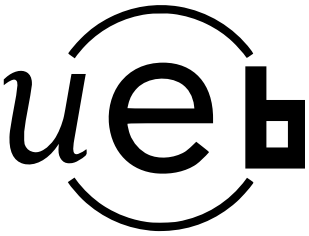
INCREMENT une approche hybride pour modéliser et analyser dans le large les exigences réglementaires de sûreté

Nicolas Sannier

ANNÉE 2013



THÈSE / UNIVERSITÉ DE RENNES 1
sous le sceau de l'Université Européenne de Bretagne

pour le grade de

DOCTEUR DE L'UNIVERSITÉ DE RENNES 1

Mention : Informatique

École doctorale Matisse

présentée par

Nicolas SANNIER

préparée à l'unité de recherche Inria
Inria Rennes Bretagne Atlantique
ISTIC

INCREMENT : une approche hybride pour modéliser et analyser dans le large les exigences réglementaires de sûreté

Thèse soutenue à Rennes
le 12 décembre 2013

devant le jury composé de :

David GROSS-AMBLARD
Professeur à l'Université de Rennes 1 / *Président*

Yves LE TRAON
Professeur à l'Université du Luxembourg / *Rapporteur*

Jean-Michel BRUEL
Professeur à l'Université de Toulouse / *Rapporteur*

Philippe DHAUSSY
Maître de conférence à l'ENSTA-Bretagne / *Examinateur*

Benoit BAUDRY
Chargé de recherche à Inria / *Directeur de thèse*

Thuy NGUYEN
Ingénieur Chercheur à EDF R&D / *Co-directeur de thèse*

A la mémoire de Lisette Sannier (1948-2010)

Remerciements

"I know half of you half as well as I should like,
and I like less than half of you half as well as you deserve."
Bilbo Baggins - The Fellowship of the Ring

Je remercie profondément Benoit Baudry du cadeau qu'il m'a offert ce printemps 2010 et durant ces trois années et demie depuis ce fameux jour de VET et ce papier de Westley Weimer [WNGF09]. Merci pour cette opportunité, cette patience, cet enthousiasme, ces encouragements, cette liberté d'explorer qui m'a été laissée et que j'espère avoir utilisée à bon escient. Merci à Laurence Picci et Thuy Nguyen qui m'ont suivi chez EDF R&D.

Je remercie Jean-Michel Bruel et Yves Le Traon pour m'avoir fait l'honneur de rapporter cette thèse ainsi que David Gross-Amblard et Philippe Dhaussy pour avoir accepté d'évaluer ce travail.

Outre Benoit, je salue tous les membres de l'équipe Triskell, passés et présents, qui m'ont fait découvrir et vivre ce milieu de la recherche. Aux permanents de l'équipe, à mes camarades d'aventure de 2010, à nos sympathiques anciens à qui nous avons sauvagement pris les places, mais avec qui on aura partagé du bon temps, aux plus jeunes qui prennent les nôtres dans cette aventure et au groupe P1A à Chatou. Un clin d'oeil à toutes ces rencontres à travers le monde, des gens qui ont eu un regard curieux, intéressé sur mon travail et avec qui j'ai pu échanger : Martin, Gunter, Daniel et Daniel, Arnaud, Tao, Eero, Sepideh et tant d'autres.

Un clin d'oeil très particulier à tout mes amis, géographiquement proches ou dans le cyber-espace et qui ont partagé mes soirées depuis tant d'années. Des soirées faites d'elfes, de chevaliers, de donjons, de jedis, de cyborgs, de jazz, de cowboys, de space marines, de garde impériaux ou tout simplement de discussions passionnées et qui alimentent mon imaginaire.

Trois ans ! Je ne les ai pas vu passer ... Et dire qu'elles auraient très bien pu ne jamais être ce qu'elles furent tellement l'histoire de cette thèse est incroyable. Trois années : un instant de vie que je dédie à ma mère, partie aux premiers jours de cette aventure, ainsi qu'à mon père pour leur amour et leur courage incroyable à tous les deux.

Résumé long

Le contrôle-commande d'une centrale nucléaire permet de mesurer et de piloter l'activité du coeur du réacteur. Les systèmes de contrôle-commande importants pour la sûreté de fonctionnement doivent répondre à un certain nombre d'exigences, au premier rang desquelles se trouvent les exigences réglementaires, édictées par les autorités nationales. Ces exigences sont complétées par un ensemble de recommandations pratiques, de textes normatifs nationaux ou internationaux. L'ensemble de ces textes expriment différentes exigences de différents niveaux et sont peu corrélés.

La transposition de ces exigences dans un contexte international a montré des écarts dans les exigences et dans les pratiques des différents pays. Cette observation soulève trois problèmes. Premièrement, les exigences de ce domaine sont peu formalisées. Leur connaissance et leur compréhension est détenue par un nombre limité d'experts. Deuxièmement, les relations de traçabilité, et par conséquent l'organisation des exigences de ce vaste domaine est souvent implicite. Troisième problème qui est la conséquence des deux premiers, les passerelles entre contextes nationaux différents sont très peu développées tandis que la compréhension de la variabilité entre les exigences et pratiques dans différents pays devient un enjeu industriel.

Les travaux de cette thèse se situent dans ce contexte industriel en partenariat avec EDF R&D (Electricité de France - Recherche et Développement) et également au sein du projet CONNEXION regroupant les acteurs majeurs du contrôle-commande nucléaire français. Les contributions de la thèse s'articulent autour de l'approche INCREMENT (Intrumentation aNd Control regulatory REquirement Modeling Environment) qui adresse les deux premiers challenges présentés. Celles-ci concernent en particulier :

- **La formalisation du domaine** où nous proposons à la fois une description du domaine et un métamodèle permettant une capitalisation et une vue globale à haut niveau d'un référentiel d'exigences.
- Une approche originale avec une **hybridation entre modélisation et recherche d'information** pour une amélioration de la traçabilité des exigences du domaine.
- **Une base outillée** avec un analyseur configurable pour l'acquisition automatique de documents, l'apport de techniques de recherche d'information pour la traçabilité des exigences et un environnement graphique pour la manipulation et l'analyse des modèles d'exigences.

De ces trois contributions, nous obtenons des résultats particulièrement encourageants. Le métamodèle proposé est utilisé dans l'industrie au sein du projet CONNEXION et nous avons posé les jalons pour son enrichissement dans le futur du projet. Les techniques de recherche d'information pour la traçabilité des exigences souffrent de limitations dues à la forte volumétrie de faux positifs dans la proposition de liens de traçabilité entre documents. Notre approche hybride a permis dans nos expérimentations de réduire, en moyenne, la taille de ces espaces de recherche de 65% comparé aux approches standard de recherche d'information sans pour autant dégrader la qualité de ces espaces de documents candidats.

Long Abstract

Instrumentation and Control systems (I&C) in nuclear power plants allow to measure and control the reactor behavior. I&C Systems important to safety must conform to their requirements, where regulatory requirements are first class entities. Regulatory requirements are written by national safety authorities and are completed using a set of national recommendation guides or national and international standards. All these documents express different levels of requirements and are weakly interrelated.

