième type de réponse, l'objectif est d'exploiter les incertitudes afin de permettre au processus de profiter de changements imprévus. Par exemple, on peut concevoir un produit configurable

selon les besoins des clients, qui permettra de mieux s'adapter aux désirs des différents clients.

2. Les critères de sélection dépendent des exigences des clients. Ces critères sont souvent basés sur des paramètres quantifiables (coût, délai, qualité,...).

Note : on doit finir les deux premières activités pour pouvoir commencer l'activité (3).

3. Cette activité consiste à choisir une solution parmi les candidats identifiés par la première activité, en prenant en compte les critères de sélection issus de la deuxième activité.

Par exemple, lorsqu'on fait un choix entre deux scénarios pour l'architecture d'un système, on doit prendre en compte la robustesse, la disponibilité, le coût, etc. Cette activité comprend implicitement une étape d'évaluation de scénarios qui peut par exemple être guidée par une étude de type "Trade-off" sur le problème de l'éventail de scénarios (Ullman et Spiegel 2006).

Outils

Tous les outils classiques d'aide à la décision utilisés dans le cadre de l'Ingénierie des Systèmes peuvent se révéler utiles, comme par exemple :

1. *L'analyse d'utilité multi-attributs*

Cette méthode consiste à estimer la préférence des parties prenantes sur les différents attributs du système envisagé (ou du service attendu) (De Neufville 1990).

2. *L'arbre de décision*

Cette méthode consiste à expliciter graphiquement la séquence des décisions à prendre et les divers événements qui peuvent arriver (Ullman et Spiegel 2006).

Sorties

1. les solutions envisageables pour réduire et/ou exploiter les effets des incertitudes ;
2. les critères de sélection ;
3. la solution retenue.

3.1.5 Etape 4 : validation des solutions et contrôle des incertitudes

3.1.5.1 Objectifs

L'objectif de cette étape est de s'assurer que les solutions nécessaires ont été correctement mises en place et que les incertitudes nouvellement introduites sont prises en compte par une nouvelle itération de gestion ou bien par une procédure de modification.

Un bon suivi des incertitudes fournit des informations utiles qui pourront aider à la prise de décisions à l'avance, au cas où de nouvelles incertitudes se produiraient.

3.1.5.2 Entrées / Activités recommandées / Outils / Sorties

Entrées

Les éléments d'entrée de cette étape sont les éléments de sortie de l'étape précédente.

Activités recommandées

Les activités principales pour la validation des solutions consistent à :

1. vérifier si la solution retenue pour le traitement des incertitudes a été correctement mise en oeuvre comme prévu ;
2. vérifier son efficacité, c'est-à-dire vérifier si les effets des incertitudes susceptibles d'avoir une influence négative (risques) sur le projet ont

été effectivement réduits ou éliminés, et si les effets des incertitudes susceptibles d'avoir une influence positive (opportunités) ont été effectivement exploités.

Si la validation échoue, il faut appeler une procédure de modification de la solution retenue.

Les activités principales pour le contrôle des incertitudes consistent à :

1. identifier les nouvelles incertitudes qui peuvent apparaître et mettre alors à jour la liste des incertitudes ;
2. choisir les stratégies et les procédures qui seront adoptées pour contrôler les nouvelles incertitudes identifiées.

En effet, les incertitudes évoluent au fur et à mesure du développement du projet ; par exemple, certaines incertitudes prévues disparaissent ou d'autres apparaissent.

Pour prendre en compte une incertitude nouvellement identifiée, on pourra déclencher une nouvelle itération de gestion des incertitudes ou bien appeler une procédure de modification, selon le contexte concret.

Outils méthodologiques

1. Revues et audits

Des revues et audits d'incertitudes permettent de surveiller et contrôler les incertitudes, et de vérifier régulièrement, à chaque phase d'un projet, si leur priorité et leurs effets ont changé. Tout changement peut nécessiter une nouvelle itération. Les vérificateurs des incertitudes examinent et documentent également l'efficacité des solutions mises en jeu pour traiter les incertitudes du projet.

Sorties

Les éléments de sortie de cette étape comportent :

1. La liste des incertitudes mises à jour, avec leurs caractéristiques ;
2. les résultats des revues et audits portant sur l'efficacité du traitement des incertitudes.

3.1.6 Activité auxiliaire : représentation des incertitudes

3.1.6.1 Objectifs

La représentation des incertitudes démarre à la fin de l'étape d'identification et est exécutée en parallèle avec les trois autres étapes.

Elle consiste à enregistrer la présence des incertitudes dans la base de connaissances du projet et à les modéliser, de sorte à établir et gérer un référentiel des incertitudes qui permette de les analyser et de les traiter.

L'utilisation de ce référentiel de gestion des incertitudes sera utile tout au long de l'organisation de projet et permettra de constituer une base de connaissances sur la gestion des incertitudes dans le domaine visé ou l'organisation concernée.

3.1.6.2 Entrées / Activités recommandées / Outils / Sorties

Entrées

Les éléments d'entrée de cette étape sont les éléments de sortie de toutes les autres étapes de la méthodologie : identification des incertitudes, analyse des effets des incertitudes, réduction et exploitation des incertitudes, validation des solutions et contrôle des incertitudes.

Activités recommandées et outils

Les activités recommandées et les outils adaptés dépendent de la démarche envisagée par l'organisation pour représenter les incertitudes

et ne peuvent pas être fixés a priori.

Comme nous l'avons indiqué dans le chapitre 2, les modélisations de produit et de processus basées sur des langages descriptifs comme UML ou SPEM ont été largement utilisées pour faciliter le développement de systèmes complexes. On pourra sans doute souvent représenter les incertitudes par ce type de langages.

Dans cette thèse, nous proposons de réaliser cette tâche par la méta-modélisation des incertitudes. L'établissement d'un méta-modèle de l'incertitude consiste à définir l'ensemble des concepts abstraits concernant l'incertitude ainsi que leurs interrelations. Il permettra de représenter les incertitudes de façon cohérente et normalisée, ce qui sera particulièrement utile pour les projets multidisciplinaires.

Dans la section 4 de ce chapitre, nous allons décrire en détail notre méta-modélisation des incertitudes dans le contexte d'un processus de développement.

Sorties

L'élément de sortie de cette étape est un référentiel des incertitudes.

3.1.7 Conclusion

Dans cette section, nous avons proposé une méthodologie de gestion des incertitudes qui est composée de cinq étapes : l'étape d'identification et de sélection des incertitudes, l'étape d'analyse de leurs effets, l'étape de réduction et d'exploitation, l'étape de validation des solutions et de contrôle des incertitudes, et la représentation des incertitudes. A chaque étape, nous avons décrit l'objectif, les éléments d'entrées et de sorties, les activités principales, ainsi que certains des outils à utiliser pour la réalisation de cette étape.

L'ensemble de ces descriptions permet de clarifier quels sont les éléments et activités nécessaires pour réaliser la gestion des incertitudes dans

un projet.

Dans la section suivante, nous allons voir comment intégrer la méthodologie de gestion dans un projet.

3.2 INTÉGRATION DES TÂCHES DE LA MÉTHODOLOGIE DANS UN PROJET

La construction d'un programme de gestion des incertitudes s'appuie sur la réalisation de l'ensemble d'activités défini dans la section précédente tout au long d'un projet.

Dans cette section, nous allons d'abord présenter les différentes phases d'un projet de développement de systèmes complexes ; puis nous montrerons le déroulement de chaque activité de la méthodologie au cours de ces phases et les liens logiques entre ces activités.

3.2.1 Phasage d'un projet

A chaque secteur industriel correspondent des spécificités du point de vue du découpage du projet en phases, ainsi que du vocabulaire. Cette spécificité est fonction :

- du métier : prescripteur, constructeur, bureau d'étude, ...,
- du secteur d'activité : énergie, transport, automobile, chimie et pétrochimie, ...,
- des habitudes et des normes de travail utilisées : définitions propres aux groupes industriels, définitions liées au SIL (Safety Integrity Level), ...,
- du projet lui-même : installation de production, produit grand public, produit industriel,...

Afin de pouvoir intégrer la méthodologie proposée dans un projet, le phasage suivant a été retenu, à partir de nombreux documents traitant

du développement de systèmes complexes dans le cadre des industries françaises (Blancho et Durand 2004) :

- Phase 1 : Etude de faisabilité (analyse fonctionnelle au niveau système, besoin, procédé)
- Phase 2 : Etude de base (architecture, analyse fonctionnelle au niveau des sous-systèmes)
- Phase 3 : Etude de détail (choix des matériels et des logiciels, études détaillées de réalisation)
- Phase 4 : Réalisation (intégration, validation, exploitation).

Le Tableau 3.1 reprend le phasage et le vocabulaire propre à chacun des domaines industriels et les met en correspondance avec le phasage retenu.

3.2.2 Intégration des activités de la méthodologie dans un projet

L'intégration des activités de gestion des incertitudes dans chaque phase d'un projet est montrée dans les Tableaux 3.2 et 3.3. Ce tableau décrit l'ensemble des activités de la méthodologie proposée, comme nous avons décrit dans la section 3.1 (cf. §3.1.2 à §3.1.5), et donne pour chacune d'elles :

- son libellé
- la ou les phases d'un projet dans lesquelles elle s'applique.

Ce tableau montre l'évolution de chaque activité au cours des différentes phases d'un projet (dans le sens horizontal) et la succession de ces activités (dans le sens vertical). Le contenu de chaque case du tableau correspond aux éléments de sortie de l'activité.

Phase retenue	Nucléaire	Transport	Défense	Automobile	Pétrole/ Chimie
Phase 1 Etude de faisabilité	Analyse (initiale) - Besoins - Etude fonctionnelle au niveau système	Etape 1 : Besoin - Analyse fonctionnelle - Profil de mission - Performances attendues Etape 2 : Faisabilité	Faisabilité - Cahier des charges fonctionnel - Faisabilité technico-économique - Enveloppe coût/délais	Préparation processus - Cahier des charges fonctionnel - Performances attendues - Pré dimensionnement - Enveloppe coût/délais	Etude préliminaire - Cahier des charges fonctionnel - Hypothèses techniques - Pré dimensionnement - Estimation coût/délais
Phase 2 Etude de base	Etude de base - Etudes fonctionnelles au niveau sous- système - Implantations - Spécification de réalisation (configuration géométrique)	Etape 3 : avant de projet - Technologie - Choix	Définition - Fixer le besoin - Fixer l'enveloppe coût/délais	Avant-projet sommaire - Plan de AP général - Simulation fonctionnelle - Implantation - Organisation atelier - Estimation coût global Avant-projet détaillé - Découpage marchés - Simulations flux - Besoin bâtiment - Elaboration contrats	Etude de base - Exigences fonctionnelles - Descriptif procédé et dimensionnement - Implantations - Définition technologies - Coût/délais investissements
Phase 3 Etude de détail	Etude de détail/prototypes - Etudes détaillées de réalisation - Références physiques des constituants	Etapes 4 et 5 : présérie développement - Etude de conception détaillée - Plans détaillés des contractants	Développement Qualification via études et essais	Commandes - Consultation fournisseurs - Choix fournisseur Etudes - Analyse fonctionnelle détaillée - Mise en œuvre politique - Standardisation - Mise à jour plans - Modes opératoires finaux	Etude de détail - Schématique procédé - Manuel opératoire - Schémas Tuyauterie - Instrumentation détaillée - Implantation détaillée - Cahier des charges construction - Métés - Dossier appel d'offre - Dossier de commandes
Phase 4 Réalisation	Réalisation - Fabrication - Essais - Livraison	Etape 6 : Construction - Réception des matériels - Initialisation des règles d'exploitation et de maintenance	Production - Fabrication - Contrôle - Acceptation	Réalisation - Suivi des approvisionnements - Réception de sous-ensembles - Programme détaillé (essaimontée en cadence) - Réception - Configuration de pièces de rechanges	Construction - Contrôle étude exécution - Contrôle essais finaux - Dossier acceptation mécanique - Dossier documents contractuels

TABLE 3.1 – Le phasage d'un projet de développement de systèmes complexes (Blanchot et Durand 2004)

Phases d'un projet	Etude de faisabilité	Etude de base	Etude de détail	Réalisation
Activités de gestion des incertitudes	Recueil des données de base relatives à un système existant et au processus de son développement	Données de base au niveau sous-système	Données de base jusqu'au niveau le plus bas	...
	Identification des incertitudes liées aux éléments du processus	Incertitudes liées aux éléments du processus au niveau sous-système	Incertitudes liées aux éléments du processus jusqu'au niveau le plus bas	Mise à jour d'incertitudes (prise en compte des modifications)
Identification des incertitudes	Sélection et la hiérarchisation des incertitudes	Incertitudes traitées en priorité au niveau sous-système	Incertitudes traitées en priorité jusqu'au niveau le plus bas	Mise à jour d'incertitudes (prise en compte des modifications)
	Identification et analyse des effets des incertitudes	Risques et opportunités au niveau système	Risques et opportunités jusqu'au niveau le plus bas	Mise à jour des risques et opportunités (prise en compte des modifications)
Analyse des effets	Identification des solutions envisageables pour réduire les risques et/ou pour exploiter les opportunités	Solutions envisageables au niveau système	Solutions envisageables jusqu'au niveau le plus bas	Mise à jour des solutions envisageables (prise en compte des modifications)
	Identification des critères de sélection de solutions	Critères de sélection au niveau système	Critères de sélection jusqu'au niveau le plus bas	Mise à jour des critères (prise en compte des modifications)
Réduction et exploitation des incertitudes	Détermination de la solution selon les critères de sélection de solutions	Solution retenue au niveau sous-système	Solutions retenues jusqu'au niveau le plus bas	Mise à jour des solutions retenues (prise en compte des modifications)

TABLE 3.2 – L'intégration des activités de la méthodologie dans un projet (a)

Phases d'un projet	Etude de faisabilité	Etude de base	Etude de détail	Réalisation
Activités de gestion des incertitudes
Validation des solutions et contrôle des incertitudes	<p>Vérification de la bonne mise en œuvre de la solution retenue pour le traitement des incertitudes</p> <p>Vérification de l'efficacité de la solution retenue</p> <p>Identification des nouvelles incertitudes</p>	<p>Résultat de la mise en œuvre de solutions retenues à tous les niveaux</p> <p>Risques réduits ou éliminés ; opportunités exploitées à tous les niveaux</p> <p>Mise à jour des informations sur les incertitudes à tous les niveaux (prise en compte des modifications)</p>
Représentation	<p>Le choix de stratégies pour contrôler les nouvelles incertitudes identifiées</p> <p>Base de connaissances au niveau système</p>	<p>Le choix de stratégies (vers une nouvelle itération ou la procédure de modification) au niveau sous-système</p> <p>Base de connaissances des incertitudes au niveau sous-système</p>	<p>Le choix de stratégies (vers une nouvelle itération ou la procédure de modification) jusqu'au niveau le plus bas</p> <p>Base de connaissances des incertitudes jusqu'au niveau le plus bas</p>	<p>Le choix de stratégies (vers une nouvelle itération ou la procédure de modification)</p> <p>Mise à jour de la base de connaissances (prise en compte des modifications)</p>

TABLE 3.3 – L'intégration des activités de la méthodologie dans un projet (b)

3.3 CATÉGORISATION DES INCERTITUDES DANS LE DOMAINE DES PROCESSUS DE DÉVELOPPEMENT

Nous avons discuté, dans le chapitre précédent, de la classification traditionnelle de l'information imparfaite (cf. Tableau 2.1) et conclu qu'elle est utile pour l'analyse de l'imperfection de l'information, mais qu'elle ne suffit pas pour organiser la gestion des incertitudes et représenter leur structure dans les processus de développement de systèmes.

Pour arriver à mieux gérer et représenter les incertitudes, nous allons donc proposer une autre catégorisation, en nous limitant au cas le plus fréquent, celui où l'imprécision peut être considérée comme une source de l'incertitude.

Cette catégorisation des incertitudes est présentée dans la Figure 3.2. Elle est basée sur les deux éléments fondamentaux des processus : l'activité et le livrable, que nous avons introduit dans le chapitre 2 (cf. § 2.2.1), et la manière dont les individus (ou les organisations) agissent dans l'exécution du processus.

Cette catégorisation permet de raffiner les incertitudes liées au processus de développement (en anglais : Process uncertainty) qui se rangent selon trois catégories : incertitude sur les activités (Activity uncertainty), incertitude sur les livrables (Deliverable uncertainty) et incertitude sur le comportement des processus (Process behavior uncertainty).