Putting these requirements in an international context showed important gaps between requirements and practices in different countries. This observation sets three important challenges. First, the global domain knowledge is scattered, not formalized and hold by few experts. Second, traceability links and, said differently, the organization within the domain, is implicit. The third problem is the consequence of the two firsts. Bridges between different national practices are not developed, whereas the understanding of requirements and practices variability concerns becomes a significant industrial issue.

The thesis sets up in an industrial context with EDF R&D (Electricité de France, Research and Development division), and within the CONNEXION project that gathered the French nuclear I&C industry. Its contributions are defined around the INCREMENT approach (Instrumentation aNd Control Regulatory Requirement Modeling Environment) that addresses the two first challenges previously introduced. In particular, they consist in :

- **the domain formalization** itself by the proposal of a metamodel that allows a high level capitalization of a requirements corpus as well as its organization,
- the proposal of an original hybrid approach, **mixing both metamodeling and information retrieval**, and combine them in a mutual beneficial joint use,
- **a tool-support basis** to gather partial knowledge from the textual documents, manipulate such models that conform to the proposed metamodel, and Information retrieval techniques to support better requirements traceability.

From these contributions, results are encouraging. The metamodel and its tool support are used in the industrial context of the CONNEXION project. Where information retrieval techniques for requirements traceability suffer from large sets of false positives limitations, our hybrid approach allowed us to reduce this noise and reduce the size of the candidate links research space by a mean of 65% without decreasing their global quality.

Table des matières

Introduction

Contexte

Durant le cycle de vie d'une centrale nucléaire, de sa conception à son démantèlement, un certain nombre d'exigences doivent être respectées. Les exigences réglementaires de sûreté se situent au premier rang et sont édictées dans des textes nationaux, propres à chaque pays, ainsi que par des normes internationales et sont complétées par des recommandations pratiques dont l'application vaut conformité aux exigences vis-à-vis des autorités de sûreté. Ces exigences peuvent correspondre à des objectifs ou bien à des moyens et les autorités de sûreté jugent l'acceptabilité des pratiques mises en place pour y répondre.

Les travaux présentés dans cette thèse prennent leur origine dans la volonté d'industriels du nucléaire, en particulier EDF (Electricité de France), de mieux appréhender les exigences de sûreté de fonctionnement des systèmes de contrôle-commande, avec une dimension de certification à l'international et plus seulement sur le plan national.

Pour présenter nos travaux, nous partons de deux constats que nous développons dans cette introduction, à savoir : (1) l'évolution depuis le milieu des années 1980 des systèmes de contrôle-commande vers des systèmes numériques et la difficulté à qualifier ces systèmes lorsqu'ils revêtent une importance pour la sûreté ; (2) un "renouveau" du nucléaire avec l'émergence de nouveaux projets au niveau mondial, bien que l'accident de Fukushima en 2011 ait entravé ce renouveau dans certains pays.

L'évolution numérique du contrôle-commande

Les systèmes de contrôle-commande des centrales sont les systèmes qui permettent de mesurer et de piloter l'activité de la centrale afin de la maintenir dans les limites de ses conditions normales d'opération ou de la ramener dans un état sûr en cas de déviation en dehors de ces limites. Conçu dans les années 80 (1982), et mis en service pour la première fois en 1984 sur le Centre Nucléaire de Production d'Electricité (CNPE) de Paluel, le contrôle-commande des réacteurs d'EDF pour le palier 1300MWe (megawatt électriques, niveau de la production électrique de la centrale) a été le premier système au monde mis en œuvre sur des centrales nucléaires à l'aide de technologies numériques. Jusqu'alors, le contrôle-commande reposait sur des technologies analogiques (relais, circuits câblés, etc.).

Cette première mondiale a depuis fait florès mais ce changement de dimension à

eu un impact majeur sur la qualification des systèmes importants pour la sûreté. En effet, si les industriels avaient la maîtrise complète des circuits analogiques, démontrer la sûreté d'un système programmé est d'une toute autre complexité. Il devient difficile de prévoir leur comportement de manière exhaustive ainsi que leur mode de défaillance. L'apparition progressive des composants de moins en moins dédiés à un domaine et de plus en plus génériques (composants sur étagères) ne permettent plus la maîtrise complète de ces systèmes. Encore aujourd'hui, on ne sait pas démontrer l'absence d'erreurs dans un système informatique, qu'il s'agisse du programme informatique lui même, ou du système et des composants sur lequel il est mis en oeuvre. Les exemples malheureux inscrits dans l'histoire de l'informatique tels que les erreurs dans les systèmes d'exploitation grands publics et les ratés industriels beaucoup plus spectaculaires avec, entre autres exemples, le vol inaugural de la fusée Ariane 5 en 1996 ou le crash de la sonde Mars Climate Orbiter en 1999 n'ont pas contribué à améliorer la confiance vis-à-vis du logiciel. La confiance des autorités de sûreté envers les systèmes numériques reste donc relativement limitée.

En 1986, le sous-comité SC45-A de la commission électrotechnique internationale (CEI) a publié une norme internationale dédiée aux aspects logiciels pour les systèmes programmés réalisant des fonctions importantes pour la sûreté : la norme CEI 60880. Etendue en 2000 puis mise à jour en 2006, cette norme est la première d'une longue liste de normes nationales et internationales, génériques ou dédiées au contrôle-commande des centrales nucléaires, qui sont venues peupler le paysage réglementaire en général, celui du contrôle-commande en particulier au sujet de ces composants programmés. On peut noter un certain nombre de préoccupations telles que les critères généraux pour les systèmes de contrôle-commande (CEI 61513 - 2001, puis 2011), des critères plus particuliers vis-à-vis de systèmes classés réalisant des fonctions avec différents niveaux d'importance (CEI 60880, CEI 62138 - 2004), des normes sur la classification des systèmes eux mêmes (CEI 61226 - 1993 puis 2005 puis 2009), sur les communications de données entre systèmes, sur les défaillances de cause commune (62340), le développement des circuits intégrés programmés en HDL (ou plus généralement les FPGA (Field Programmable Gate Arrays)) (CEI 62566, 2011).

En dehors de la CEI, d'autres organismes émettent des normes et recommandations autour du contrôle-commande. L'Agence Internationale de l'Energie Atomique (AIEA), organisation internationale sous l'égide de l'ONU, et dont le but est d'assurer un usage pacifique et sûr des technologies et sciences liées à l'énergie nucléaire a pour rôle, entre autres d'informer et de publier des standards pour la stabilité et la sûreté des installations nucléaires. De même, l'Institute of Electrical and Electronics Engineers (IEEE) publie un certain nombres de normes dont l'application est exigée, recommandée, acceptée ou interprétée par des autorités de sûreté ou encore l'ISO (International Organization for Standardization).

On peut observer différents niveaux d'abstraction dans les différentes normes (critères généraux vs critères spécifiques à certains systèmes classés), ou de précision en termes de domaine concernés. La chronologie présentée brièvement montre aussi l'évolution des préoccupations face aux technologies émergentes mais aussi l'évolution des pratiques avec, par exemple, l'évolution de la classification.