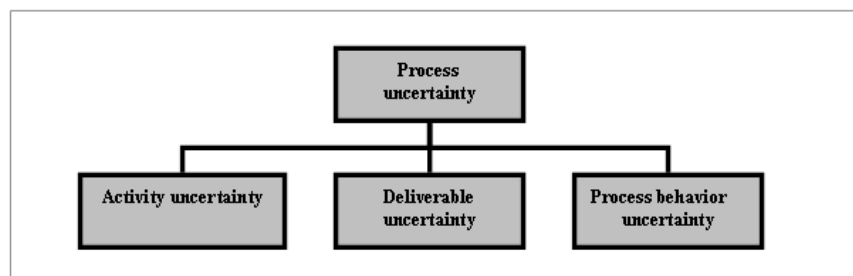


FIGURE 3.2 – Catégorisation des incertitudes dans le processus de développement

Une définition de chaque catégorie d'incertitudes est présentée ci-dessous :

- **Incertitude sur les activités** : à chaque activité sont associés des attributs sur lesquels des incertitudes peuvent apparaître, comme la durée, le coût ou encore l'état d'avancement. Les incertitudes sur une activité dépendent de la complexité de ses interactions avec les autres activités.
- **Incertitude sur les livrables** : à chaque livrable est associée une série de propriétés qui peuvent être incertaines, telles que la maturité du produit intermédiaire, les possibilités de réutilisation fonctionnelle, la complétude et l'exactitude de l'information, la résistance physique, etc.
- **Incertitude sur le comportement de processus** : c'est l'incertitude dans la manière dont les individus ou organisations agissent dans l'exécution du processus (Cohen et al. 1985). Dans la pratique, ce sont des incertitudes liées à l'intervention humaine dans le processus de conception et développement, dans la documentation, etc. Par exemple, des erreurs humaines peuvent se produire au cours du processus de maintenance d'un système mécanique.

Il convient de noter que la catégorisation proposée et les définitions fournies ne sont pas exhaustives ni universellement acceptées, mais sont représentatives pour l'ensemble des incertitudes dans le domaine de processus de développement.

Nous allons maintenant proposer notre méta-modélisation des incertitudes sur cette précédente catégorisation.

3.4 MÉTA-MODÉLISATION DES INCERTITUDES

Nous avons proposé, dans la méthodologie de la gestion des incertitudes, une étape de représentation des incertitudes. Rappelons que son

objectif est de mettre en évidence la présence des incertitudes dans la base de connaissances d'un projet et d'établir un référentiel qui permette de prévoir les incertitudes, leurs analyses et traitements pour les futurs projets.

Nous proposons de réaliser cette étape par la méta-modélisation des incertitudes pour pouvoir représenter les incertitudes de façon cohérente dans la modélisation de processus et/ou de produit, en nous limitant au domaine des processus de développement.

Pour arriver à établir un méta-modèle de l'incertitude, il nous faut répondre aux questions suivantes :

- *Q1 : Quel est le langage que nous choisissons pour modéliser les incertitudes ? Quels sont ses avantages et ses inconvénients ?*
- *Q2 : Comment modéliser les incertitudes en utilisant ce langage ?*

Dans cette section, nous allons d'abord choisir le langage de modélisation et le workflow de celle-ci, puis nous proposerons un méta-modèle de représentation des incertitudes, que nous appellerons "Uncertainty-of"; nous produirons un nouveau profil UML à partir de ce méta-modèle pour pouvoir ensuite décrire et intégrer la représentation des incertitudes plus facilement au niveau de modèle de processus; ce profil constituera une interface de modélisation indépendante du modèle de processus choisi. Nous concluons finalement sur les extensions possibles du méta-modèle "Uncertainty-of" et les limites auxquelles on se heurte.

3.4.1 Choix du langage de modélisation

Nous avons introduit, dans le chapitre 2, deux langages de modélisation des processus de développement de systèmes : UML et SPEM, et nous avons comparé leurs propriétés (Mécanisme d'extension, Popularité industrielle, Supporté par des outils pour la création de profils, Facilité d'intégration des modèles dans le projet). En fonction de ce qui est indiqué dans le Tableau 2 du chapitre 2, nous choisissons le langage UML

comme langage de modélisation des incertitudes plutôt que SPEM pour trois raisons essentielles :

Premièrement, du point de vue du cadre d'application de ces deux langages, UML peut permettre la modélisation des incertitudes grâce à ses mécanismes d'extension. L'utilisation des trois concepts de stéréotypes, de valeurs marquées, et de contraintes permet de s'adapter aux besoins descriptifs de l'incertitude dans le cadre du développement de systèmes. En revanche, SPEM est un profil d'UML qui ne peut servir qu'à la modélisation de processus de développement de logiciels.

Deuxièmement, du point de vue des outils opérationnels qui supportent ces deux langages, peu d'outils existants permettent d'étendre le langage SPEM, bien qu'il introduise les notions d'activité, rôles, produits de travail, processus et cycle de vie dans la représentation des processus.

Enfin, du point de vue plus général de l'intégration des modèles, nous pensons que les incertitudes ne sont pas seulement liées aux systèmes de processus, mais également aux systèmes de produits. Dans la perspective de recherches futures, le langage que nous choisissons doit aussi être capable de décrire les systèmes de produits. C'est pourquoi nous avons décidé de choisir le langage UML.

3.4.2 Workflow de la modélisation

Nous proposons un workflow de modélisation pour modéliser les incertitudes avec UML (Xiao et al. 2008) (Aranega 2008). Il comporte plusieurs étapes, comme il est indiqué dans la Figure 3.3 :

1. Etablir le méta-modèle des incertitudes, que nous allons décrire en détail dans la section 3.4.3. La construction du méta-modèle "Uncertainty-of" permet de représenter les concepts principaux du domaine de gestion des incertitudes. Chacun des concepts sera alors être représenté par une méta-classe du méta-modèle "Uncertainty-of" et les relations entre concepts seront représentées par des as-

sociations entre les méta-classes du méta-modèle. Dans notre cas, l'outil Topcased (Gauffillet 2006) a été utilisé pour la création du méta-modèle au format Ecore.

2. A partir du méta-modèle, créer un nouveau profil UML à partir des classes prédéfinies dans le méta-modèle UML 2.0. Dans notre cas, nous avons réalisé cette création par l'outil Papyrus UML et certains des modeleurs basés sur l'implémentation UML 2.0. Nous allons décrire cette partie en détail dans la section 3.4.4.
3. Intégrer les descriptions des incertitudes dans les modèles basés sur UML à partir des éléments fournis par le nouveau profil, c'est-à-dire modéliser les incertitudes au niveau du modèle, comme nous allons le voir sur un exemple dans l'étude de cas du chapitre 4.

Il est obligatoire que le nouveau profil respecte les principes existants de la modélisation UML (Muller 2000) (OMG 2003a) (Sinan 2002). En d'autres mots, nous devons garantir la conformité entre le méta-modèle des incertitudes et les modèles de processus fondés sur UML en respectant les contraintes originales UML.

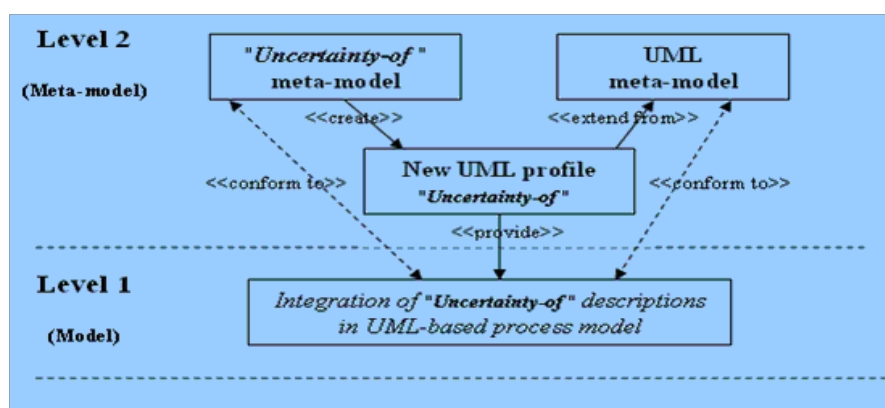


FIGURE 3.3 – Workflow de la modélisation

3.4.3 Méta-modèle de représentation des incertitudes "Uncertainty-of"

Pour modéliser les incertitudes, nous suivons le workflow de modélisation présenté dans la section 3.4.2.

Le méta-modèle "Uncertainty-of" se concentre sur le concept "incertitude" qui est représenté par la méta-classe "uncertainty_of". Il est constitué de trois paquetages principaux :

1. paquetage des incertitudes
2. paquetage des effets des incertitudes
3. paquetage des traitements des incertitudes

Dans les trois sous-sections suivantes, nous allons décrire chacun de ces trois paquetages, y compris la définition et les attributs de chaque méta-classe et ses associations.

Bien que certaines des définitions aient été fournies plus tôt dans les sections précédentes, ces définitions sont reprises intégralement dans cette section pour plus de clarté.

3.4.3.1 Paquetage des incertitudes

Ce paquetage contient trois méta-classes principales : "uncertainty_of", "uncertainty_of_requirement", "uncertainty_of_process", comme le montre la Figure 3.4.

La méta-classe "uncertainty_of"

Définition

La méta-classe "uncertainty_of" permet de représenter des doutes sur la validité et la vérité d'une information associée aux éléments d'un processus, ainsi que des imprécisions dans l'expression de cette information

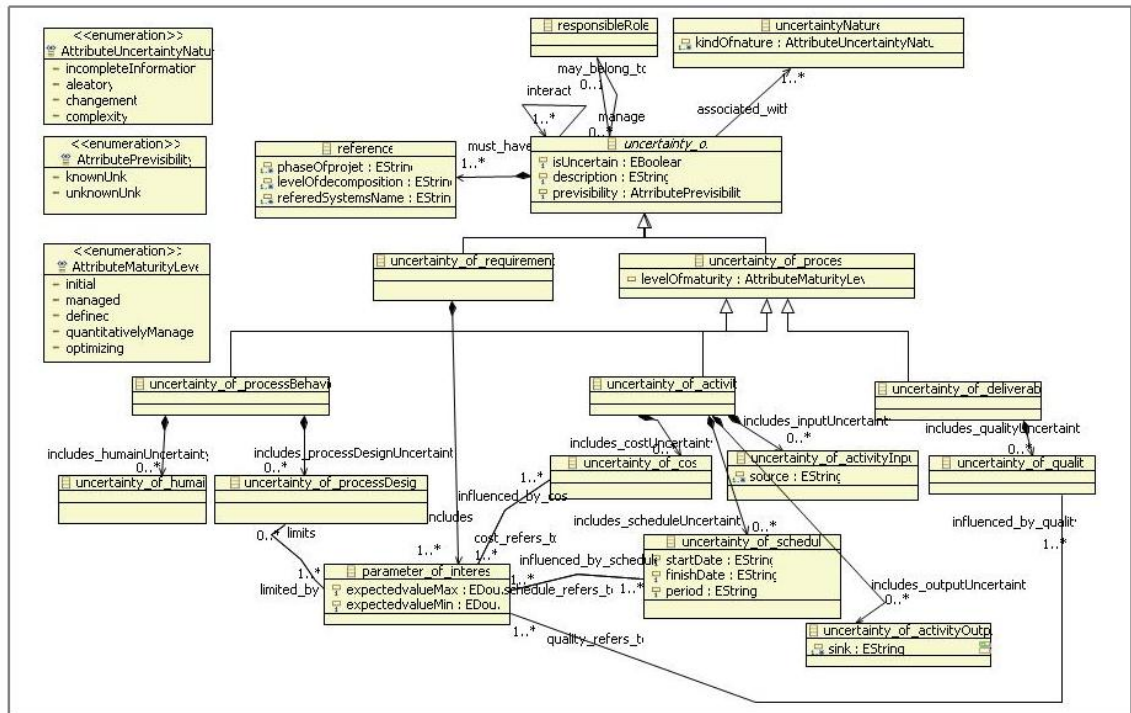


FIGURE 3.4 – Paquetage des incertitudes

(cf. § 2.1.5.1 du chapitre 2).

Attributs

- L'attribut **isUncertain** indique l'état d'une incertitude sur un élément; isUncertain est vrai si une incertitude identifiée n'a été pas traitée ou si son traitement n'est pas encore fini.
- L'attribut **descriptionOfuncertainty** décrit l'incertitude en détail.
- L'attribut **previsibility** indique le type de l'incertitude : "Known Unknown" ou "Unknown Unknown" (cf. chapitre 2, § 2.1.5.3)

Associations

- Une incertitude doit être associée à au moins une référence; la méta-classe "**reference**" permet d'indiquer à quels éléments et à quelle phase de projet l'incertitude est liée;

- Une incertitude peut être associée avec un **"responsibleRole"**, représentant le rôle qui est officiellement responsable de la gestion de cette incertitude.
- Une incertitude peut être associée à au moins une nature d'incertitude ; **"uncertaintyNature"** est utilisée pour indiquer si l'incertitude découle d'un manque d'information, d'un aléa, d'un changement ou de la complexité ; la valeur de cet attribut est du type énumération : `incompleteInformation`, `aleatory`, `changement`, `complexity` (cf. chapitre 2, § 2.1.5.2).
- Les méta-classes **"uncertainty_of_process"** et **"uncertainty_of_requirement"** sont des spécialisations de **"uncertainty_of"**. Elles héritent des attributs et des associations de leur méta-classe supérieure **"uncertainty_of"**.

La méta-classe "uncertainty_of_process"

Définition

La méta-classe **"uncertainty_of_process"** permet de représenter une famille d'incertitudes associées aux éléments des processus de développement.

Attributs

- L'attribut **levelOfmaturity** correspond à l'atteinte d'un niveau de capacité pour un processus.
- En plus, **"uncertainty_of_process"** hérite les attributs de sa méta-classe supérieure **"uncertainty_of"**.

Associations

La méta-classe **"uncertainty_of_process"** possède trois sous-méta-classes : **"uncertainty_of_activity"**, **"uncertainty_of_deliverable"**, **"uncer-**

tainty_of_processBehavior". Leur signification est conforme au contenu de la section 3.3 de ce chapitre.

Parmi ses sous-méta-classes,

- la méta-classe "uncertainty_of_activity" est composée des méta-classes suivantes :
 - "**uncertainty_of_activityInput**" : indique les incertitudes liées aux éléments d'entrée d'une activité,
 - "**uncertainty_of_activityOutput**" : indique les incertitudes liées aux éléments de sortie d'une activité,
 - "**uncertainty_of_cost**" : indique les incertitudes liées au budget de la réalisation d'une activité,
 - "**uncertainty_of_schedule**" : indique les incertitudes liées à la durée et aux délais de la réalisation d'une activité.

Une activité est incertaine quand un ou plusieurs des éléments qui la composent sont incertains.

- la méta-classe "**uncertainty_of_deliverable**" est composée de la méta-classe :
 - "**uncertainty_of_quality**" : indique les incertitudes liées aux propriétés de qualité d'un livrable de processus.

Un livrable est incertain quand il y a des doutes sur sa validité.

- la méta-classe "**uncertainty_of_processBehavior**" est composée des méta-classes :
 - "**uncertainty_of_processDesign**" : indique un choix restant ouvert parmi des solutions alternatives pour savoir comment réaliser un processus.
 - "**uncertainty_of_human**" : indique les incertitudes sur les compétences (et autres caractéristiques) des personnes lors de l'exécution

du processus, sur ce que les humains eux-mêmes décideront.

La méta-classe "uncertainty_of_requirements"

Définition

La méta-classe "**uncertainty_of_requirements**" permet de représenter les incertitudes liées aux exigences du projet.

Attributs

Elle hérite des attributs de la classe "**uncertainty_of**".

Associations

Elle est composée de la méta-classe "**parameter_of_interest**". Les paramètres d'intérêt indiquent les propriétés quantifiables à propos de l'objectif du projet, comme la qualité, le coût, le délai du projet.

3.4.3.2 Paquetage des effets des incertitudes

Le paquetage effet des incertitudes se concentre sur la méta-classe "**uncertainty_effect**", comme le montre la Figure 3.5.

La méta-classe "uncertainty_effect"

Définition

La méta-classe "**uncertainty_effect**" permet de représenter une influence de l'incertitude sur les objectifs du projet.

Attributs

L'attribut description décrit un effet d'une incertitude en détail.

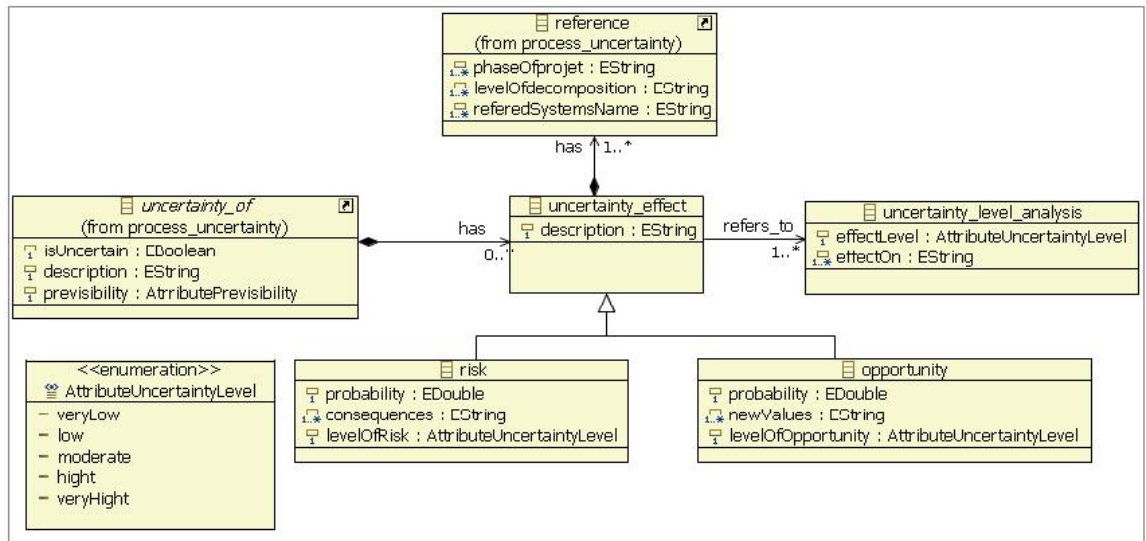


FIGURE 3.5 – Paquetage des effets des incertitudes

Associations

- Une incertitude peut avoir des influences sur l’objectif du projet. La méta-classe "**uncertainty_effet**" permet de les représenter.
- Un effet doit être associé à au moins une référence ; la méta-classe "**reference**" permet d’indiquer à quels éléments et à quelle phase de projet l’effet est lié ;
- La méta-classe "**uncertainty_effet**" a deux sous méta-classes : "**risk**", "**opportunity**", qui héritent les attributs et les association de leur méta-classe supérieure.
- Pour chaque effet de l’incertitude, on peut réaliser une analyse en termes qualitatifs avec une estimation de ses effets qualifiée de très forte, forte, modérée, faible ou très faible. La méta-classe "**uncertainty_level_analysis**" représente cette analyse.