De même, la pratique internationale se divise en deux "mondes" normatifs relativement distincts. Des pays, tels que la France et les autres pays d'Europe de l'ouest suivent le courant CEI/AIEA de la réglementation. D'autre part, un courant ISO/IEEE est suivi aux Etats-Unis ou en Asie. Ces deux référentiels évoluent indépendamment l'un de l'autre même s'il existe depuis plusieurs années des tentatives d'harmonisation dans les pratiques, comme par exemple les travaux d'harmonisation de la WENRA (Western Europe Nuclear Regulators Association) depuis 2006.

Un marché du nucléaire qui se renouvelle au niveau international

EDF (Electricité de France) possède et exploite un parc nucléaire de 58 tranches sur 19 sites qui est articulé autour de 3 paliers, 900MWe (34 réacteurs), 1300MWe (20 réacteurs), 1450MWe (4 réacteurs) auxquels s'ajoutent le futur EPR de Flamanville en construction et celui de Penly en projet. Cependant chaque centrale est vue comme une entité indépendante, du fait de contraintes locales particulières (classement sismique, positionnement en bord de rivière ou en bord de mer, température et débit des rivières, etc). Travaillant en étroite collaboration avec l'Autorité de Sûreté Nucléaire (ASN) depuis sa création, le paysage réglementaire du nucléaire français est un domaine maîtrisé par ses parties prenantes.

La France est le second pays au monde dans le nombre de centrales nucléaires construites sur son territoire, elle est le leader mondial si l'on considère la part de l'électricité d'origine nucléaire. Elle a donc une position de leader affirmée dans le domaine. Avec le pic longtemps programmé des énergies fossiles, le choix de l'énergie nucléaire, peut apparaître comme une alternative à ces énergies (en plus des énergies alternatives et renouvelables). De nombreux pays se sont donc intéressés à cette énergie. Bien que l'accident de Fukushima en 2011 ait modifié l'image du nucléaire, on peut constater que des projets de nouvelles centrales sont à l'étude.

En dehors de la Chine, depuis les années 80, et la construction de 9 réacteurs de 900MW, l'industrie nucléaire française s'est peu exportée hors de France. A partir du milieu des années 2000, des projets émergent à destination des Etats-Unis (consortium franco-américain créé en 2005 pour la promotion de la technologie EPR, 7 réacteurs en projet), de la Finlande (contrat pour un réacteur EPR signé en décembre 2003, construction en cours), Grande-Bretagne (EPR, projet en cours d'évaluation), Inde (contrat pour deux EPR signé en 2009) ou encore très récemment la Turquie (réacteur ATMEA, contrat en cours 2012-2013).

Problématique

Les travaux de cette thèse s'inscrivent dans ce paysage mixte qui a beaucoup évolué ces dernières années avec un "renouveau" du nucléaire et l'émergence de projets de nouvelles centrales, mais aussi avec la profonde évolution de la réglementation autour des systèmes de contrôle-commande. La volumétrie et l'hétérogénéité des documents et des exigences réglementaires, l'indépendance entre les référentiels, et les relations ou le manque de relations entre documents qui définissent des exigences sur la sûreté posent

un certain nombre de problèmes au passage de l'expertise française sur des marchés hors de France.

un problème de formalisation de la réglementation Les projets d'une telle ampleur ont un cycle de vie s'étalant sur plusieurs décennies. Les centrales des anciens paliers ont une durée de vie théorique de 40 ans. L'EPR a été conçu pour aller au delà de 60 années d'exploitation. Le démantèlement des premières centrales implantées dans les années 1950-1960 n'est pas encore achevé. Le cycle de vie de la centrale va donc bien au delà de la carrière pleine d'un ingénieur qui aurait démarré son parcours professionnel sur un tel projet. Se pose alors la question de *la capitalisation d'un savoir*, aujourd'hui détenu par un panel restreint d'experts et qui évolue au gré de l'émergence des technologies, de la réglementation, des pratiques. Se pose également la question de la transposition de ce savoir dans des contextes différents avec l'ouverture à l'international de nouveaux marchés.

un problème de modélisation du domaine Cette connaissance est aujourd'hui morcelée au fil de la documentation, implicite, mouvante au fil des projets, adresse un nombre important de préoccupations différentes. Cette complexité du domaine de la sûreté de fonctionnement de ses systèmes programmés pose donc un second défi. Capturer et formaliser la réglementation, les exigences, les pratiques sont seulement une dimension du problème. Il est aussi indispensable de modéliser ce domaine à travers la description de ses éléments mais également à travers la définition des relations qu'ils possèdent entre eux.

un problème de traçabilité à plusieurs perspectives La traçabilité de ces exigences s'expriment à travers trois dimensions. (1) Cela concerne l'organisation du domaine des exigences du contrôle-commande que nous avons décrit précédemment. (2) Il est également important de comprendre la traçabilité de ces exigences telle que définie par Gotel et Finklestein [GF94], c'est à dire de ces exigences vers leur réalisation dans l'architecture du contrôle-commande prescrit, mais aussi sa justification par rapport à la sûreté. (3) Comprendre les exigences réglementaires qui portent sur le contrôle-commande dans différents pays est une autre dimension de traçabilité mais cette fois-ci entre différents référentiels d'exigences.

un problème de variabilité. Le quatrième défi identifié correspond à la transposition de ce domaine du contrôle-commande français dans un contexte international, c'est à dire, non seulement en le comparant à la formalisation d'autres référentiels, pays avec des marchés nouveaux pour l'implantation de projets nouveaux ou de projets de maintenance. Cette dimension n'a pas été abordée en profondeur dans cette thèse. Elle est abordée à travers les deux défis précédents et est mise en perspective pour la poursuite des travaux après la thèse.

Contributions et résultats associés

Contributions de la thèse

Les travaux de cette thèse se présentent au travers de quatre contributions qui répondent aux trois premiers problèmes soulevés. La tableau 1 illustre la répartition des contributions relativement aux problématiques de la thèse.

TABLE 1 – Organisation des contributions par rapport aux problématiques

	Formalisation du domaine	Modélisation du domaine	Traçabilité des exigences	Variabilité des exigences
Analyse du domaine	Présentation du domaine	-	-	-
(IDM) Métamodèle Connexion	Définition des éléments du MM	Organisation du MM	Définition des relations de traçabilité	définition des relations de comparaison
(RI) Evaluation des approches de recherche d'information	-	-	Evaluation de 4 approches	-
Hybridation IDM/RI	-	-	amélioration de la recherche	-

Elles s'articulent autour d'une utilisation originale de deux approches distinctes que sont l'Ingénierie Dirigée par les Modèles (IDM) [JCV12] et les techniques de recherche d'information. A travers ces quatre contributions, nous apportons une première réponse à un défi plus global que représente la synthèse et l'utilisation d'un modèle d'exigences réglementaires de sûreté. Ces contributions s'inscrivent dans un contexte industriel dans le cadre du projet CONNEXION [1], regroupant les acteurs industriels majeurs du contrôle-commande nucléaire français.

La première contribution consiste en **l'analyse détaillée du domaine particulier des exigences réglementaire de sûreté pour les systèmes de contrôle-commande dans les centrales nucléaires**. Cette analyse se trouve à la conjonction de trois domaines : le contrôle-commande des centrales nucléaires, la sûreté de fonctionnement des systèmes programmés et les exigences réglementaires.