Les méta-classes "risk" "opportunity"

Définition

- "risk" représente l'effet négatif d'une incertitude sur l'objectif du projet ;
- "opportunity" représente l'effet positif d'une incertitude sur l'objectif du projet.

Attributs

En plus des attributs de leurs méta-classe supérieure "uncertainty_effect", les attributs supplémentaires de la méta-classe "risk" sont :

- **probability** : représente l'évaluation du caractère "probable" d'occurrence d'un risque ;
- **consequences** : représente les événements prévus ou imprévus faisant suite à un risque ;
- **levelOfRisk** : représente le niveau d'un risque à partir de la combinaison de la probabilité d'occurrence et des conséquences de ce risque.

Les attributs supplémentaires de "opportunity" sont :

- **probability** : représente l'évaluation du caractère " probable " d'occurrence d'une opprtunité ;
- **newValues** : représente les événements prévus ou imprévus faisant suite à une opportunité ;
- **levelOfOpportunity** : représente le niveau d'une opportunité à partir de la probabilité d'occurrence et des nouvelles valeurs apportées par cette opportunité.

Associations

Il n'y a pas d'association supplémentaire pour ces deux méta-classes.

La méta-classe " uncertainty_level_analysis "

Définition

La méta-classe "**uncertainty_level_analysis**" permet de représenter l'analyse en termes qualitatifs d'un effet de l'incertitude.

Attributs

- L'attribut **effectLevel** est utilisé pour définir le niveau de l'effet en termes qualitatifs tels que : très fort, fort, modéré, faible et très faible.
- L'attribut **effectOn** est utilisé pour indiquer si l'effet d'une incertitude est sur les techniques (par exemple, la faisabilité, l'opérabilité, productibilité, testabilité, l'efficacité de systèmes), les coûts (par exemple, les estimations, le budget), le calendrier (par exemple, milestones), et le programme (par exemple, contrat).

Associations

Il n'y a pas d'association supplémentaire pour ces deux méta-classes.

3.4.3.3 Paquetage des traitements des incertitudes

Le paquetage des traitements des incertitudes se concentre sur deux méta-classes principales : "**uncertainty_treatment**", "**impact_evaluation**", comme le montre la Figure 3.6.

La méta-classe "uncertainty_treatment"

Définition

La méta-classe "**uncertainty_treatment**" permet de représenter un traitement envisageable pour réduire ou exploiter les effets des incertitudes (risques ou opportunités).

Attributs

- L'attribut **treatmentDescription** décrit en détail le traitement des incertitudes envisagé.

- L'attribut **referredCharacteristic** décrit les caractéristiques de systèmes référées au traitement des incertitudes envisagé.

Associations

- Chaque effet doit prévoir au moins un traitement représenté par "uncertainty_treatment".
- La méta-classe "uncertainty_treatment" a deux sous méta-classes :
 - "risk_treatment" indique le traitement envisageable pour réduire les risques du projet;
 - "opportunity_treatment" indique le traitement envisageable pour exploiter les opportunités du projet.
- Un traitement des incertitudes peut prévoir une évaluation des impacts de l'implémentation de ce traitement sur l'objectif du projet. Cette évaluation est représenté par la méta-classe "impact_evaluation".

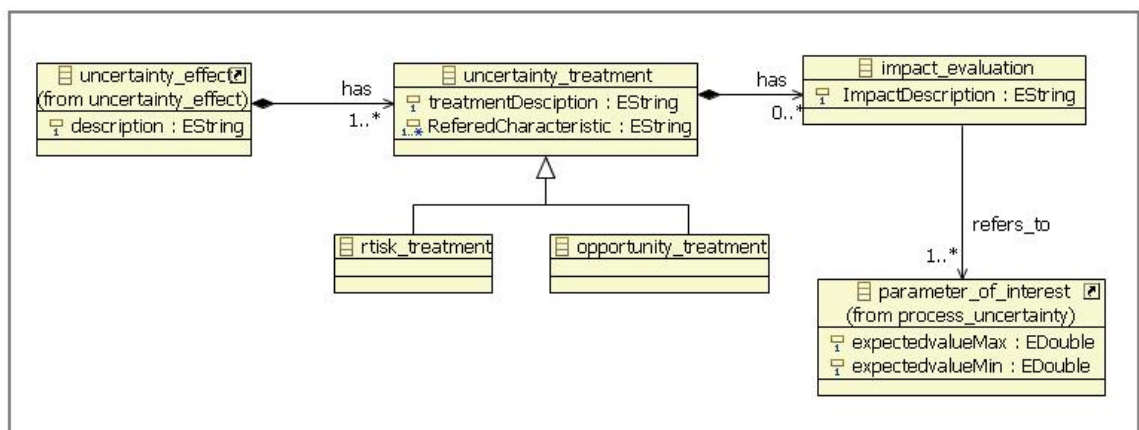


FIGURE 3.6 – Paquetage des traitements des incertitudes

La méta-classe "impact_evaluation"

Définition

La méta-classe "impact_evaluation" permet de représenter l'évaluation

d'impacts de l'implémentation d'un traitement des incertitudes sur l'objectif de projet.

Attributs

L'attribut `impactDescription` décrit en détail les impacts qui sont introduits par l'implémentation d'un traitement.

Associations

"**impact_evaluation**" se réfère à "**parameter_of_interest**" en représentant l'évaluation d'impact relative aux propriétés quantifiables concernant l'objectif du projet.

3.4.3.4 Conclusion du méta-modèle "Uncertainty-of"

Le méta-modèle "Uncertainty-of" a pour objectif de décrire les incertitudes et le principe de leur traitement dans le processus de développement de systèmes. Il s'établit selon trois axes :

- premièrement, le raffinement des incertitudes dans le processus de développement ;

L'ensemble des méta-classes "uncertainty_of_process", "uncertainty_of_activity", "uncertainty_of_activityInput", "uncertainty_of_activityOutput", "uncertainty_of_cost", "uncertainty_of_schedule", "uncertainty_of_processBehavior", "uncertainty_of_processDesign", "uncertainty_of_humain", "uncertainty_of_deliverable", "uncertainty_of_quality", "uncertainty_of_requirements", "parameter_of_interest", contient les incertitudes représentatives dans le processus de développement ;

- deuxièmes, l'effet de ces incertitudes sur l'objectif du projet ;

L'ensemble des méta-classes "uncertainty_effect", "risk", "opportunity", "uncertainty_level_analysis" décrit les effets potentiels des incertitudes et l'analyse réalisée en termes qualitatifs pour attribuer

un niveau à chaque effet dans le processus de développement ;

- troisièmes, le traitement des effets des incertitudes.

L'ensemble des méta-classes "uncertainty_treatment", "risk_treatment", "opportunity_treatment", "impact_evaluation" représente les traitements envisageables face aux différents effets des incertitudes et l'évaluation des impacts de la réalisation du traitement dans le processus de développement.

Le raffinement des incertitudes a été proposé de point de vue processus de développement de systèmes. Pour représenter les incertitudes de point de vue système dans nos recherches futures, on pourra étendre ce méta-modèle en ajoutant une branche "uncertainty_of_system" dans la même couche que "uncertainty_of_process" et "uncertainty_of_requirements".

Le principe de traitement des incertitudes est toujours valable pour les nouvelles méta-classes introduites. C'est-à-dire : les incertitudes conduisent à des risques, des opportunités ou des modifications, qui sont gérés techniquement par des approches de réduction de risques, d'exploitation d'opportunités, ou de corrections qui espèrent aboutir à des résultats souhaités.

Nous créons un nouveau profil en décrivant la définition technique de ce profil dans la section suivante.

3.4.4 Définition technique du profil "Uncertainty-of"

Techniquement, un profil UML est un ensemble de stéréotypes, de valeurs marquées et de contraintes. La définition de ces éléments a été fournie dans le chapitre 2 (cf. § 2.2.3.1).

Les stéréotypes, valeurs marquées et contraintes permettent d'établir une correspondance entre les concepts UML et les concepts du domaine représentés par le profil. Il est important de noter qu'en principe seule

la partie "définition technique" suffit à définir un profil selon le standard UML (OMG 2005) (OMG 2003a).

Notons que, dans notre cas, les concepts du domaine ont été décrits par notre méta-modèle "Uncertainty-of" établi dans la section précédente (cf. § 3.4.3).

3.4.4.1 Stéréotypes, valeurs marquées, contraintes structurelles

La Figure 3.7, la Figure 3.8 et la Figure 3.9 montrent la représentation graphique du profil "Uncertainty-of" dans le diagramme de classe. Ce profil est très proche de notre méta-modèle "Uncertainty-of". Nous allons décrire ce profil en listant les stéréotypes et en identifiant le concept du domaine, les "classes de base"¹, les valeurs marquées ainsi que les contraintes structurelles associées à chaque stéréotype.

Le Tableau 3.4 et le Tableau 3.5 donnent la liste complète des stéréotypes, des concepts, des classes de base, des stéréotypes supérieurs, des valeurs marquées et des contraintes associées à un stéréotype lors de la création du profil dans le diagramme de classes.

En général, le nom du stéréotype correspond à son nom dans le méta-modèle; les définitions des stéréotypes et leurs valeurs marquées sont proches des attributs des méta-classes correspondantes dans le méta-modèle (cf. § 3.4.3).

Les classes de base associées à un stéréotype sont désignées selon l'utilisation du profil. Par exemple, le stéréotype «Uncertainty-of» peut

1. Les stéréotypes, valeurs marquées et contraintes d'un profil sont rattachés à des méta-classes du méta-modèle UML; nous appelons ces méta-classes "classe de base".

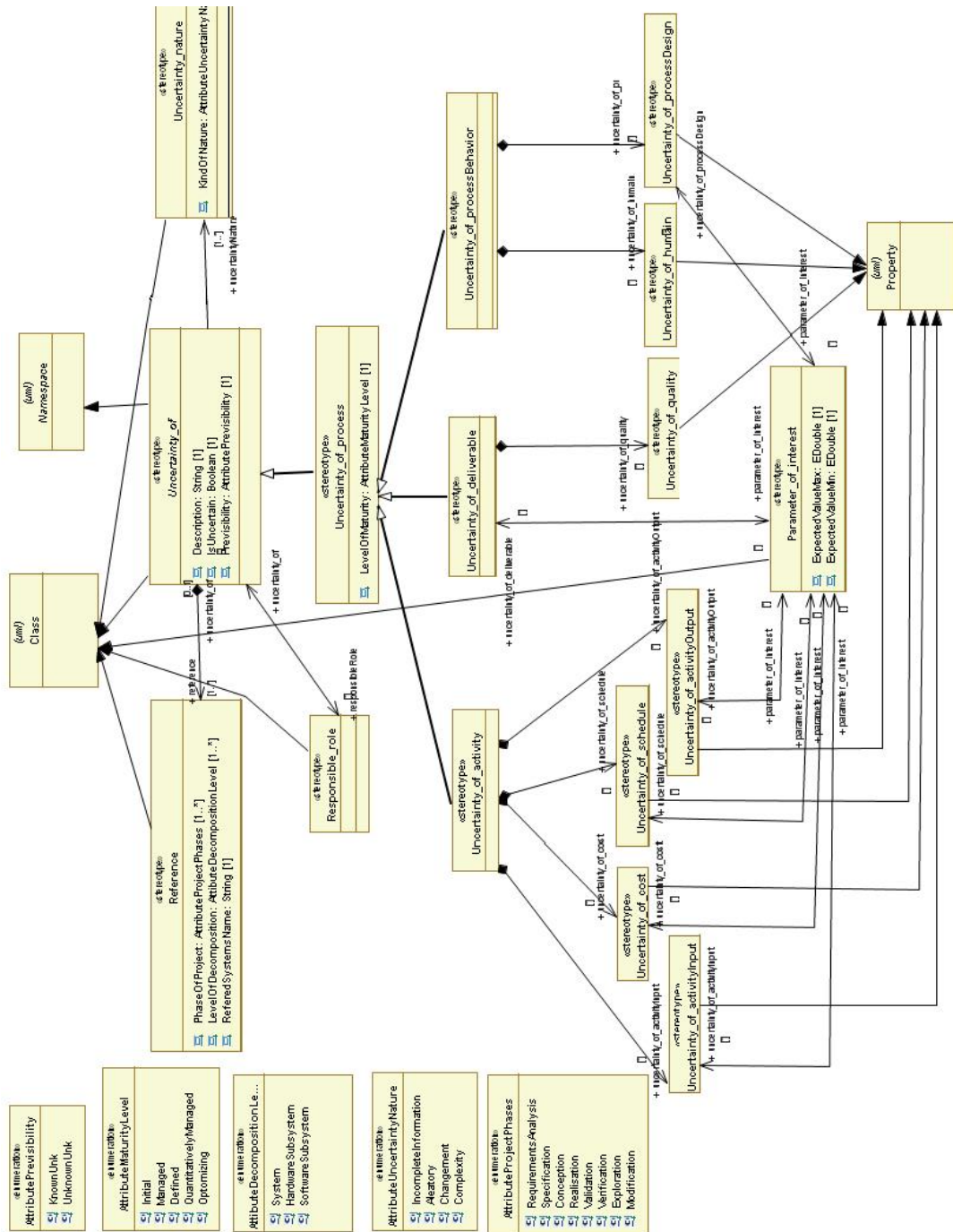


FIGURE 3.7 – Représentation graphique du profil "Uncertainty-of" dans le diagramme de classes (a)

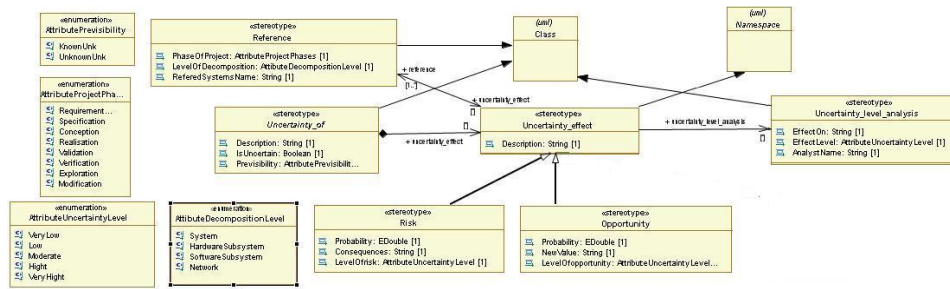


FIGURE 3.8 – Représentation graphique du profil "Uncertainty-of" dans le diagramme de classes (b)

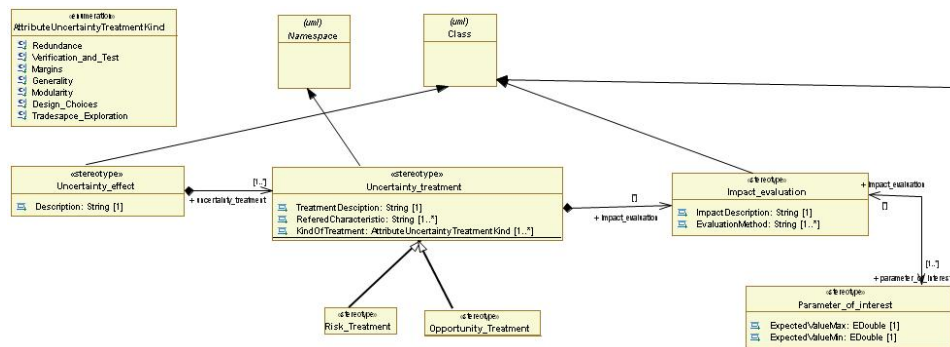


FIGURE 3.9 – Représentation graphique du profil "Uncertainty-of" dans le diagramme de classes (c)

être une classe, un commentaire, un paquetage dans les diagrammes UML.

La contrainte structurelle associée à un stéréotype correspond à une dépendance forte de ce stéréotype à un autre stéréotype.

Nous n'avons pas défini d'icônes pour les éléments du profil.

3.5 CONCLUSION DU CHAPITRE

Les propositions de ce chapitre explicitent une méthodologie de gestion des incertitudes dans le développement de systèmes complexes. Nous avons mis l'accent sur une méta-modélisation des incertitudes basée sur une nouvelle catégorisation des incertitudes liées aux éléments du processus. Nous avons également proposé un nouveau profil UML "Uncertainty-of" qui permet de représenter les incertitudes avec les outils UML afin de rendre ce méta-modèle opérable.