La seconde contribution reprend la première et la seconde problématiques identifiées. A partir de cette description, **nous proposons une formalisation sous la forme d'un métamodèle du domaine.** Ce métamodèle est mis en rapport avec ses propres évolutions au cours de la thèse et offre un regard vis-à-vis d'une problématique tierce que représente l'évolution des métamodèles et ses conséquences. Cette contribution a donné lieu à deux livrables autour de la modélisation du domaine et d'une démarche outillée dans le cadre de la thèse. Ce métamodèle est aujourd'hui exploité dans le cadre du projet CONNEXION et constitue le support de la base outillée que nous avons initiée, à savoir : un analyseur pour l'acquisition et la modélisation de documents, un

1. https://www.cluster-connexion.fr/

environnement graphique pour la manipulation et la navigation dans le modèle.

La troisième contribution de la thèse s'intéresse à la seconde problématique identifiée à propos de la traçabilité. **Nous évaluons la viabilité des techniques de recherche d'information**, déjà employées pour la traçabilité des exigences mais appliquées au domaine des normes du contrôle-commande. Cette étude est basée sur un corpus d'analyse de huit normes internationales concernant les systèmes de contrôle-commande nucléaire. Elle porte sur les avantages et les limitations de quatre approches de recherche d'information à savoir : (1) une approche statistique naïve, (2) l'utilisation d'algorithmes de clustering, (3) l'utilisation d'approche par apprentissage et de modélisation de thèmes, (4) l'utilisation standard d'une pondération TF-IDF de recherche d'information.

La quatrième contribution de la thèse est une hybridation de la seconde et troisième contribution proposant de tirer parti du meilleur des mondes que sont les modèles d'un côté, les index de l'autre. Cette dernière contribution fait l'objet d'une nouvelle brique que de notre base outillée. Il s'agit en effet de les utiliser de manière synchronisée et de maintenir cette synchronisation au sein d'un environnement. Cette hybridation permet à la fois la manipulation des éléments de modèles à différents niveaux de granularité mais également de fournir un mécanisme de filtre via le typage des éléments. Ce mécanisme de filtrage basé sur le modèle offre **une amélioration des techniques de recherche d'information pour la traçabilité des exigences**.

Publications liées à la thèse

La majorité de ce travail a été publié ou soumis pour publication, mais cette thèse offre une vue globale et homogène de l'approche INCREMENT.

- Dans les publications [SB11, SBN11], nous présentons des éléments d'état de l'art académique autour de l'ingénierie des exigences avec un éclairage sur les approches de modélisation ainsi qu'un état de la pratique industrielle autour des exigences réglementaires de sûreté et abordons la formalisation du domaine à travers la métamodélisation.
- Dans la publication [SB12a], nous abordons les questions de traçabilité des exigences et présentons nos expérimentations autour de la recherche d'information pour la détection de thèmes dans un corpus d'exigences normatives.
- Dans la publication [SB12b], nous introduisons l'hybridation entre notre approche IDM et notre approche recherche d'information pour l'analyse dans le large de corpus d'exigences réglementaires.
- Un article en cours de soumission [SB14] fournit une vision globale autour de l'approche INCREMENT comme hybridation IDM/RI et présente les résultats de nos expérimentations autour de la réduction des espaces de liens candidats avec le mécanisme de filtre porté par le typage et les données issues de la modélisation pour la recherche d'information.

Organisation du document

Ce document s'articule autour de cinq chapitres qui abordent de manière progressive la construction de la contribution INCREMENT.

Le chapitre 1 présente le paysage des exigences réglementaires de sûreté pour le contrôle-commande, démontre la singularité des exigences réglementaires vis-à-vis des exigences techniques, fonctionnelles ou non fonctionnelles ("traditionnellement" rencontré dans les travaux autour des exigences) et replace le contexte et les enjeux de la thèse.

Le chapitre 2 présente l'état de l'art selon trois perspectives. Une première perspective Ingénierie des Exigences [Poh10] qui est le cadre global des travaux qui ont été menés durant la thèse. Une seconde perspective liée à l'Ingénierie Dirigée par les Modèles (IDM ou MDE en anglais) [JCV12] et plus particulièrement à l'Ingénierie des Exigences Dirigées par les Modèles (IEDM ou MoDRE) [MMAS11, MAS12c]. La dernière perspective s'articule autour de la problématique de traçabilité des exigences [GF94], et notamment autour de l'application des techniques de recherche d'information (Information Retrieval) [SM86] en traçabilité des exigences.

Le chapitre 3 aborde le premier volet de la contribution INCREMENT. Nous y présentons la (méta)modélisation du domaine des exigences du contrôle-commande nucléaire, l'historique de la construction du métamodèle, son instanciation via un analyseur permettant une capture automatique des éléments issus des documents, ainsi que la manipulation graphique de modèle à travers une interface.

Le chapitre 4 aborde le second volet de la contribution INCREMENT. Dans cette partie, nous abordons la question de la traçabilité dans le domaine et l'utilisation de techniques de recherche d'information comme assistance à l'ingénieur pour la traçabilité des exigences, le regroupement sémantique d'éléments du corpus documentaire au travers de thèmes.

Le chapitre 5 aborde le troisième volet de la contribution INCREMENT. Si les deux parties précédentes présentaient des perspectives indépendantes, modélisation d'un côté, recherche d'information de l'autre, la troisième partie présente l'hybridation de ces deux approches pour une utilisation jointe dans un même environnement.

La conclusion reprend les contributions de la thèse et présente les perspectives de recherche associées.

L'annexe du document présente plusieurs vues différentes sur les différents éléments du métamodèle développé au cours de la thèse.

Première partie

Contexte et état de l'art

Introduction de la partie I

Le premier chapitre de la partie sera consacré à la première contribution de la thèse. Nous décrivons, d'un point de vue extérieur au domaine, le paysage du contrôle-commande nucléaire et ses évolutions vers des systèmes utilisant non plus seulement des composants analogiques mais des systèmes programmés. Nous abordons les challenges nouveaux qui sont apparus du point de vue de la sûreté de fonctionnement et des exigences réglementaires de sûreté pour le contrôle-commande. Nous abordons ensuite un second aspect lié à l'histoire récente du contrôle-commande et qui est lié à ce qui a été appelé le "renouveau du nucléaire", l'ouverture du marché du nucléaire à l'international pour des projets de nouvelles centrales. Ce nouveau contexte impose désormais aux acteurs du nucléaire de se préoccuper à la fois de la certification de leurs systèmes importants pour la sûreté d'un point de vue local mais également d'étendre ce processus aux différents pays où ces acteurs veulent s'implanter et de proposer des systèmes plus génériques et plus facilement adaptables vis-à-vis des pratiques réglementaires locales.

Le second chapitre de la partie se consacre à établir un état de l'art vis-vis des problèmes de formalisation et de traçabilité qui ont été identifiés pour cette thèse. Ce chapitre se consacrera à un panorama du domaine de l'ingénierie des exigences, avec des éclairages particuliers qui concerneront :

- la modélisation d'exigences
- les exigences réglementaires et les travaux autour de la conformité
- la traçabilité des exigences
- la traçabilité des exigences basée sur les techniques de recherche d'information

Chapitre 1

Qualification du contrôle-commande : évolutions et nouveaux défis

Dans ce chapitre, nous présentons la première contribution de la thèse. Nous proposons un certain nombre de définitions autour de la notion de sûreté et présentons plus en détails le paysage de la sûreté de fonctionnement dans le nucléaire, ses acteurs, ses exigences réglementaires (section 1.1). Nous présentons ensuite les évolutions récentes du domaine, et les enjeux qui ont conduits aux travaux de cette thèse (sections 1.2 et 1.3). Finalement, nous posons les prémices des contributions techniques de la thèse, du périmètre de l'état de l'art que nous présenterons par la suite (1.4).

1.1 Guide de survie autour des exigences réglementaires du contrôle-commande

Les systèmes qui revêtent une importance pour la sûreté sont soumis à un processus de certification (ou qualification) vis-à-vis d'exigences émises par les autorités. Bien que ces exigences soient relativement documentées, elles restent de nature textuelle, de haut niveau et donc imprécise, ce qui ne facilite pas les travaux autour de la sûreté elle même mais aussi la communication autour de la sûreté entre les acteurs de la sûreté.

La première section de ce chapitre propose de définir la sûreté de fonctionnement et de la sûreté. La section suivante s'intéresse aux acteurs de la sûreté nucléaire en France et dans le monde. La section suivante s'intéresse plus particulièrement aux exigences elles même et à leur organisation.

1.1.1 Introduction : sûreté de fonctionnement et sûreté (nucléaire)

La sûreté de fonctionnement est l'aptitude d'une entité à satisfaire à une ou plusieurs fonctions requises dans des conditions données. Elle traduit la confiance que l'on peut accorder à un système, *"la propriété qui permet aux utilisateurs du système de*

placer une confiance justifiée dans le service qu'il leur délivre" [VCd88, Lap07]. Dans le glossaire AIEA, elle se définit comme la fiabilité globale d'un système, c'est à dire le degré de confiance que l'on peut raisonnablement accorder à ce système. La fiabilité, la disponibilité et la sûreté sont des attributs de la sûreté de fonctionnement [AIE07]. La sûreté (nucléaire), quant à elle, se définit comme l'accomplissement des conditions appropriées d'exploitation, de la prévention des accidents ou de la réduction des conséquences d'accidents, ayant pour résultat la protection des travailleurs, du public et de l'environnement vis-à-vis des risques excessifs de rayonnement.

On retrouve des définitions similaires pour la sûreté et la sûreté de fonctionnement dans les autres domaines utilisant des systèmes critiques tels que l'aéronautique ou le ferroviaire. Nancy Leveson du MIT [Lev12], propose une définition en deux temps qui reprend les définitions de sûreté comme étant *"freedom from accident"* et où les "accidents" sont définis comme des évènements avec perte (perte de vies humaines ou blessures, impacts sur l'environnement). Cette définition rejoint la définition précédente, mais sans la particularité due aux radiations propres à l'industrie nucléaire pour laquelle le terme sûreté sous-entend sûreté nucléaire. Les définitions divergent pourtant sur la notion de fiabilité. Leveson indique que l'on amalgame, à tort, fiabilité et sûreté de fonctionnement, arguant que l'on peut avoir des systèmes sûres mais non fiables (avec des erreurs qui ne portent pas à conséquences) et inversement des systèmes fiables mais non sûres. La sûreté dans le monde nucléaire repose sur ce principe de fiabilité.

1.1.2 Qui sont les acteurs de la sûreté nucléaire ?

1.1.2.1 Les acteurs de la sûreté

Dans cette section, nous décrivons tout d'abord le modèle français pour la sûreté nucléaire et proposons une comparaison de ce modèle sur deux autres pays : la Grande Bretagne et les Etats-Unis d'Amérique.

Au sommet de la pyramide de la sûreté, l'autorité de sûreté nucléaire (ASN) [1] assure, au nom de l'état, le contrôle de la sûreté nucléaire et de la radioprotection en France, pour protéger les travailleurs, les patients, le public et l'environnement des risques liés à l'utilisation du nucléaire.

En support de l'ASN, l'institut de radioprotection et de sûreté nucléaire (IRSN) représente la compétence technique de l'ASN et émet des avis pour la gestion du risque nucléaire et radiologique, y compris en cas de crise.Ce fut le cas lors de l'accident de Fukushima où des experts de l'IRSN ont assisté les parties prenantes sur place. Outre sa mission de support technique à l'ASN, l'IRSN est active en recherche de développement, participe à la normalisation, à la formation, à l'information du public sur les risques nucléaires et radiologiques.

Viennent enfin les différents exploitants, utilisateurs du nucléaire ou rayonnements ionisants. Ils sont responsables de la sûreté de leurs installations et doivent en justifier devant l'autorité de sûreté. Au premier rang desquels les activités industrielles comme la production d'électricité pour EDF ou Areva. On y compte également l'entreprise

1. http://www.asn.fr

d'armement naval DCNS (anciennement Direction des Chantiers Navals) qui conçoit et construits des sous-marins et porte-avions à propulsion nucléaire dans le domaine de la défense ou encore d'autres industriels ayant recours aux rayonnements ionisants pour la stérilisation de matériel, le contrôle de paramètres (empoussièrement de l'air, radiographie industrielle, détection d'usure), contrôle aux rayons X. On y compte également le CEA (Commissariat à l'énergie atomique et aux énergies alternatives) qui possède et exploite quelques réacteurs dédiés à la recherche ou à la défense. Sont également concernées les utilisations médicales faisant appel, tant pour le diagnostic que pour la thérapie, à diverses sources de rayonnements ionisants qui sont produits soit par des générateurs électriques, soit par des radionucléides.

1.1.2.2 Les interactions entre ces acteurs dans un scénario

Le modèle français de la sûreté est défini autour d'une collaboration proche entre les autorités d'un côté et les exploitants de l'autre. On peut envisager cet entrelacement tel que montré dans la figure 1.1.

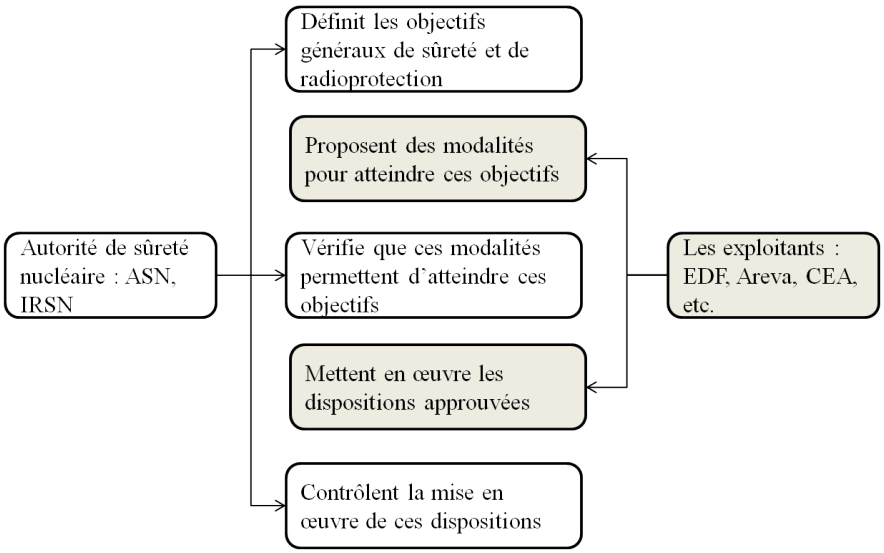


FIGURE 1.1 – les acteurs de la sûreté nucléaire en France

En haut de cet entrelacement, il est important de noter que les AS définissent des objectifs généraux pour la sûreté. Il y a donc un écart important entre, d'un côté, (0) ces objectifs de haut niveau, les exigences réglementaires et la façon dont ils sont mis en oeuvre, ce qui nécessite : (1) la proposition de telles dispositions par les exploitants, (2) la validation de ces dispositions par les AS et, (3) la mise en oeuvre de ces dispositions.