Stereotype	Meta-model concept	Base UML 2.0 class	Super stereotype	Tagged values	Structure constraint
« Uncertainty_of »	Uncertainty_of	Namespace	-	isUncertain : Boolean DescriptionOfUncertainty : String Previsibility : AttributePrevisibility	-One uncertainty must have at least one reference which is represented by stereotype <<reference>> to indicate the presence of this uncertainty related to which elements and in which phase of the project.
« Reference »	Reference	Classe	-	PhaseOfProject : AttributeProjectPhases LevelOfDecomposition : AttributeDecompositionLevel ReferredSystems : String	-
« ResponsibleRole »	Responsible Role	Classe	-	Reponsible : String	-
« UncertaintyNature »	Uncertainty Nature	Classe	-	KindOfNature : AttributeUncertaintyNature	-
« Uncertainty_effect »	Uncertainty_effect	Namespace	-	EffectOn : String Description : String	Each effect must have at least one uncertainty treatment.
« Uncertainty_of_requirements »	uncertainty_of_requirements	Namespace	« uncertainty_of »	It inherits tagged values of its super stereotypes.	-
« Parameter_of_interest »	parameter_of_interest	Classe	-	ExpectedValueMax : UnlimitedNatural ExpectedValueMin : UnlimitedNatural	-
« Uncertainty_of_process »	uncertainty_of_process	Namespace	« uncertainty_of »	It inherits tagged values of its super stereotypes. Other tagged values: LevelOfMaturity : AttributeMaturityLevel	-
« Uncertainty_of_deliverable »	uncertainty_of_deliverable	Namespace	« uncertainty_of_process » :: « uncertainty_of »	It inherits tagged values of its super stereotypes.	-
« Uncertainty_of_quality »	uncertainty_of_quality	Property	-	ReferredCharacteristic : String	-
« Uncertainty_of_activity »	uncertainty_of_activity	Namespace	« uncertainty_of_process » :: « uncertainty_of »	It inherits tagged values of its super stereotypes.	-
« Uncertainty_of_cost »	uncertainty_of_cost	Property	-	MaxValue : UnlimitedNatural MinValue : UnlimitedNatural	-
« Uncertainty_of_schedule »	uncertainty_of_schedule	Property	-	StartDate : String FinishDate : String Period : String	-
« Uncertainty_of_activityInput »	uncertainty_of_activityInput	Property	-	Source : String	-
« Uncertainty_of_activityOutput »	uncertainty_of_activityOutput	Property	-	Sink : String	-

TABLE 3.4 – Description technique du profil "Uncertainty-of" dans le diagramme de classes (a)

Stereotype	Meta-model concept	Base UML 2.0 class	Super stereotype	Tagged values	Structure constraint
« Uncertainty of processBehavior »	uncertainty_of_processBehavior	Namespace	« uncertainty_of_process » :: « uncertainty_of »	If inherits tagged values of its super stereotypes.	-
« Uncertainty of humain »	uncertainty_of_humain	Property	-	HumanErrorDescription : String	-
« Uncertainty of processDesign »	uncertainty_of_processDesign	Property	-	Description : String	-
« Risk »	risk	Namespace	« uncertainty_effect »	If inherits tagged values of its super stereotypes. Other tagged values: Probability : UnlimitedNatural Consequences : String LevelOfRisk : AttributeUncertaintyLevel	-
« Opportunity »	opportunity	Namespace	« uncertainty_effect »	If inherits tagged values of its super stereotypes. Other tagged values: Probability : UnlimitedNatural NewValue : String LevelOfOpportunity : AttributeUncertaintyLevel	-
« Uncertainty_level_analysis »	uncertainty_level_analysis	Classe	-	EffectLevelDefinition : AttributeUncertaintyLevel analysisDescription	-
« Uncertainty_treatment »	uncertainty_treatment	Namespace	-	TreatmentDescription : String ReferredCharacteristic : String KindOfTreatment : AttributeUncertaintyTreatmentKind	-
« RiskTreatment »	risk_treatment	Namespace	« uncertainty_treatment »	If inherits tagged values of its super stereotypes.	-
« OpportunityTreatment »	opportunity_treatment	Namespace	« uncertainty_treatment »	If inherits tagged values of its super stereotypes.	-
« Impact_evaluation »	impact_evaluation	Classe	-	ImpactDescription : String EvaluationMethod : String	-
« AttributeUncertaintyNature »	AttributeUncertaintyNature	Enumeration	-	{IncompleteInformation, Aleatory, Changeament, Complexity}	-
« AttributePrevisibility »	AttributePrevisibility	Enumeration	-	{knownUnk, unknownUnk}	-
« AttributeMaturityLevel »	AttributeMaturityLevel	Enumeration	-	{Initial, Managed, Defined, QuantitativelyManaged, Optimizing}	-
« AttributeUncertaintyLevel »	AttributeUncertaintyLevel	Enumeration	-	{VeryLow, Low, Moderate, High, VeryHigh}	-
« AttributeUncertaintyTreatmentKind »	AttributeUncertaintyTreatmentKind	Enumeration	-	{Redundance, Verification and Test, Margins, Generality, Modularity, Design_Choices, Tradesapce_Exploration}	-
« AttributeDecompositionLevel »	AttributeDecompositionLevel	Enumeration	-	{System, HardwareSubsystem, SoftwareSubsystem, Network}	-
« AttributeProjectPhases »	AttributeProjectPhases	Enumeration	-	{RequirementsAnalysis, Specification, Conception, Realization, Validation, Verification, Exploration, Modification}	-

TABLE 3.5 – Description technique du profil "Uncertainty-of" dans le diagramme de classes (b)

L'avantage de la méthodologie proposée est de clarifier un cadre de pensée qui permet de guider l'organisation des activités de gestion des incertitudes dans un projet. En définissant une méthodologie de gestion, nous fournissons :

- un moyen d'organiser des activités de gestion des incertitudes dans un projet ;
- une démarche de modélisation des incertitudes avec le langage UML ;
- un méta-modèle des incertitudes dans le cadre des processus de développement qui est réutilisable sur des projets similaires, et extensible pour la modélisation des incertitudes liées au produit ;
- un nouveau profil UML "Uncertainty-of" qui peut rendre ce méta-modèle opérable avec les outils UML.

Notons que l'état actuel de l'implémentation du profil reste dans le cadre du diagramme de classe UML. C'est une des limites de notre étude.

Nous espérons par ce biais donner aux entreprises et organisations les moyens de mieux maîtriser et suivre les incertitudes dans leur projets en diminuant leurs effets négatifs et en exploitant leurs effets positifs sur l'objectif du projet.

Dans le chapitre suivant, nous allons appliquer la méthodologie que nous venons de proposer à un exemple industriel.

CAS D'ÉTUDE

LE chapitre précédent portait sur la proposition d'une méthodologie de la gestion des incertitudes et d'un nouveau profil UML "Uncertainty_of" permettant de les représenter. Ce chapitre a pour objectif d'illustrer l'applicabilité de la démarche proposée dans un vrai projet industriel. L'étude de ce cas nous a permis non seulement de valider certains aspects essentiels de nos propositions, mais aussi d'apporter un regard critique sur la méthodologie et le profil eux-mêmes, sur leur construction et leur utilisation. Pour des raisons de confidentialité, nous ne pouvons citer ni l'entreprise ni le produit.

4.1 CONTEXTE

L'entreprise A a été choisie pour concevoir un nouveau modèle de système d'arrêt d'urgence pour une centrale électrique au sud de la Chine. La conception et le développement d'un tel système doit répondre à la norme de sécurité CEI 61508¹ afin de obtenir une licence Safety Integrity Level (en terme de sécurité) pour son utilisation en Chine.

Un système d'arrêt d'urgence est en général constitué d'automates industriels. S'ils sont trop fréquemment utilisés, des arrêts d'urgence

1. La norme CEI 61508 est une norme internationale traitant de la sécurité fonctionnelle des systèmes électriques/électroniques et électroniques programmables (E/E/PE); elle correspond à la norme européenne EN 61508 adoptée par le CENELEC en 2002. Elle est aujourd'hui utilisée comme référentiel par tous les grands secteurs industriels (transport, énergie, avion). Cette norme garde une approche très générique de façon à pouvoir intégrer tous les systèmes de sécurité E/E/PE en proposant un cycle de vie de sûreté (CEI61508 2003).

répétés sur le procédé de production d'électricité se révèlent économiquement préjudiciables, voire dangereux. Pour les limiter, il est impératif que les architectures des automates offrent à la fois sécurité et disponibilité.

Dans ce contexte, il existe un nombre important d'incertitudes dans le processus de développement du système, du point de vue de la sécurité et de la disponibilité. Nous nous sommes intéressés à la gestion des incertitudes dans le processus de conception système. Pour cela, une étude de six mois sur l'analyse de la conformité du futur produit aux exigences CEI 61508 a été réalisée en préalable.

Le résultat de cette étude nous a permis d'identifier les incertitudes les plus importantes et urgentes à traiter. Les deux problèmes les plus importants et urgents, des points de vue de la sécurité et de la gestion du projet, sont traités dans les deux sections suivantes :

- l'incertitude sur l'architecture du système d'arrêt d'urgence,
- les incertitudes sur la durée des activités du processus de définition préliminaire dans ce projet.

Nous avons traité ces incertitudes en mettant en oeuvre notre méthodologie de gestion des incertitudes. Notons que, dans ce cas d'étude, nous avons appliqué certaines étapes de notre méthodologie en tenant compte des réalités du traitement des incertitudes identifiées ; par exemple, l'étape de validation des solutions et de contrôle des incertitudes, qui doit être réalisée après la mise en oeuvre des solutions, n'a pas pu être réalisée, puisqu'on était en amont du processus de développement dans ce projet. Nous avons donc réalisé cette mise en oeuvre en suivant le schéma bleu montré dans la Figure 4.1.

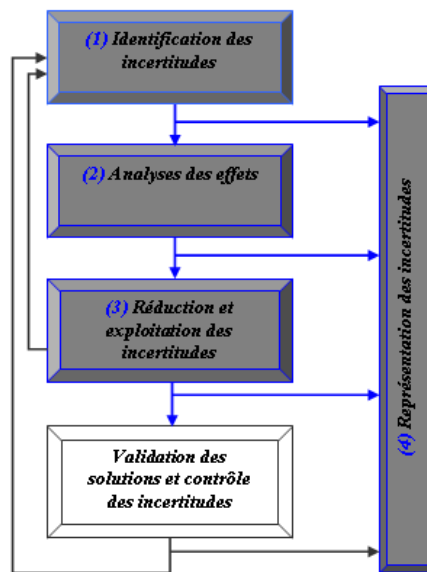


FIGURE 4.1 – Le schéma de synthèse de la mise en oeuvre de la méthodologie dans ce cas d'étude

4.2 TRAITEMENT DES INCERTITUDES SUR L'ARCHITECTURE DU SYSTÈME

Pour arriver à traiter l'incertitude sur l'architecture du système, dans cette section, nous allons d'abord décrire en détail l'architecture du système que nous avons étudié (cf. §4.2.1) ; puis nous appliquerons successivement, dans les sous-sections suivantes, les étapes de notre méthodologie de gestion des incertitudes (cf. §4.2.2-§4.2.4), qui sont :

1. l'identification et la sélection des incertitudes
2. l'analyse des effets des incertitudes ;
3. la réduction et /ou l'exploitation des incertitudes.

Nous montrerons enfin la représentation des incertitudes avec le diagramme de classe UML en appliquant notre profil "Uncertainty-of" (cf. §4.2.5).

4.2.1 Introduction : l'architecture du système

Pour rendre le système pleinement qualifiable, les architectures des automates de sécurité doivent fournir des garanties à la fois de sécurité et de disponibilité. Ceci passe par la mise en oeuvre d'architectures redondantes.

La redondance servira à garantir que le système de sécurité met le procédé dans un état sûr quelles que soient les circonstances. Selon le résultat que l'on recherche, les redondances ne s'organisent pas de la même façon. Le choix d'une architecture plutôt qu'une autre dépend en effet du niveau de sécurité que l'on veut atteindre, mais aussi du niveau de disponibilité que l'on souhaite.

Pour répondre aux exigences des clients chinois, les experts de l'entreprise A ont décidé d'adopter le principe de la TMR (Triple Modular Redundancy) (cf. Figure 4.2), appelé aussi tolérance Deux sur Trois (2003) ou bien Triplex, qui correspond à une architecture impliquant (Motet et Geffroy 2002) :

- trois modules fonctionnels identiques (P_1, P_2, P_3),
- un module de vote qui élabore la sortie finale à partir des trois sorties des modules par un vote majoritaire, et
- un module de détection qui compare deux à deux les sorties S_1, S_2 et S_3 et délivre un signal d'erreur dès qu'une différence est décelée.

Lorsqu'une différence est signalée, le dispositif continue à fonctionner, mais une maintenance est déclenchée.

Basée sur le principe Triplex, la définition d'une architecture d'arrêt d'urgence va principalement dépendre des paramètres suivants :

- le nombre et les caractéristiques des modules d'acquisitions des entrées qui vont alimenter les modules fonctionnels ;
- l'existence (ou non) d'un dialogue entre les modules fonctionnels ;

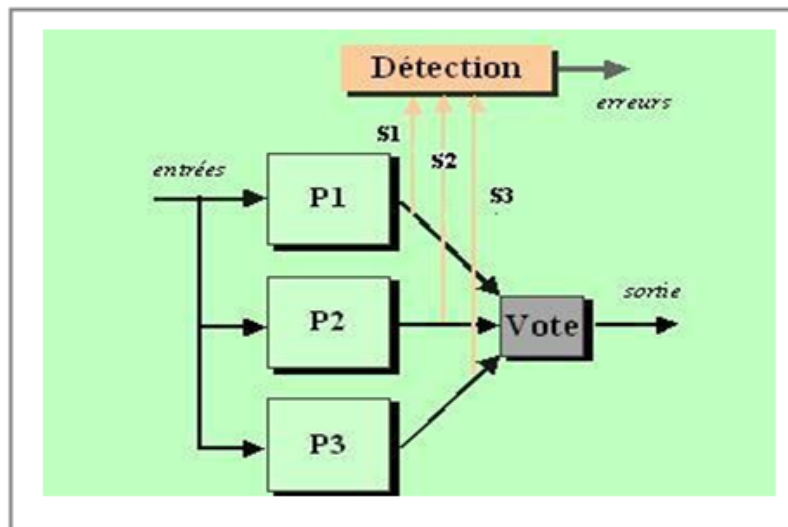


FIGURE 4.2 – Le principe de la TMR avec détection des erreurs

- la technologie de détection des pannes des capteurs.

Les experts de l'entreprise A ont proposé très rapidement une première maquette conforme à cette définition, qui avait déjà été exploitée pour des applications similaires dans des projets antérieurs de l'entreprise. Cette architecture, comme le montre la Figure 4.3, comporte :

- trois modules d'acquisition d'entrées de pression (P_1 , P_2 et P_3);
- trois modules d'acquisition d'entrées de température (T_1 , T_2 et T_3);
- trois unités de traitement (UT) qui incluent les mécanismes de détection de pannes et de reconfiguration du produit.

En outre, il n'y a pas de dialogue entre les trois modules fonctionnels.

4.2.2 Identification et sélection des incertitudes

Cette architecture possède un défaut que nous avons identifié lors de notre étude : elle ne tolère pas certaines fautes des capteurs qu'elle devrait tolérer.

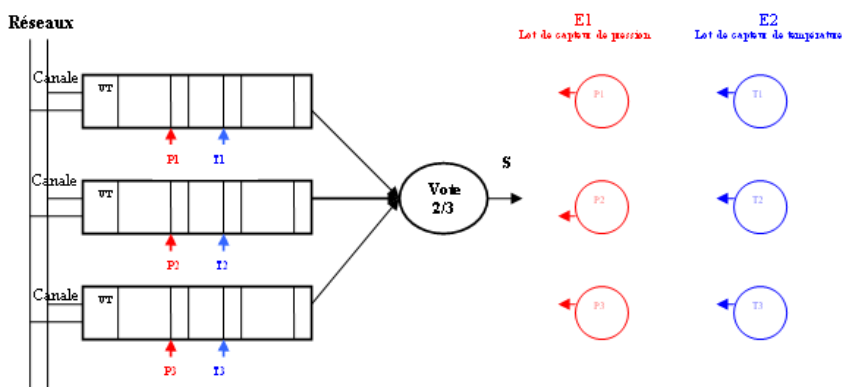


FIGURE 4.3 – Première architecture proposée par les experts

Ainsi, le Tableau 4.1 montre les scénarios de tolérance aux fautes de l'architecture proposée. Par exemple, une panne sur le capteur de pression P1 et une autre panne sur le capteur de température T2 conduisent à la perte de deux modules, et donc à un déclenchement intempestif de l'arrêt d'urgence.

Pannes sur	Tolérance aux fautes	effets
P1 ou P2 ou P3	Oui	Sans effets
T1 ou T2 ou T3	Oui	Sans effets
P1 et T1	Oui	Sans effets
P2 et T2	Oui	Sans effets
P3 et T3	Oui	Sans effets
(P1 et T2) ou (P1 et T3)	Non	Déclenchement intempestif
(P2 et T1) ou (P2 et T2)	Non	Déclenchement intempestif
(P3 et T1) ou (P3 et T2)	Non	Déclenchement intempestif

TABLE 4.1 – Les scénarios de tolérance aux fautes de l'architecture 2003

L'objectif de notre étude a donc été :

- de comprendre (ou prévoir) les effets de ce défaut de conception sur la sécurité et la disponibilité du système ; ces effets étaient incertains, et il fallait étudier cette incertitude pour guider les efforts du concepteur dans l'objectif d'améliorer l'architecture proposée ;

- de comparer les performances des différentes solutions pour optimiser l'architecture ;
- de démontrer la conformité de la solution choisie au niveau architecture avec les critères sécurité et disponibilité ;
- de gérer les données concernant cette incertitude avec l'aide du langage UML.