La section suivante s'intéresse à décrire cet écart entre les exigences réglementaires et leur mise en oeuvre.

1.1.3 Les exigences de sûreté

Durant le cycle de vie d'une centrale nucléaire, de sa conception à son démantèlement, un certain nombre d'exigences doivent être respectées. Au premier rang desquelles se situent les exigences réglementaires de sûreté qui sont édictées par des textes nationaux, propres à chaque pays, ainsi que par des normes internationales. Ces exigences peuvent correspondre à des objectifs ou bien à des moyens et les autorités de sûreté jugent l'acceptabilité des pratiques mises en place pour y répondre.

Dans le cas de la France, les Règles Fondamentales de Sûreté (RFS) sont des documents de très haut niveau, qui en général spécifient des objectifs à atteindre, et non des exigences techniques. Cependant, certaines RFS (en particulier celles sur le logiciel) mentionnent des normes CEI comme étant des pratiques acceptables pour atteindre un objectif donné. Dans la plupart des pays, les exigences réglementaires sont ainsi complétées par des recommandations pratiques : leur application vaut conformité aux exigences. Ces recommandations ne sont pas à proprement parler des exigences, et un concepteur a toujours la possibilité de ne pas les appliquer. Cependant, un tel choix conduira à une instruction beaucoup plus longue et beaucoup plus hasardeuse quant au résultat. Ces recommandations réglementaires sont donc perçues comme des exigences. Dans la suite de ce document, le terme générique exigence réglementaire couvre également les recommandations.

Face à la réglementation, EDF et AREVA ont développé des RCC (Règles de Construction et de Conception). Ces documents sont plus détaillés et sont revus et approuvés par l'ASN comme étant des pratiques acceptables pour satisfaire aux RFS. Pour les fonctions de contrôle-commande, les exigences réglementaires sont reprises dans le cahier des charges de centrale via les RCC-E (RCC des matériels Electriques). Les RCC conservent une part d'ambiguïté et ne sont pas appliquées dans les autres pays.

En plus des exigences réglementaires édictées par les autorités de sureté nationales, on trouve un certain nombre d'exigences issues de normes nationales et internationales et dont l'application est requise ou recommandée plus ou moins explicitement par les autorités. Le monde du nucléaire est divisé en deux mondes normatifs : les pays qui suivent le couple AIEA-CEI (dont font partie les pays européens), et ceux qui suivent le couple ISO-IEEE (parmi lesquels les Etats-Unis, le Japon, la Corée ou Taiwan).

L'AIEA énonce des règles relatives à la sûreté des installations nucléaires dans différents documents répartis en catégories selon leur force (fondamentaux de sûreté, exigences de sûreté, guides de sûreté, documents techniques). L'ensemble des exigences reste de haut niveau. En complément, les normes CEI suivent les principes édictés par ces documents et énoncent des exigences et recommandations plus détaillées. Les normes CEI sont d'application volontaire, et tandis que les documents AIEA sont pris en bloc, un pays ou un concepteur de centrale peut choisir les normes CEI qu'il veut appliquer.

La NRC, autorité de sûreté nucléaire aux Etats-Unis, a commencé à développer son approche avant la publication des documents de l'AIEA. Des règles détaillées ont ainsi été édictées et complétées par les documents de mise en œuvre définis par les industriels américains au sein de l'IEEE. En plus des textes internationaux, on trouve les réglementations nationales, propres à chaque pays et qui entraînent des approches

différentes. En tout état de cause, les textes réglementaires restent majoritairement peu prescriptifs.

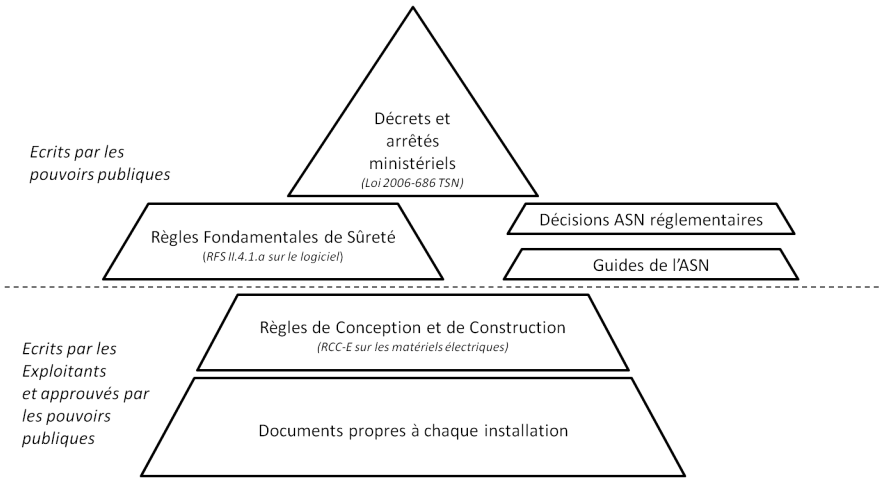


FIGURE 1.2 – La réglementation française

La pyramide de la figure 1.2 présente cet étagement et l'organisation de la réglementation au niveau français. Elle ne tient pas compte de la documentation internationale qui vient en support tout au long du processus de qualification des systèmes.

En élargissant cette description propre à la France, on peut s'apercevoir que la réglementation dans les autres pays du monde procède de manière relativement similaire. Ainsi les grands pays sont dotés d'instances réglementaires comme la Nuclear Regulatory Commision (NRC) aux Etats-Unis, la toute récente (création en février 2011) ONR (office for nuclear regulation) en Grande Bretagne, agence issue du HSE (Health and Safety Executives). D'un point de vue plus global, l'AIEA (sous l'égide de l'ONU) assure les mêmes missions, ce qui tend à illustrer la similitude dans les actions mais une dispersion et une hétérogénéité dans les pratiques de chaque pays.

Tous s'assurent de la sûreté des systèmes au nom de leur état et publient guides et recommandations et s'appuient sur un corpus normatifs, documents techniques issus d'organismes internationaux ou nationaux.