Les sections suivantes détaillent le déroulement de l'étude et ses résultats.

4.2.3 Analyse des effets des incertitudes

Nous allons maintenant analyser les effets de l'incertitude identifiée, correspondant à la deuxième étape du schéma de la méthodologie (cf. Figure 4.1).

Une feuille d'analyse des effets des incertitudes, inspirée de la méthode AMDEC (Analyse des Modes de Défaillance, de leurs Effets et de leurs Causes)(Faucher 2004) mais "customisée" et déjà couramment utilisée dans l'entreprise A, peut être utilisée pour rationaliser cette étape et assurer une évaluation des effets. La Figure 4.4 montre un exemple de l'utilisation de cette feuille pour décrire en détail un effet négatif (le déclenchement intempestif) d'une défaillance simultanée de deux capteurs appartenant à des modules distincts et mesurant des grandeurs différentes.

Nom effet négatif :	Déclenchement intempestif		
Phase de projet :	Etude Faisabilité (analyse fonctionnelle, besoin, procédé)	Etude base (architecture)	Etude de détail (études détaillées de réalisation, choix matériel et logiciel)
Evénement dangereux	Perte de production		
Système(s) affecté(s) :	Système de production		
Conséquence ¹⁻⁴ :	humain 1	mission 2	équipement 1
Description Effet :	Le système est victime de deux fois plus de déclenchements intempestifs ; Ceci conduit à une perte de disponibilité préjudiciable au point de vue perte de production		
Facteurs causals :	Défaut de l'architecture 2003 : ne pas être tolérante aux fautes des capteurs		

FIGURE 4.4 – Feuille d'identification et analyse de l'effet

L'évaluation de l'effet de cette double défaillance potentielle (par le champ "Conséquence") est évaluée par des experts selon une échelle de gravité (en terme de risque) propre à l'entreprise A, et définie pour tenir compte des pertes liées aux personnes, à la mission remplie par le système, et aux équipements composant le système (cf. Tableau 4.2).

Severity Level	General Description		
	Human	Mission	Equipment
4	Loss of Life	Mission abort or mission loss	System Loss
3	Severe injury or illness	Major mission objectives incomplete	Major System Damage
2	Minor injury or illness	Minor mission objectives incomplete	Minor System Damage
1	Less than minor injury or illness	All mission objectives completed	Less than minor System Damage

TABLE 4.2 – Une échelle de gravité personnalisée (en terme de risque)

Dans le cas où un déclenchement intempestif du système d'arrêt d'urgence a lieu, il n'y pas (ou peu) de pertes liées aux personnes et aux équipements. Cependant, ce déclenchement intempestif conduit à une perte de disponibilité préjudiciable du point de vue de la production, donc il a des conséquences sur l'objectif de la mission ; ces conséquences ont été jugées de niveau 2.

4.2.4 Réduction / exploitation des incertitudes

L'étape de réduction / exploitation des incertitudes se déroule en plusieurs phases : nous allons d'abord proposer des solutions envisageables pour la prévention des déclenchements intempestifs ; ensuite identifier les critères de sélection de solutions selon les exigences normatives ; enfin, choisir une solution architecturale par l'analyse quantitative de la fiabilité et de la disponibilité des différentes solutions proposées (y compris l'architecture proposée initialement).

4.2.4.1 Identification des solutions envisageables

Avec l'aide des experts de l'entreprise A, nous avons identifié deux solutions sur la base de la première architecture proposée, dans le but de

prévenir des déclenchements intempestifs.

Solution 1

Une première solution, destinée à assurer une indisponibilité minimum de l'exploitation, consiste à proposer l'architecture suivante (voir Figure 4.5) caractérisée par :

- L'ajout de redondances par le triplement des acquisitions sur les entrées critiques, donc la présence de 9 cartes d'acquisitions ;
- L'application d'un vote 2003 applicatif dans chaque unité de traitement.

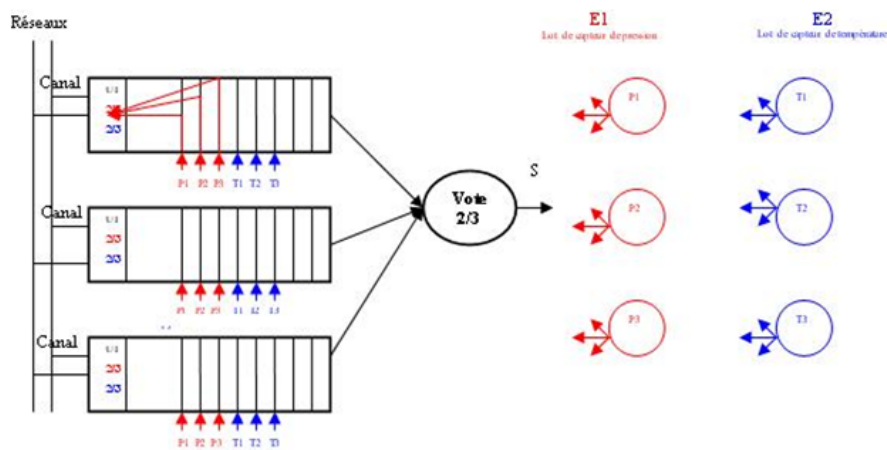


FIGURE 4.5 – Solution 1 : Tripler les acquisitions d'entrée critiques

Solution 2

Une deuxième solution consiste à échanger les acquisitions via des réseaux redondants, comme le montre la Figure 4.6. Elle nécessite également un vote 2003 applicatif (dans chaque unité de traitement).

Avec ces deux architectures, on peut tolérer au maximum une panne par lot de capteurs et donc améliorer fortement la disponibilité du sys-

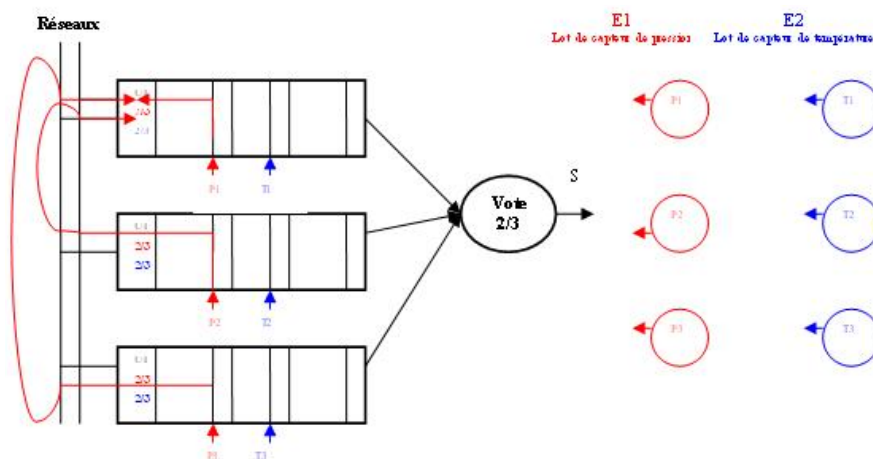


FIGURE 4.6 – Solution 2 : Echanger les acquisitions via les réseaux redondants

tème (cf. § 4.2.4.3).

Les deux sous-sections suivantes montrent comment on arrive au choix de la solution finale, parmi les deux solutions ci-dessus, en analysant les incertitudes intrinsèques du système sur la fiabilité et la disponibilité.

4.2.4.2 Identification des critères de sélection de solutions

Les critères pour le choix de la solution sont : la fiabilité, la disponibilité du système et le coût. Ce dernier inclut :

- un coût d'acquisition : c'est le coût habituel des parties fixes d'acquisition d'un module, y compris le coût de développement de celui-ci, la contribution aux travaux de recherche associés, les coûts d'acquisition des éléments de base du soutien comme la documentation, la formation du personnel d'exploitation et de maintenance, les lots initiaux de pièces de rechange, les équipements de test et de diagnostic ;
- un coût de maintenance : ce coût est partagé entre le coût de maintenance corrective et préventive.

4.2.4.3 Détermination du choix de solution

Pour pouvoir déterminer notre choix, nous allons d'abord calculer la fiabilité et l'indisponibilité du système selon les différentes architectures ; ensuite, nous baserons notre choix sur le résultat de ces calculs.

Analyse de la fiabilité et de l'indisponibilité du système

La norme CEI 61508 propose une technique de calcul de la probabilité de défaillance d'un dispositif de sécurité. Cette technique de calcul repose sur les Diagrammes de Fiabilité. La Figure 4.7 montre le diagramme de fiabilité de l'architecture 2003 proposée par cette norme.

En fonction de l'architecture définie par le diagramme de fiabilité, des équations permettent de calculer :

- t_{CE} : Temps moyen d'indisponibilité d'un module
- t_{GE} : Temps moyen d'indisponibilité d'un groupe (de deux modules)
- PFD_{avg} : Probabilité moyenne de défaillance à la sollicitation de déclenchement

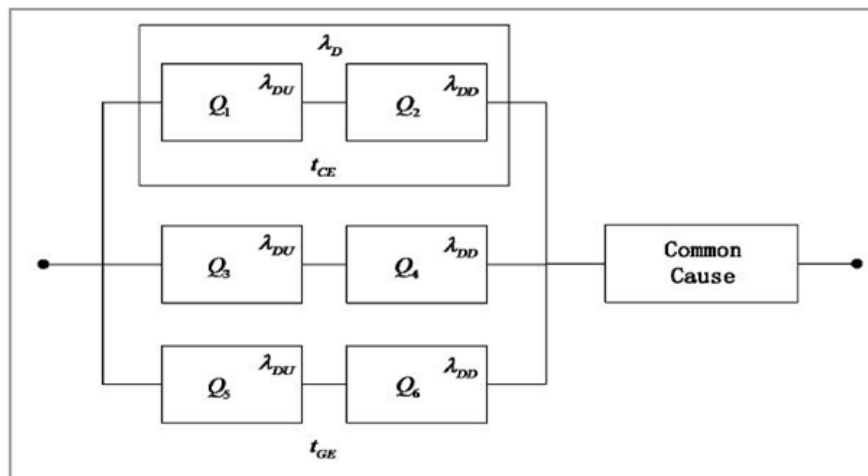


FIGURE 4.7 – Diagramme de Fiabilité de l'architecture 2003 (CEI61508 2003)

Le Tableau 4.3 montre les paramètres d'entrée qui sont pris en compte dans le calcul de ces trois indicateurs. Il montre également les signifi-

tions de ces paramètres, les formules et les ressources de données que nous avons utilisées pour ce calcul, ainsi que les valeurs approximatives des indicateurs obtenues.

Les valeurs des paramètres $\lambda_{D,DC}$, $MTTR$, β , β_D ont été obtenues par l'étude d'AMDEC (Analyse des Modes de Défaillance, de leurs Effets et de leurs Causes) effectuée par les ingénieurs de sûreté de fonctionnement de l'entreprise A ; nous supposons que le système fonctionne pendant 1 an pour déterminer la valeur de TI ; les formules de calcul sur λ_{DD} , λ_{DU} , t_{CE} , t_{GE} , PFD_{avg} sont proposées par la norme CEI 61508.

Le premier résultat approximatif que nous avons obtenu sur l'indicateur PFD_{avg} ($2.2299 * 10^{-5}$) démontre que la première architecture proposée par les experts de l'entreprise A possède un niveau de sécurité très important, puisque la probabilité d'une défaillance dangereuse satisfait à la contrainte normative : $10^{-4} < PFD_{avg} < 10^{-5}$.

Selon le même principe de calcul, le Tableau 4.4 montre les résultats de l'analyse quantitative de la fiabilité et l'indisponibilité des trois architectures.

Choix de la solution

D'après les résultats de notre calcul obtenus dans le Tableau 4, nous pouvons voir que :

- du point de vue de la fiabilité, les trois architectures répondent à la contrainte normative de $10^{-4} < PFD_{avg} < 10^{-5}$;
- du point de vue de la disponibilité, l'indisponibilité de la première architecture proposée est deux fois plus longue que les deux solutions proposées ;
- du point de vue du coût, il est évident que la première solution coûte plus cher que la deuxième solution, dans la mesure où elle

Paramètres	Signification	Formules de calcul et Ressources de données	Valeur approximative
Entrées	λ_D	Taux de défaillances dangereuses 1/MTBF Compte- rendu d'intervention et AMDEC (Analyse des Modes de Défaillance, de leurs Effets et de leurs Causes)	$5 \cdot 10^{-7}$
	λ_{DD}	Taux de défaillances dangereuses détectées $\lambda_{DD} = \lambda_D \cdot DC$	$4.5 \cdot 10^{-7}$
	λ_{DU}	Taux de défaillances dangereuses non détectées $\lambda_{DU} = \lambda_D \cdot \beta$	$0.5 \cdot 10^{-7}$
	T_I	Intervalle du test périodique 365 * 24 (Supposons que le système fonctionne pendant 1 an)	8760 h
	DC	Couverture du diagnostic des fautes dangereuses AMDEC (Analyse des Modes de Défaillance, de leurs Effets et de leurs Causes)	90%
	$MTTR$	Durée moyenne de rétablissement Compte- rendu d'intervention	8 h
	β	Fraction de défaillances détectées qui ont une cause commune AMDEC (Analyse des Modes de Défaillance, de leurs Effets et de leurs Causes)	10%
	β_D	Fraction de défaillances non détectées qui ont une cause commune AMDEC (Analyse des Modes de Défaillance, de leurs Effets et de leurs Causes)	5%
Sorties	r_{CE}	Temps moyen d'indisponibilité d'un module $\frac{\lambda_{DU}}{\lambda_D} \left(\frac{T_I}{2} + MTTR \right) + \frac{\lambda_{DD}}{\lambda_D} MTTR$	446h
	r_{GE}	Temps moyen d'indisponibilité d'un groupe (de deux modules) $\frac{\lambda_{DU}}{\lambda_D} \left(\frac{T_I}{3} + MTTR \right) + \frac{\lambda_{DD}}{\lambda_D} MTTR$	300 h
	$PF_{D\text{ avg}}$	Probabilité moyenne de défaillance $PF_{DG} = 6[(1 - \beta)\lambda_{DU} + (1 - \beta_D)\lambda_{DD}]^2 t_{CE} t_{GE} + \beta\lambda_{DU} \left(\frac{T_I}{2} + MTTR \right) + \beta_D\lambda_{DD}MTTR$	$2.2299 \cdot 10^{-7}$

TABLE 4.3 – Résultats de l'analyse quantitative de la fiabilité et de l'indisponibilité de la première architecture proposée

Paramètres	Valeur approximative	Première architecture proposée	Architecture de la solution 1	Architecture de la solution 2
λ_D (Taux de défaillances dangereuses)	$5.000 \cdot 10^{-7}$	$5.000 \cdot 10^{-7}$	$10.100 \cdot 10^{-7}$	$6.900 \cdot 10^{-7}$
λ_{DD} (Taux de défaillances dangereuses détectées)	$4.500 \cdot 10^{-7}$	$4.500 \cdot 10^{-7}$	$9.969 \cdot 10^{-7}$	$6.624 \cdot 10^{-7}$
λ_{DU} (Taux de défaillances dangereuses non détectées)	$0.500 \cdot 10^{-7}$	$0.500 \cdot 10^{-7}$	$0.404 \cdot 10^{-7}$	$0.276 \cdot 10^{-7}$
T_I (Intervalle du test périodique)	8760 h	8760 h	8760 h	8760 h
DC (Couverture du diagnostic des fautes dangereuses)	90%	90%	96%	96%
$MTTR$ (Durée moyenne de rétablissement)	8 h	8 h	8 h	8 h
β (Fraction de défaillances détectées qui ont une cause commune)	10%	10%	5%	7%
β_D (Fraction de défaillances non détectées qui ont une cause commune)	5%	5%	2.5%	3%
r_{CE} (Temps moyen d'indisponibilité d'un module)	446h	446h	183.2h	183.2h
r_{GE} (Temps moyen d'indisponibilité du group (sur deux modules)	300 h	300 h	124.8h	124.8h
$PF_{D\text{ avg}}$ (Probabilité moyenne de défaillance)	$2.2299 \cdot 10^{-7}$	$2.2299 \cdot 10^{-7}$	$0.9403 \cdot 10^{-5}$	$1.4761 \cdot 10^{-5}$

TABLE 4.4 – Résultats de l'analyse quantitative de la fiabilité et de l'indisponibilité des trois architectures

triple le nombre de cartes d'acquisition.

En conclusion, l'architecture de la deuxième solution est la solution la meilleure parmi les trois. Cette solution apporte une opportunité économique au projet de centrale électrique dans laquelle le système d'arrêt d'urgence sera implanté. Elle augmente la disponibilité du système en diminuant les déclenchements intempestifs, avec seulement un surcoût léger pour son acquisition et sa maintenance. Elle a été par conséquent retenue comme la nouvelle architecture du système d'arrêts d'urgence pour le projet concerné.

4.2.5 Représentation des incertitudes

Pour représenter les incertitudes, leurs effets et les traitements que nous avons identifiés, analysés, et traités dans les sections précédentes, nous appliquons le nouveau profil "Uncertainty-of" qui a été implémenté pour le diagramme de classe UML dans cette thèse.