D'un point de vue plus général, on peut aborder le paysage des exigences réglementaires comme l'illustre la figure 1.3 [SB12a]. Ainsi, lorsqu'un industriel propose un système (quelque soit sa taille, sa granularité) dont la sûreté de fonctionnement peut être questionné, celui-ci doit se conformer non seulement à ses propres exigences, mais aussi aux exigences de sureté. Ces exigences sont issues de documents divers et variés allant des textes et guides réglementaires à l'application de normes particulières.

Cette organisation, qui repose sur un couplage entre autorité de sûreté, exploitant, et organismes internationaux, montre la diversité des différents types de documents qui définissent les exigences de sûreté. Elle met aussi en avant le faible couplage entre ces différents documents et un défi en terme de traçabilité pour les exigences de ce domaine.

TABLE 1.1 – Réglementation à l'échelle internationale

Niveau	**France**	**Etats-Unis**	**Grande-Bretagne**	**AIEA**
Législatif	Loi 2006-686 relative à la transparence et à la sécurité en matière nucléaire (TSN) ; Ordonnance 2012-6 du 5 janvier 2012 modifiant les livres Ier et V du code de l'environnement ; etc.	Atomic Energy Act (1954) + amendments	Nuclear Installations Act (1965), Health and Safety at Work Act (1974)	N/A
Réglementaire	ASN, IRSN ; Règles Fondamentales de Sûreté (RFS)	NRC (Nuclear Regulatory Commission) ; Code for Federal Regulation 10CFR50, 10CFR51, etc.	ONR (office for Nuclear Regulations) Safety Assesment Principles	SF (Safety Fundamentals), SR (Safety Regulations)
Guides et Recommandations réglementaires	Guides et règles de L'ASN	NRC regulatory Guidance	ONR, "Technical Assesment Guides" ; "Licensing Nuclear Installations"	SG (Safety Guides)
Normes, documentations techniques	Normes CEI, ISO, etc.	Standards IEEE, documents EPRI, Branch Technical Positions (BTPs), etc.	Normes CEI, ISO, etc.	AIEA NS-R1, NS-G1.3, etc.

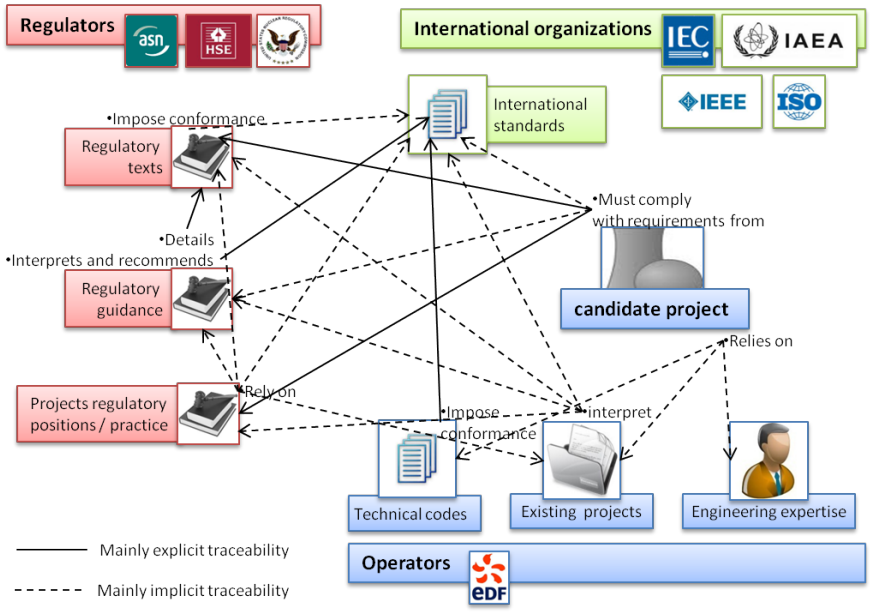


FIGURE 1.3 – Organisation de la réglementation d'un point de vue général

1.2 Evolution des systèmes de contrôle-commande et des exigences

1.2.1 Le logiciel dans les systèmes critiques : "je t'aime moi non plus"

Depuis la première mondiale que fut l'installation d'un contrôle-commande numérique dans la centrale de Paluel, le monde du contrôle-commande connaît une révolution numérique avec la disparition sur le marché et le remplacement progressif (mais aussi subi car problématiques pour la sûreté) des composants analogiques, spécifiques aux missions et aux industries les utilisant vers l'utilisation de composants pris sur étagère (Commercial Off The Shelf, COTS). Ces derniers sont génériques, et vendus en grande quantité. Cette révolution s'applique à tous les niveaux du contrôle-commande avec l'apparition des réseaux de capteurs sans fil, l'utilisation de bus de données en fibre optique, jusqu'aux interfaces homme-système (IHS) numériques.

La part croissante des COTS depuis les années 1990, et la bascule des comportements analogiques ou logiques "câblés" et dédiés au domaine, vers des comportements logiques "programmés" n'est pas sans soulever des problèmes importants. Parmi eux, on peut considérer la volatilité et la non maîtrise de l'obsolescence des composants par l'industriel (disparition / rachat du fournisseur par des tiers, perte du marché, arrêt du support du composant, etc.), les problèmes d'intégration dans les systèmes, de performance fonctionnelle (l'adéquation du composant à remplir sa fonction dans son environnement). La perte de maîtrise sur les COTS et leur obsolescence a récemment été illustrée avec le projet de supercalculateur Condor de l'US Air Force, utilisant un

cluster de consoles de jeux Sony PlayStation3 mises en réseau. L'arrêt, en avril 2010, du support à l'utilisation de Linux par Sony, a eu pour effet collatéral l'arrêt du cluster utilisé par l'US Air Force.

S'y ajoute une mauvaise image du logiciel tant chez le grand public que parmi les industriels, due notamment à la longue litanie d'évènements/accidents (et leurs coûts relatifs) déclenchés par des erreurs dans le logiciel tout au long de ces dernières décennies (vol 501 inaugural du lanceur Ariane 5 le 4 juin 1996 lié à l'utilisation hors dimensionnement d'un composant logiciel utilisé sur le lanceur précédent et réputé fiable) ; "écrans bleus" dans les machines grands public (1995/1996 Ping of death), la perte, le 23 septembre 1999, de la sonde Mars Climate Orbiter suite à une erreur d'unité de mesure dans le logiciel de navigation, etc.), autant d'erreurs incroyablement simples mais aux conséquences significatives qui sont autant de rappels qui ravivent la méfiance envers le logiciel pour les systèmes critiques.

En conséquence, la réglementation vis-à-vis du logiciel a fortement évolué ces dernières années. Cette évolution fait écho à l'évolution du marché et des préoccupations de sûreté. Cette évolution s'est traduite à travers tous les domaines, au niveau générique d'une part, avec la norme CEI 61508 publiée dans sa première version entre 1998 et 2000 et mise à jour en 2010) ou spécifiquement aux domaine. C'est le cas dans la santé avec l'évolution à la fois de la réglementation sur les matériels utilisant du logiciel mais aussi la communication des données informatisées des patients. C'est également le cas dans l'avionique avec la norme mondialement acceptée et mise en avant par les autorités américaines (Federal Aviation Administration - FAA) et européennes (European Aviation Safety Agency - EASA) DO-178B (1992) et sa mise à jour DO-178C (2012). Une telle réglementation est également apparue plus récemment pour l'automobile (ISO 26262 - 2011). En ce qui concerne le nucléaire, cela s'est traduit par un durcissement de la réglementation et une forte augmentation du nombre des normes et donc des exigences.