La Figure 4.8, la Figure 4.9, la Figure 4.10 montrent les représentations de l'incertitude liée à l'architecture préliminaire du système d'arrêts d'urgence, ses effets et son traitement.

La structure de ces représentations est conforme à ce qui a été défini dans le profil "Uncertainty_of".

Dans la Figure 4.8, le paquetage "Design_process_uncertainties" est stéréotypé <<uncertainty_of_process>>. Ce stéréotype raffine le concept de paquetage d'UML en lui ajoutant des informations sur les incertitudes identifiées dans le processus de conception du système.

De la même manière, les incertitudes identifiées sur l'architecture du système sont représentées (dans cette même figure), par la classe "Preliminary_architecture_uncertainty" stéréotypée par <<uncertainty_of_deliverable>>. Ce stéréotype raffine le concept de classe d'UML en lui rajoutant quatre " tagged values

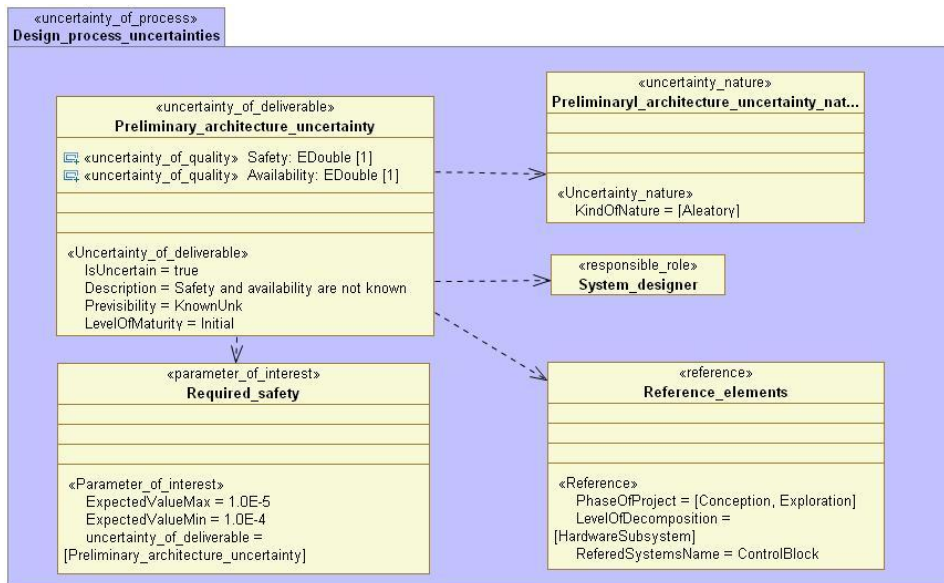


FIGURE 4.8 – Représentation des incertitudes liée à l'architecture préliminaire avec diagramme de classe UML

" comme compléments d'information : "LevelOfMaturity", "IsUncertain", "Previsibility" et "Description". En revanche, les attributs "Safety" et "Availability" sont tous deux des propriétés particulières de la classe "Preliminary_architecture_uncertainty". Stéréotypés par <<uncertainty_of_quality>>, ils représentent des incertitudes sur la qualité envisagés, et donc ajoutés, par le concepteur.

De façon plus détaillée, nous avons la classe "Preliminary_architecture_uncertainty_nature", stéréotypée par <<uncertainty_nature>>, représentant la nature de l'incertitude sur l'architecture. Par l'ajout de la classe "System_designer", stéréotypée par <<responsible_role>>, le concepteur indique qu'il est responsable de la gestion de cette incertitude. Pour ensuite décrire les éléments reliés à cette incertitude, la classe "Reference_elements", stéréotypée par <<reference>>, est utilisée. Cette incertitude indique que les phases de projets sont la conception et l'exploitation du systèmes au niveau "sous-matériel" et que le nom du système référé est "ControlBlock". La classe "Required_safety", stéréotypée <<parameter_of_interest>>, représente les valeurs espérées de la fiabilité de système.

La Figure 4.9 montre la représentation de l'effet de l'incertitude liée

à l'architecture préliminaire du système d'arrêts d'urgence. Dans cette figure, cet effet est représenté par la classe "System_breakoff_untimely" stéréotypée par «risk». Ce stéréotype inclut trois "tagged values" pour décrire l'évaluation de cet effet : "Probability", "Consequences" et "LevelOfrisk". Comme nous avons vu dans la section 4.2.3, l'analyse qualitative de cet effet est réalisée selon une échelle de gravité (en terme de risque) propre à l'entreprise A, et définie pour tenir compte des pertes liées aux personnes, à la mission remplie par le système, et aux équipements composant le système (cf. Tableau 4.2). Les résultats de cette analyse sont représentés par les classes "Analysis_1", "Analysis_2", et "Analysis_3" stéréotypées par «uncertainty_level_analysis» en les décrivant par deux "tagged values" : "EffectOn" et "EffectLevel". La classe "Reference_element", stéréotypée par «reference», est utilisée pour décrire les éléments liés à cet effet.

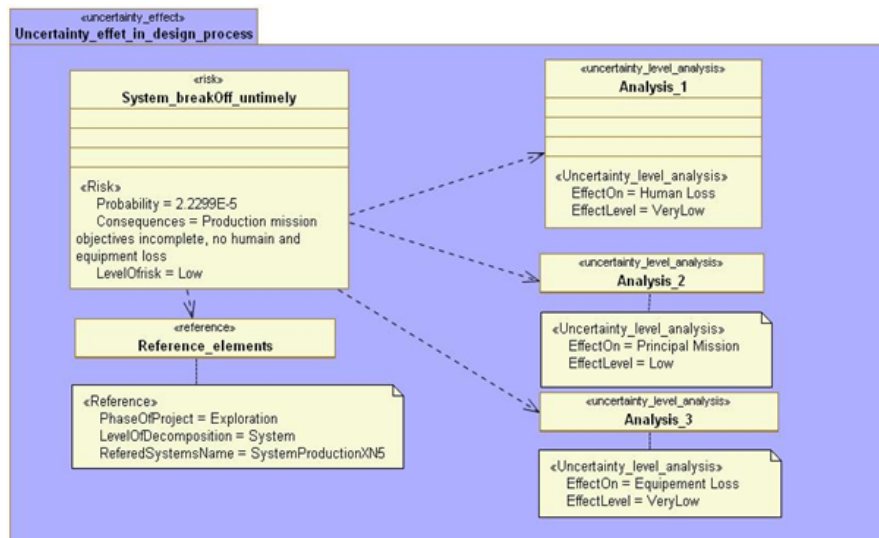


FIGURE 4.9 – Représentation de l'analyse des effets de l'incertitude liée à l'architecture préliminaire avec diagramme de classe UML

La Figure 4.10 montre la représentation des traitements de l'incertitude liée à l'architecture préliminaire du système d'arrêts d'urgence. Comme nous avons vu dans la section 4.2.4, deux solutions (cf. Figure 4.5 et 4.6) ont été proposées pour améliorer la disponibilité et la fiabilité du système.

Dans cette figure, elles sont représentées par la classe "Solution_1" stéréotypée par <<risk_treatment>> et la classe "Solution_2" stéréotypée par <<opportunity_treatment>>. Ces deux stéréotypes incluent les mêmes "tagged values" comme compléments d'information : "TreatmentDescription", "ReferredCharacteristic" et "KindOfTreatment". Le résultat du choix des solutions est représenté en commentaire liée à la "Solution_2" qui a été choisie dans notre cas.

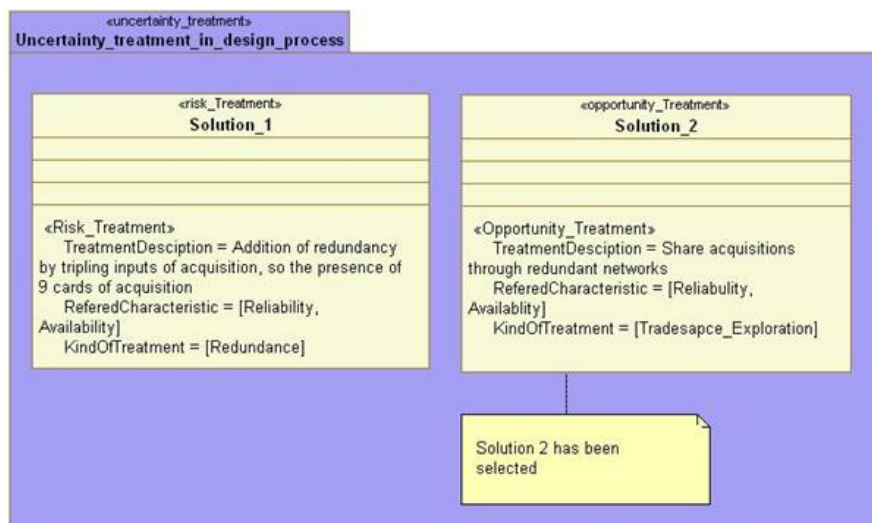


FIGURE 4.10 – Représentation des traitements de l'incertitude liée à l'architecture préliminaire avec diagramme de classe UML

4.3 TRAITEMENT DES INCERTITUDES SUR LA DURÉE ET LE COÛT DES ACTIVITÉS LIÉES AU PROCESSUS DE DÉFINITION DU SYSTÈME

Cette section consiste à traiter les incertitudes sur la durée des activités liées au processus de définition du système dans ce projet. Ces incertitudes intrinsèques sur les activités des processus provoquent en effet des influences sur la planification du projet. L'objectif de notre étude sur ces incertitudes a été donc d'estimer la probabilité d'atteindre l'objectif du projet, plus précisément dans le délai de 119.2 jours prévu pour le processus de définition.

Dans cette section, nous allons d'abord décrire en détail les activités dans le processus de définition préliminaire du système (cf. §4.3.1) ; puis nous appliquerons directement l'étape d'analyse des effets des incertitudes de notre méthodologie, ce qui sera suffisant pour répondre à notre objectif (cf. §4.3.2).

4.3.1 Introduction : les activités liées au processus de définition préliminaire du système

Le processus de définition préliminaire du système de l'entreprise A entremêle le processus de spécification et de conception préliminaire du système. Nous listons neuf activités concourant au processus de définition préliminaire :

1. Analyser les exigences de la définition du système
2. Spécifier les fonctions, les interfaces, les modes de fonctionnement et les contraintes de réalisation du système
3. Organiser et préparer la validation du système.
4. Créer l'architecture de conception préliminaire du système
5. Etudier l'incertitude sur la conception préliminaire du système d'un point de vue technique (sécurité et disponibilité)
6. Décrire et vérifier l'architecture retenue d'un point de vue technique (faisabilité, performances, objectifs qualité) ou budgétaire (coûts et/ou délais) et finaliser les éléments de la définition du système
7. Organiser et préparer l'intégration du système
8. Organiser le projet : estimation et planification initiales du développement du système, suivi d'avancement de la phase de définition du système, gestion des configurations, assurance qualité
9. Vérifier les éléments de sortie du processus de définition du système

Le diagramme DSM de la Figure 4.11 matérialise le séquençage et les interactions de ces activités pendant le processus. Ainsi, le début du processus est l'activité d'analyse des exigences (représentée par A1) qui est un préalable indispensable à la spécification (A2) et à l'organisation du

projet (A8) ; la spécification du système conditionne le démarrage de l'activité de conception préliminaire (A4) ; etc. Ces précédences temporelles sont représentées par des cases marquées d'une croix dans la matrice.

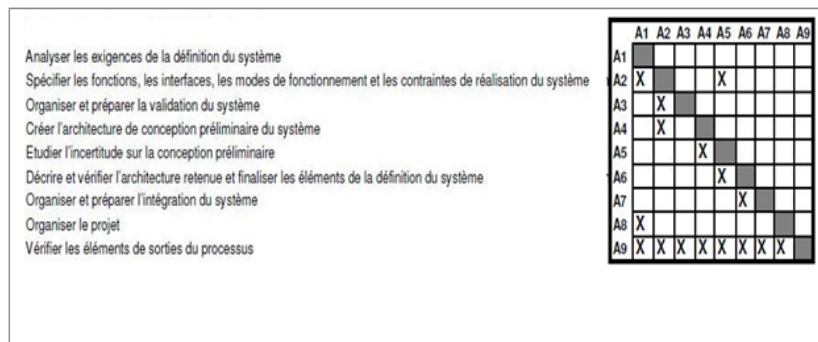


FIGURE 4.11 – Diagramme DSM des activités du processus de définition préliminaire du système

Regardons de plus près l'étude de l'incertitude (A5) : cette activité peut amener une reprise de la spécification du système en cas de non-conformité aux exigences normatives d'un point de vue technique (sécurité et disponibilité), qui sera représentée par une itération de l'activité 5 vers l'activité 2 dans le graphe (croix dans A25). La vérification opérée sur les éléments de sortie du processus attend, par contre, la fin de toutes les autres activités.

4.3.2 Analyse des effets des incertitudes

Nous allons analyser les effets des incertitudes sur la durée des activités en calculant les distributions des probabilités de la durée du processus. Nous réaliserons cette analyse par une simulation en utilisant la technique de Monte-Carlo (Owen 1995).

La densité de probabilité est souvent utilisée pour représenter l'incertitude sur la durée d'activité (Pritsker et al. 1997) (Williams) Browning et Eppinger (1998). Dans notre cas, cette fonction dépend de trois paramètres

pour chaque activité : la durée dans le cas plus optimiste BCV (Best Case Value), le cas standard MLV (Most Likely Value) et le cas pessimiste WCV (Worst Case Value) (cf. §2.3.3.2 du chapitre 2). Le Tableau 4.5 liste les données sur la durée de chaque activité de ces trois points de vue, qui sont obtenues par des interviews avec les parties prenantes du projet.

Notons que, pour l'activité A2, nous avons déjà ajouté la durée de sa reprise due à la modification de la conception préliminaire, en supposant qu'une seule itération a eu lieu (cf. l'étude de l'incertitude sur l'architecture du système dans la section précédente).

Le choix de la forme de distribution des probabilités peut se faire a priori parmi trois types de distribution : uniforme, gaussienne et triangulaire. Cependant, la distribution uniforme ne permet pas de prendre en compte la valeur de MLV de l'activité, et la distribution gaussienne exige la symétrie entre la valeur de BLV et WLW, ce qui ne correspond pas à la réalité ; nous avons donc décidé de prendre une distribution triangulaire des probabilités pour notre simulation.

Activité		Durée (jours)		
ID	Nom	BCV	MLV	WCV
1	Analyser les exigences	7	8.5	10
2	Spécifier le système	22	30	38
3	Organiser et préparer la validation du système	7.7	10.5	13.3
4	Créer l'architecture de conception préliminaire du système	7.7	10.5	13.3
5	Étudier l'incertitude sur la conception préliminaire	11	15	19
6	Décrire et vérifier l'architecture retenue	16	19	22
7	Organiser et préparer l'intégration du système	15	17.5	20
8	Organiser le projet	30	35	40
9	Vérifier les éléments de sortie du processus	15	18.5	22

TABLE 4.5 – Durées des activités de définition préliminaire du système

La Figure 4.12 résume les données d'entrée, les variables, la sortie et l'algorithme du modèle de simulation que nous avons utilisé pour calculer les distributions de la durée du processus de définition. La séquence des activités (cf. la matrice DSM de la Figure 4.11), la gamme de durée pour

chaque activité (cf. Tableau 4.5) et les densités de probabilité d'activité que nous avons obtenues précédemment sont les entrées indispensables pour ce modèle de simulation.

Chaque série de simulation initialise les durées écoulées des activités à 0, et leur donne un temps aléatoire d'exécution respectant les densités de probabilité choisies. Les activités sont exécutées selon la séquence prédéfinie dans le diagramme DSM. A1 est la première activité qui est exécutée. La fin d'exécution de chaque activité est conservée dans le vecteur "*executedActn*". La date de fin de l'exécution de l'activité *i* est égale à la somme de la date de fin d'exécution la plus grande des activités qui l'ont précédée, augmentée de la durée de l'activité *i* elle-même. Une fois que toutes les activités ont pris fin, la durée cumulée du processus est la date de fin d'exécution plus grande parmi toutes les activités.

Comme la simulation est exécutée à plusieurs reprises (10000 fois dans notre cas), un certain nombre de durées aléatoires sont générées. Les résultats forment une distribution des durées, comme le montre dans la Figure 4.13. La simulation montre que le processus de définition du système possède à peu près 70% de chances de finir dans la durée prévue de 119.2 jours (Target =119.2 jours).

4.4 CONCLUSIONS DU CHAPITRE

Ce chapitre a détaillé un cas d'étude sur le traitement des incertitudes dans le processus de conception d'un système d'arrêt d'urgence. Nous avons choisi de traiter deux problèmes de l'incertitude dans ce cas d'étude :

- les incertitudes sur le livrable du processus : ce sont les incertitudes sur l'architecture du système, par rapport à ses conséquences sur la sécurité et la disponibilité du système ;

Données : DSM_{ij} → séquençement des activités (cf. Figure 9)
 $Act_i (BCV, MLV, WLV)$ → paramètres de durée pour chaque activité (cf. Tableau 5)
 $PDF(Probability Density Functions)$ → densité de probabilité pour la durée de chaque activité, qui est fonction de BCV, MLV, WLV
 $Target$ → représente l'objectif de durée du processus (119.2 jour prévus)

Variables : $Ndrawing$ → nombre de tirages effectués
 $Nactivities$ → nombre d'activités considérées
 $ActS_n$ → tableau de longueur n contenant une durée aléatoire pour chaque activité
 $executedAct_n$ → tableau des dates de fin d'exécution des activités

Sortie : $processDur$ → durée cumulative du processus pour une simulation

Algorithme :

```

Pour tirage de 1 à  $Ndrawing$ 
  Pour  $i$  de 1 à  $Nactivities$ 
     $executedAct(i) \leftarrow 0$ ; /* Aucune activité n'a été encore exécutée */
  Fin pour

   $ActS_n \stackrel{iid}{\sim} PDF[BCV, MLV, WCV]$ ; /* on tire au hasard la durée de chaque activité de façon
  indépendante et identiquement distribuées (iid) */
   $executedAct(1) \leftarrow ActS_n(1)$ ; /* l'activité A1 est exécutée */
  Tant que  $\exists i \text{ } executedAct(i) = 0$  /* Tant qu'il reste une activité non-exécutée */
    Pour chaque activité  $i$  /* pour l'activité courante  $i$  */
       $Pred$  = liste des activités précédant l'activité  $i$ 
      Si toutes les activités précédant l'activité  $i$  ont été exécutées
         $executedAct(i) = \max_{j \in Pred} (executedAct(j)) + ActS_n(i)$ ; /* la date de fin d'exécution de
        l'activité  $i$  = la date de fin
        d'exécution la plus tardive des
        activités précédentes + la durée de
        l'activité  $i$  */
      Fin si
    Fin pour
  Fin Tant

   $processDur(\text{tirage}) = \max_{\forall i} (executedAct(i))$ ;
Fin pour

```

FIGURE 4.12 – Algorithme du modèle de simulation de durée du processus de définition

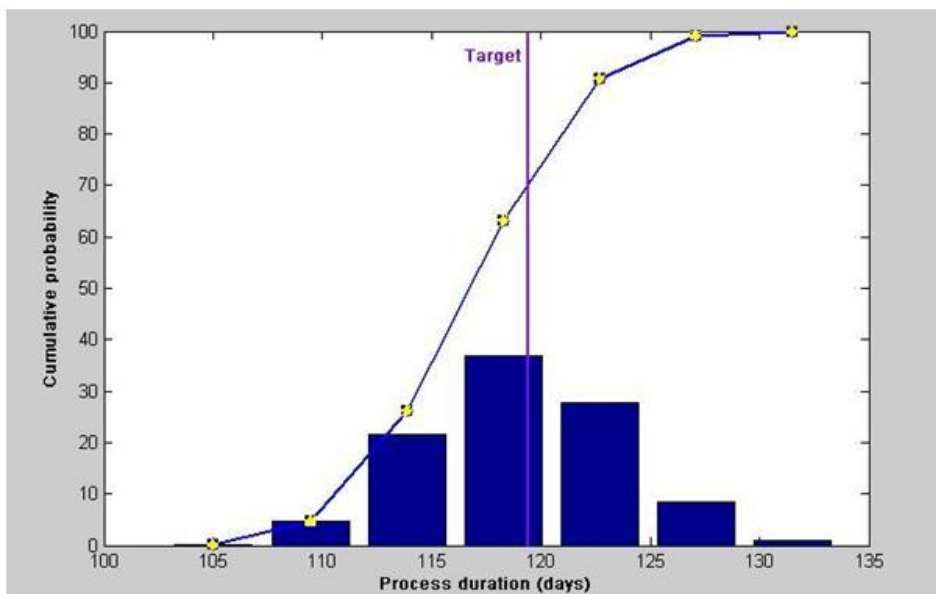


FIGURE 4.13 – Algorithme du modèle de simulation de durée du processus de définition

- les incertitudes sur les activités du processus : ce sont les incertitudes sur la durée des activités du processus dans ce projet.

Nous cherchions, à travers ce cas d'étude, à :

- montrer comment la méthodologie de gestion des incertitudes et le profil UML "Uncertainty-of" peuvent être appliqués dans un projet industriel,
- identifier les limites de notre méthodologie et des outils UML proposés.

Quatre étapes de la méthodologie de gestion des incertitudes proposée dans le chapitre 3 ont été appliquées pour traiter les incertitudes sur l'architecture du système : l'identification et la sélection, l'analyse des effets, la réduction et /ou l'exploitation, et la représentation des incertitudes. Dans cette application, nous avons capturé les informations pertinentes sur les incertitudes, leurs effets et leurs traitements sur les éléments du processus étudié, ce qui peut permettre de guider les efforts des ingénieurs dans le développement de processus. En plus, les résultats obtenus (cf. §4.2.4, sur les deux solutions pour l'architecture 2003) montrent également qu'une incertitude peut être gérée de façon à réduire les risques et à exploiter les opportunités.

Dans le traitement des incertitudes liées aux activités du processus, l'étape d'analyse des effets des incertitudes a été la seule étape concernée. Nous avons notamment appliqué une simulation en utilisant la technique de Monte-Carlo pour calculer les distributions des probabilités de la durée du projet. Le modèle de simulation que nous avons établi dans ce cas peut être amélioré en prenant en compte la probabilité de la reprise d'une activité dans nos futurs travaux.

Le déroulement des traitements des deux incertitudes montrent comment notre méthodologie peut être appliquée dans un projet industriel, et souligne que son application dépend du problème concret d'incertitude et des besoins réels du projet. Avec les résultats obtenus dans notre cas d'étude, les concepteurs du système ont trouvé que la définition d'une

telle méthodologie et le profil UML "Uncertainty-of" étaient utiles pour l'organisation de la gestion des incertitudes dans la conception du système, et que l'application de cette méthodologie sur une longue période aidera significativement la constitution d'une base de connaissance des incertitudes et la gestion des projets.

L'étude de ce cas nous a permis également de trouver des limites sur la méthodologie et le profil eux-mêmes, sur leur construction et leur utilisation ; par exemple, la construction de notre méthodologie reste dans le cadre d'un projet et d'une entreprise, et les incertitudes sur les ressources partagées lors de projets multiples n'ont été pas prises en compte. La limite à laquelle nous nous sommes heurtés dans l'utilisation du profil est que, pour le moment, les implémentations du profil restent dans le cadre des diagrammes de classe ; etc. Nous allons discuter ces limites dans le chapitre 5 en répondant aux questions soulevées dans le premier chapitre de cette thèse.

CONCLUSIONS ET PERSPECTIVES

5

5.1 RÉSUMÉ SUR LES QUESTIONS DE RECHERCHE ET NOS CONTRIBUTIONS

Le contenu de ce mémoire adresse deux objets de recherche principaux, la gestion des incertitudes dans le processus de développement de systèmes complexes et leur modélisation. Nous avons soulevé en introduction six questions relatives à ces domaines, qui ont conduit nos recherches. Après avoir développé dans les précédents chapitres ces questions, nous les reprenons ici en conclusion et résumons pour chacune les résultats auxquels nous avons abouti, synthétisant ainsi les contributions principales de cette thèse.

5.1.1 Gestion des incertitudes dans le processus de développement de systèmes complexes

La première question de recherche à propos de ce sujet était :

- **Q1 : Quelles sont les méthodes qui ont été proposées pour la gestion des incertitudes ? Quelles sont leurs forces et leurs faiblesses ?**

La littérature a révélé que les méthodes de modélisation des incertitudes dans le cadre du processus de développement de systèmes s'appuient en général sur l'analyse des caractéristiques de processus. Ces caractéristiques comprennent notamment le coût et le délai des processus ainsi que leurs risques.

Dans le chapitre 2 (cf. §2.2.2 et §2.3.3), nous avons discuté des méthodes comme PERT, CPM, Gantt, ou DSM qui permettent de mieux gérer la complexité des systèmes ou des projets, et fournissent des outillages intéressants surtout pour la planification de projet. Nous avons également cité certains outils et techniques pour l'analyse qualitative et quantitative des risques inhérents aux processus.

Cependant, ces méthodes sont très tournées vers la modélisation des incertitudes liées aux coûts et délais, mais elles ignorent l'existence des incertitudes liées aux autres éléments de processus, comme les livrables ou l'humain. En plus, elles sont peu applicables au traitement des incertitudes imprévisibles de type "inconnues inconnues", parce qu'elles supposent que l'ensemble des activités des processus sont prévisibles, ce qui n'est pas toujours vrai dans la réalité.

De nombreuses études rétrospectives, nous avons retenu que l'incertitude doit plutôt être traitée comme un élément essentiel pour la conception et le développement de systèmes complexes. Les actions sur cet élément ne doivent plus se limiter à l'analyse des caractéristiques de processus, mais s'étendre à la gestion des incertitudes. Cela signifie surtout une gestion proactive : il s'agit de prévoir les incertitudes au plus tôt des processus de développement de systèmes.

Nous avons également retenu que le risque est l'effet de l'incertitude le plus fréquemment considéré et évalué dans un projet, cependant l'effet positif de l'incertitude, c'est-à-dire l'existence d'opportunités, doit être également prise en compte dans la gestion des incertitudes.

La question de recherche suivante était :

- **Q2 : Comment gérer les incertitudes dès le début du processus de développement de systèmes ?**

La définition d'une méthodologie de gestion des incertitudes (cf. section 3.1 du chapitre 3) dans le processus de développement est au coeur de nos travaux. Cette méthodologie, qui respecte les principes de traite-

ment des incertitudes (cf.§2.3.2 du chapitre 2) offre les intérêts suivants :

- elle permet de recueillir les connaissances des incertitudes liées aux processus de DS complexes,
- elle permet l’analyse des effets potentiels découlant de ces incertitudes,
- la mise en oeuvre de moyens de réduction des risques et d’exploitation des opportunités,
- la surveillance des incertitudes traitées et le contrôle des incertitudes nouvellement introduites.

Cette méthodologie comprend les quatre étapes séquentielles suivantes, ainsi qu’une activité menée en parallèle :

- Etape 1 : identification des incertitudes
- Etape 2 : analyse de leurs effets
- Etape 3 : réduction et /ou exploitation des incertitudes
- Etape 4 : validation de solution et contrôle des incertitudes
- Activité en parallèle : représentation des incertitudes

Par la définition de cette méthodologie, nous avons montré qu’il était possible d’organiser les activités de gestion des incertitudes. Grâce à elle, nous parvenons à atteindre l’objectif global de notre travail : l’amélioration de la flexibilité et de la prédictibilité du processus de développement de systèmes.

- **Q3 : La procédure de gestion des incertitudes proposée dans cette thèse est-elle pertinente? Quelles sont ses qualités et ses faiblesses?**

La validité de notre méthodologie a été testée par l’étude d’un cas industriel. Grâce à cette étude, nous avons capturé les informations pertinentes sur les incertitudes et leurs effets sur les éléments du processus étudié, ce qui peut permettre de guider les efforts des ingénieurs dans le développement de processus. En outre, les résultats obtenus (cf. §4.2.4 du chapitre 4, sur les deux solutions pour l’architecture 2003) montrent éga-

lement qu'une incertitude peut être gérée de façon à réduire les risques et à exploiter les opportunités.

Cependant, notre méthodologie comporte plusieurs limitations : verticalement, si on reste dans le cadre du processus de développement,

- du point de vue de l'hypothèse : la méthodologie est applicable lors d'une phase de définition suffisamment précoce (par exemple, la phase d'étude de faisabilité) avec des données historiques sur un système existant dont le fonctionnement est similaire à celui qui va être développé. Cela veut dire que nous n'avons pas pu gérer les incertitudes avant la phase de définition. Par exemple, la maturité des exigences est une incertitude importante qui peut allonger ou raccourcir le délai du processus de conception ;
- du point de vue des projets multiples : la définition de notre méthodologie est valide essentiellement dans le cadre d'un projet et d'une entreprise, car les incertitudes sur les interfaces des projets n'ont pas été observées (par exemple, les incertitudes sur les ressources partagées entre les projets) ;
- du point de vue de la validité de la méthodologie : nous avons validé notre méthodologie avec une seule étude de cas. D'autres études devraient être menées, sur des exemples qualitativement différents, dans nos recherches futures, dans le but de corroborer ou d'affiner les résultats obtenus.

Horizontalement, si nous sortons du cadre du processus de développement,

- du point de vue de l'environnement de l'entreprise : les incertitudes externes ne sont pas prises en compte, comme par exemple les incertitudes sur le marché, les incertitudes environnementales, etc ; pourtant ces incertitudes ont souvent des impacts importants sur le coût, le délai et l'organisation du processus.

En conclusion, la gestion des incertitudes est un exercice difficile et pourtant essentiel pour le développement de systèmes complexes à long terme. L'état de l'art sur les méthodes existantes dans ce domaine a permis de clarifier et de proposer une classification des incertitudes et des moyens existants pour les identifier et de les traiter. En même temps, la méthodologie proposée offre une fondation solide pour les futurs travaux.

5.1.2 Méta-modélisation des incertitudes

Pour arriver à mieux gérer et représenter les incertitudes dans le processus de développement de systèmes, nous avons proposé une méta-modélisation des incertitudes. La première question de recherche à propos de ce sujet était :

- **Q4 Quelles sont les incertitudes associées au processus de DS et leurs catégorisations ?**

Les incertitudes associées au processus de DS sont identifiées par une catégorisation des incertitudes qui permet de raffiner les incertitudes dans le contexte étudié. Nous pensons que la classification traditionnelle des incertitudes est utile pour l'analyse de l'imperfection de l'information (cf. §2.1.5.1), mais qu'elle ne suffit pas pour organiser la gestion des incertitudes et représenter leur structure dans les processus de développement de systèmes.

Nous avons donc proposé une autre catégorisation des incertitudes en nous limitant au cas le plus fréquent, celui où l'imprécision peut être considérée comme une source de l'incertitude. L'origine de cette catégorisation est basée sur les deux éléments fondamentaux des processus : l'activité et le livrable, en prenant compte les incertitudes liées à la manière dont les individus (ou les organisations) agissent dans l'exécution du processus. La catégorisation proposée comprend donc trois catégories : incertitudes sur les activités, incertitudes sur les livrables, incertitudes sur le comportement de processus.

- **Q5 : Est-ce qu'il est possible de modéliser les incertitudes avec**

UML? Quels sont les avantages et les inconvénients ?

La réponse à cette question est oui. Nous avons finalement choisi UML, plutôt que SPEM comme le langage de notre modélisation (cf. § 3.4.1 du chapitre 3). Les avantages de la modélisation des incertitudes avec UML sont en effet :

- Premièrement, UML peut permettre la modélisation des incertitudes grâce à ses mécanismes d’extension. L’utilisation des trois concepts de stéréotypes, de valeurs marquées, et de contraintes permet de s’adapter aux besoins de la description de l’incertitude dans le cadre du développement de systèmes.
- Deuxièmement, UML permet de représenter les incertitudes graphiquement et de clarifier leur présence dans le processus de développement, et facilite donc leur compréhension par les ingénieurs.
- Troisièmement, UML permet de décrire non seulement les incertitudes liées aux processus, mais également celles liées au produit et aux exigences, et donc de garder la cohérence entre les modèles des incertitudes de processus, de produit, et d’exigences.

Cependant, l’inconvénient est que cette modélisation ne permet pas d’effectuer directement des analyses qualitatives et quantitatives sur les incertitudes. Pour arriver à ce stade, une étape supplémentaire, complexe, de transformation des modèles UML en modèles analytiques, serait nécessaire.

- **Q6 : Si la réponse à Q5 est oui, comment modéliser les incertitudes avec UML ? La création d’un nouveau profil UML apporte-t-elle une solution ? Quelles sont les limites auxquelles on se heurtera ?**

Pour pouvoir représenter les incertitudes avec UML, nous avons proposé un workflow de la modélisation de trois étapes, en nous basant sur le mécanisme d’extension d’UML (cf. §3.4.2 du chapitre 3) :

1. l'établissement du méta-modèle des incertitudes "Uncertainty of", qui permet d'identifier et de représenter les concepts principaux du domaine de gestion des incertitudes et les relations entre ces concepts ;
2. la création d'un nouveau profil UML à partir des classes prédéfinies dans le méta-modèle UML 2.0, qui permet de rendre notre méta-modèle opérable ;
3. la modélisation des descriptions des incertitudes par des diagrammes UML à partir des éléments fournis par le nouveau profil.

Nous avons appliqué le profil proposé dans le chapitre 4 (cf. son §4.2.5), et vérifié que la création d'un nouveau profil UML apporte une solution pertinente dans la représentation des incertitudes. La limite à laquelle nous nous sommes heurtés est que, pour le moment, les implémentations du profil restent dans le cadre des diagrammes de classe.

5.1.3 **Résumé de nos contributions du point de vue des processus de développement de systèmes complexes**

En définissant une méthodologie de la gestion des incertitudes et en proposant une méta-modélisation des incertitudes, nous pensons avoir contribué d'une part à la mise en oeuvre industrielle de l'Ingénierie des Systèmes et, d'autre part, aux avancées scientifiques dans ce domaine.

D'un point de vue industriel, ce travail permet d'aider un chef de projet et ses ingénieurs dans la mise en place d'une démarche de gestion des incertitudes sur son projet. Nous leur permettons de profiter des avantages d'une telle démarche en les aidant dans les problèmes concrets qu'ils rencontrent.

La méthodologie proposée, conformément aux principes du traitement des incertitudes (cf. §3.3.2 du chapitre 3), couvre les phases de l'ensemble d'un projet de développement de systèmes complexes et aide à représenter, analyser et traiter les incertitudes. Le méta-modèle des incertitudes proposé aide à s'appropriier et à comprendre les concepts qui les sous-tendent. Ces concepts sont réutilisables sur des projets similaires

et extensibles pour la modélisation des incertitudes liées aux produits et exigences. Le nouveau profil UML "Uncertainty-of" permet de modéliser les incertitudes par des diagrammes UML.

L'application pratique de la méthodologie est présentée dans le chapitre 4. Visant la mise en oeuvre d'une gestion des incertitudes qui conduise à mieux maîtriser les objectifs des projets (coût, délais, qualité...), elle devrait être utile aux ingénieurs dans leur démarche de développement de systèmes complexes.

D'un point de vue scientifique, nos travaux pourront servir de base à une meilleure compréhension des incertitudes au cours du cycle de vie du produit et du projet, et donc à une meilleure modélisation de leur gestion. En répondant aux six questions de recherche que nous venons de voir, nous avons obtenu :

- un moyen d'organiser des activités de gestion des incertitudes dans la phase de conception d'un projet ;
- une démarche de modélisation des incertitudes avec le langage UML ;
- un méta-modèle des incertitudes dans le cadre des processus de développement ;
- un nouveau profil UML "Uncertainty-of " qui peut rendre ce méta-modèle opérable avec les outils UML.

La méthodologie et la méta-modélisation des incertitudes proposées sont à notre avis les résultats les plus importants de notre thèse, que nous considérons comme une première étape pour des recherches futures.

5.2 PROPOSITIONS DE DIRECTIONS DE RECHERCHES FUTURES

Les enseignements tirés de cette thèse ont ouvert la porte à davantage de recherches possibles. La méthodologie et la méta-modélisation des incertitudes qui ont été proposées sont les fruits d'une première étape dans la recherche des pistes de recherches futures, qui pourraient renforcer, améliorer et étendre notre étude. L'évolution future et le raffinement de

nos propositions ont été encouragés par nos interlocuteurs tant académiques qu'industriels. Un résumé est présenté ci-dessous :

- La représentation des incertitudes a été spécifiquement étudiée dans le cadre des processus. L'extension de cette représentation aux incertitudes liées au produit et aux exigences nécessite d'être identifiée et raffinée plus profondément.
- Les implémentations actuelles du nouveau profil proposé restent dans le cadre du diagramme de classe UML. La représentation des incertitudes avec d'autres diagrammes UML, comme les diagrammes d'activité ou les machines à états, devrait être aussi implémentée.
- Certaines des techniques de l'analyse et de l'évaluation des incertitudes n'ont été que brièvement abordées dans le chapitre 2. Le défi à relever dans l'application de ces techniques dans la conception et le développement de systèmes est complexe : le nombre d'incertitudes qui sont impliquées, la diversité de ces types d'incertitude (à savoir, en continu, discret, de choix discret parmi plusieurs alternatives), l'interaction entre les incertitudes, la complexité de l'espace de conception et des modèles utilisés pour représenter cet espace de conception, etc., sont autant de facteurs à prendre en compte dans une étude plus complète.
- L'application de la méthodologie proposée pour la démonstration de sa viabilité pratique est présentée dans le chapitre 4. Nous avons traité deux incertitudes : l'incertitude liée à l'architecture de système et l'incertitude liée aux activités de processus, qui pour le moment sont représentatifs d'un processus de conception de systèmes complexes. Cette méthodologie devrait donc être intéressante pour plupart des ingénieurs, mais d'autres applications avec des incertitudes plus complexes seront l'occasion d'étendre et d'affiner la méthodologie proposée ; on peut raisonnablement supposer que les implémentations futures permettront de découvrir des leçons importantes.

BIBLIOGRAPHIE

- C.H. Antunes et L.C. Dias. Managing uncertainty in decision support models. *European Journal of Operational Research*, 181(3) :1425–1426, September 2007.
- V. Aranega. Démarche d'élaboration de profils uml à partir d'un méta-modèle métier. Rapport technique, IRIT, France, 2008.
- C. Argyris et D. Schön. *Organizational learning : A theory of action perspective*. Addison Wesley, USA, 1978.
- S. Austin, A. Baldwin, B. Li, et P. Waskett. Integrating design in the project process. Dans *International Civil Engineering*, pages 177–182, 2000.
- J.M. Ben Amor, S. et Martel. Le choix d'un langage de modélisation des imperfections de l'information en aide à la décision. Dans *Conference of the Administrative Sciences Association of Canada*, volume Vol.25, 2004.
- B.S. Blanchard. *Systems Engineering Management*. John Wileys and Sons, New York, 1965.
- P. Blancho et J. Durand. *Surêté De Fonctionnement et Matrise des risques*. Centre Technique Des Industries Mécaniques, France, 2004.
- P. Bonissone et R.M. Tong. Reasoning with uncertainty in expert systems. *International journal of Man Machine Studies*, Vol.22 :241–250, 1985.
- P. Bosc, D. Dubois, et H. Prade. *Uncertainty Management in Information Systems : From needs to solutions*. Kluwer Academic Publishers.
- B. Bouchon et B. Meunier. *La logique floue et ses applications*. Addison-Wesley, France, 1995.
- T.R. Browning. Process modeling in a system engineering context. Dans *Eleventh Annual INCOSE International Symposium*, 2001. <http://www.incose.org/symp2001/archive/program/tutorial/abstracts/F3.html>.

- T.R. Browning, E Fricke, et H. Negele. Key concepts in modeling product development processes. *Published online in Wiley InterScience, Systems Engineering*, Vol.9(2), 2006.
- Tyson R. Browning et Steven D. Eppinger. Modeling the impact of process architecture on cost and schedule risk in product development. Working papers, Massachusetts Institute of Technology (MIT), Sloan School of Management, 1998. <http://ideas.repec.org/p/mit/sloanp/2739.html>.
- CEI61508. *Norme CEI 61508 : Sécurité fonctionnelle des systèmes Électrique/Électroniques/Électroniques Programmable relatifs à la sécurité*. CE-NELEC, 2003.
- M. Chantal, J. Hugues, B. Leblanc, et O. Hugues. *Processus Métiers et systèmes d'information : Evaluation, modélisation, mise en oeuvre*. InfoPro., Dunod, 2005.
- H. chestnut. *Systems Engineering Tools*. John Wileys and Sons, New York, 1965.
- W. B. Chew, D. Leonard-Barton, et R. E. Bohn. Beating murphy's law. *Sloan Management Review*, vol.32(1), 1991.
- M. Cohen, J.Y. Jaffray, et T. Said. Individual behavior under risk and under uncertainty : an experimental study. *Theory and decision*, vol.18(2), 1985.
- R. De Neufville. *Applied Systems Analysis : Engineering Planning and Technology Management*. McGraw-Hill,Inc, New York, 1990.
- R. De Neufville. Uncertainty management for engineering systems planning and design, engineering systems monograph. Dans *Conference of Uncertainty Management for Engineering Systems, MIT USA*, 2004.
- E. De Rocquigny, N. Devictor, et S. Tarantola. *Uncertainty in Industrial Practice : A Guide to Quantitative Uncertainty Management*. Wiley Blackwell, France, 2008.
- EIA. *EIA-632 : Processes for Engineering a System*. Electronic Industries Alliance, 1999.

- J. Faucher. *Pratique De L'amdec - Assurez La Qualité Et La Sûreté De Fonctionnement De Vos Produits, Équipements Et Procédés*. Dunod, France, 2004.
- E. Fricke, A. Schulz, P. Wehlitz, et H. Negele. A generic approach to implement information based systems development. Dans *Tenth Annual International Symposium of the International Council On Systems Engineering (INCOSE)*, Minneapolis, 2000.
- P. Gaufilllet. Topcased : an eclipse based development environment for critical systems and softwares. Dans *Eclipse Summit Conference*, Germany, 2006.
- S. Grey. *Practical Risk Assessment for Project Management*. John Wileys and Sons, Inc, New York, 1995.
- M. Hammer. Seven insights about processes. Dans *Strategic Power Process Ensuring Survival Creating Competitive Advantage*, Boston, 2001.
- G.A. Hazelrigg. On the role and use of mathematical models in engineering design. *Journal of Mechanical Design*, Vol.121(3) :336–341, 1999.
- D.A. Hillson, P.W. Simon, et K.E. Newland. *Project Risk Analysis and Management Guide*. Published by APM Group, High Wycombe, UK, 2004.
- E.C. Honour. Understanding the value of systems engineering. Dans *Fourteenth Annual International Symposium of the International Council On Systems Engineering (INCOSE) International Symposium*, France, 2004.
- A.C. Hughes et P.H. Thomas. Systems, experts and computers. *The Massachusetts Institute of Technology (MIT) Press*, Vol.20(2).
- C.S. IEEE. *IEEE Guide Adoption of A Guide to the Project Management Body of Knowledge*. Software Engineering Standards Committee, 2003.
- INCOSE. *Systems engineering handbook*. International Council on Systems Engineering (INCOSE) Working Group, 2004.
- Project Management Institute. *Management de Projet : Un référentiel de connaissances*. AFNOR, France, 2003.
- D. Leonard-Barton. Core capabilities and core rigidities : A paradox in managing new product development. *Strategic Management Journal*, vol.13, 1995.

- C.H. Loch, A. DeMeyer, et M.T. Pich. *Managing the Unknown : A New Approach to Managing High Uncertainty and Risk in Projects*. John Wiley and Sons, Inc., USA, 2005.
- M. Marcel. *Conduite de projet*. Association Française de Normalisation (ANFOR), France, 2002.
- A.C. Martin et P.H. Thomas. *Systems engineering guidebook : A process for developing systems and products*. CRC Press, 1997.
- H.L. McManus et D.E. Hastings. A framework for understanding uncertainty and its mitigation and exploitation in complex systems. Dans *Fifteenth Annual International Symposium of the INCOSE, Rochester, NY, July 2005*.
- H.L. McManus et T.E. Schuman. Understanding the orbital transfer vehicle trade space. Dans *American Institute of Aeronautics and Astronautics, California, 2003*.
- J.J. Moder et C.R. Phillips. *Project management with CPM and PERT*. Blitz Publishing Company, 1995.
- G. Motet et J.C. Geffroy. *Surete De Fonctionnement Des Systemes Informatiques*. Kluwer Academic Publishers, France, 2002.
- P.A. Muller. *Modélisation object avec UML*. Eyrolles, 2000.
- NASA. *NASA Systems engineering handbook*. NASA Headquarters, Washington, D.C., 2007.
- B. O'Donovan, T.R. Browning, C.M. Eckert, , et P.J. Clarkson. "Design planning and modelling," *Design process improvement : A review of current practice*. Springer, 2005.
- OMG. Uml profile for system engineering rfp. 2003a. OMG Document, <http://www.omg.org/cgi-bin/doc?ad/2003-03-41>.
- OMG. *UML Specification v. 1.0, Final Adopted Specification*. Object Management Group (OMG), 2003b.
- OMG. *UML Specification v. 2.0, Final Adopted Specification*. Object Management Group (OMG), 2005.

- A. Owen. *Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing*. Springer, New York, 1995.
- Michael T. Pich, Christoph H. Loch, et Arnoud De Meyer. On uncertainty, ambiguity, and complexity in project management. *Management Science*, 48(8) :1008–1023, 2002.
- PMI. *A Guide to the Project Management Body of Knowledge*. Project Management Institute (PMI), 2000.
- A.A.B. Pritsker, J.J. O'Reilly, et D.K. LaVal. *Simulation With Visual SLAM and AweSim*. John Wiley and Sons, Inc., New York, 1997.
- D. Qi. *predicting and managing system interactions at Early Phase of the product Development process*. PhD thesis, Massachusetts Institute of Technology (MIT), 2000.
- J. Ralyté. *Ingénierie des méthodes à base de composants*. PhD thesis, University of Paris 1-Sorbonne, 1999.
- S. Rochet. *Formalisation des processus de l'Ingénierie Système : - Proposition d'une méthode d'adaptation des processus génériques à différents contextes d'application*. PhD thesis, Institut National Des Sciences Appliquées De Toulouse, 2008.
- A. Saltelli, S. Tarantola, et F. Campolongo. Sensitivity analysis as an ingredient of modeling. *Statistical Science*, Vol.15(4), 2000.
- S. Schrader, W.P. Riggs, et R.P. Smith. Choice over uncertainty and ambiguity in technical problem solving. *Journal of Engineering and Technology Management*, vol.10 :73–99, 1993.
- J. Sigel. Introduction to omg uml. 2007. http://www.omg.org/gettingstarted/what_is_uml.htm.
- S.A. Sinan. Extending standard uml with model composition semantics. *Distributed Computing*, Vol.44(2) :28–32, 2002.
- P Smets. *Uncertainty Management in Information Systems : Imperfect information : Imprecision - Uncertainty*. Kluwer Academic Publishers, 1999.

- M.E. Smith, J. Burgoyne, et L. Araujo. *Organizational learning and the learning organization, developments in theory and practice*. SAGE publication Inc., London, 2004.
- P.G. Smith et G.M. Merritt. *Proactive Risk Management : Controlling Uncertainty in Product Development*. Productivity Press, New York, 2006.
- R.P. Smith et J.A. Morrow. Product development process modeling. *Design Studies*, Vol.20(3) :237–261, 1999.
- D.K. Sobek II, A.C. Ward, et J.K. Liker. Toyota's principles of set based concurrent engineering. *Sloan Management Review*, vol.40(2), 1999.
- S.C. Sommer, C.H. Loch, et J. Dong. Managing complexity and unforeseeable uncertainty in startup companies : An empirical study. *Organization Science*, vol.20(1), 2009.
- SPEM. *Software Process Engineering Metamodel Specification Version 1.1*. OMG working Group, 2005.
- SPEM. *Software Process Engineering Metamodel Specification Version 2.0*. OMG working Group, 2007.
- R. Stevens, P. Brook, K. Jackson, et S. Arnold. *Systems Engineering : Coping with Complexity*. Prentice Hall Europe, New York, 1998.
- D.V. Steward. *Systems Analysis and Management : Structure, Strategy, and Design*. Petrocelli Books, New York, 2001.
- S. Thomke et D. Reinertsen. Agile product development : Managing development flexibility in uncertain environments. *California management review*, vol.41(1) :8–30, 1998.
- D.L. Thurston. Multiattribute utility analysis in design management. *IEEE Transactions on Engineering Management*, vol.37(4), 1990.
- D.G. Ullman et B.P. Spiegel. Trade studies with uncertain information. Dans *Sixteenth Annual International Symposium of the International Council On Systems Engineering (INCOSE)*, Florida, 2006.
- K.T. Ulrich et S.D. Eppinger. *Product Design and Development*. Irwin McGraw Hill, Boston, 2000.

- B. Unhelkar. *Process quality assurance for UML based projects*. Addison Wesley, 2003.
- A. Villemeur. *Surêté De Fonctionnement des Systèmes Industriels*. Eyrolles, France, 1998.
- M.A. Walton. *Managing uncertainty in space systems conceptual design using portfolio theory*. PhD thesis, Massachusetts Institute of Technology (MIT), 2002.
- T.M. Williams. Practical use of distributions in network analysis. *Journal of the Operational Research Society*, vol.43(3).
- J. Xiao, P. Pinel, L. PI, V. Aranega, et C. Baron. Modeling uncertain and imprecise information in process modeling with uml. Dans *Fourteenth International Conference on Management of Data (COMAD)*, Mumbai, 2008. Computer Society of India.

NOTATIONS

AMDEC	Analyse des Modes de Défaillance, de leurs Effets et de leurs Causes
BCV	Best Case Value
CPM	Critical Path Method
DS	développement de systèmes
DSM	Design Structure Matrices
EIA	Electronic Industries Alliance
INCOSE	International Council on Systems Engineering
IS	Ingénierie des Systèmes
LATTIS	Laboratoire de Technologie et d'Ingénierie des Systèmes
LGP	Laboratoire de Génie de Production
MLV	Most Likely Value
MTBF	Temps moyen de bon fonctionnement
MTTR	Temps moyen de rétablissement
NASA	National Aeronautics and Space Administration
OCL	Object Constraint Language
OMT	Object Modeling Technique
OMG	Object Management Group
OOSE	Object Oriented Software Engineering
PDF	Probability Density Functions
PERT	Program Evaluation and Review Technique
PMBOK	Project Management Body of Knowledge
PMI	Project Management Institute
SPEM	Software Process Engineering Metamodel
TOPCASED	Toolkit in Open-source for Critical Application and Systems Development
UML	Unified Modeling Language
WCV	Worst Case Value
ZOPH	Zielsystem, Objektsystem, Prozeßsystem, et Handlungssystem

λ_D	Taux de défaillances dangereuses
λ_{DD}	Taux de défaillances dangereuses détectées
λ_{DU}	Taux de défaillances dangereuses non détectées
β	Fraction de défaillances détectées qui ont une cause commune
β_D	Fraction de défaillances non détectées qui ont une cause commune
DC	Couverture du diagnostic des fautes dangereuses
t_{CE}	Temps moyen d'indisponibilité d'un module
t_{GE}	Temps moyen d'indisponibilité d'un groupe (de deux modules)
T_i	Intervalle du test périodique
$PFD_{avg.}$	Probabilité moyenne de défaillance