1.2.2 Evolution de la réglementation sur le logiciel dans le contrôle-commande

La réglementation et la normalisation dans le secteur nucléaire se différencient des autres dans le sens ou il n'existe pas une norme ou un ensemble de normes qui fasse consensus au niveau mondiale, mais deux grands ensembles normatifs.

La figure 1.4 propose une vue de la chronologie des grands textes réglementaires et des normes CEI sur le contrôle-commande et en particulier sur les aspects logiciels du contrôle-commande. Cette chronologie qui s'étend de 1950, avec le "Nuclear Energy Act" (refondation du Nuclear Energy Act de 1946) et sur lequel continue de se fonder la réglementation américaine, jusqu'en 2012 avec la parution tardive de la norme CEI 62566 (fin 2011) sur les aspects logiciels pour les FPGAs et la publication de la norme de sûreté AIEA (IAEA Safety Standard SS-R2/1).

La liste des normes mentionnées comprend des préoccupations à différents niveaux de granularité. Ainsi, la CEI 61508 et la CEI 61513 expriment des exigences de haut niveau (il s'agit de normes dites de premier niveau) et des critères généraux pour le logiciel important pour la sûreté, la 61508 étant une norme générique, tandis que la

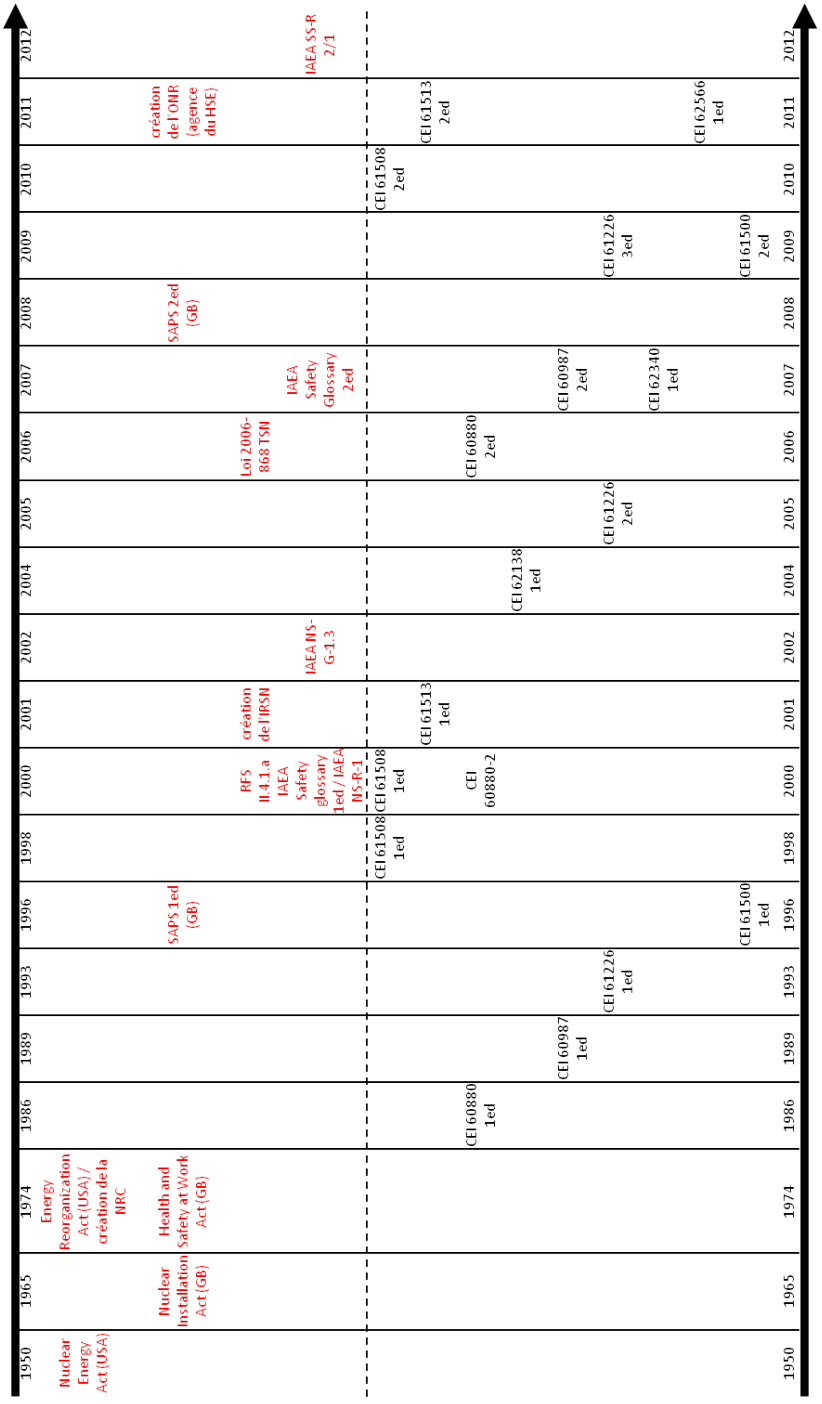


FIGURE 1.4 – Evolution des textes réglementaires nationaux et internationaux touchant au logiciel dans le contrôle-commande nucléaire

61513 est une spécialisation de cette dernière pour la filière nucléaire. On y retrouve des normes plus précises, de second niveau, autour des aspects logiciels dans les systèmes réalisant des fonctions de catégorie A (60880), B ou C (62138), la classification de sûreté (61226), la défaillance de cause commune (62340), les aspects matériels pour le logiciel (60987), ou encore le développement de logiciel à partir de FPGAs (62566). Viennent ensuite les normes de troisièmes niveaux telles que la CEI 61500 sur la communication de données pour les systèmes réalisant des fonctions de catégories A. Il existe au niveau de la CEI un quatrième niveau de normes, standards et recommandations que nous ne traiterons pas ici.

Cette chronologie montre à la fois l'étendue de la réglementation dans le temps et qui est toujours en vigueur, mais aussi l'emballement, depuis 1986 et l'introduction du premier système numérique de contrôle-commande, de la normalisation autour des aspects logiciels, tant dans des aspects généraux et de leur mise à jour progressive que dans les nouvelles préoccupations qui émergent et qui font suite à des interrogations des industriels et des autorités sur ces dites préoccupations (comme la qualification de systèmes utilisant des FPGAs non dédiés au monde nucléaire.

Autre phénomène important, les différents textes et exigences sont faiblement corrélés et la traçabilité entre eux, le plus souvent implicite et reposant sur la pratique réglementaire de chaque pays. Ainsi, la pratique française met en avant l'application de la norme CEI 60880-2006 (édition 2006) concernant les aspects logiciels pour les systèmes réalisant des fonctions de catégories A, et la norme CEI 62138 (2004) concernant les aspects logiciels des systèmes réalisant des fonctions de catégories B et C tandis que le texte réglementaire, la règle fondamentale de sûreté sur le logiciel, n'a pas été mise à jour depuis sa première publication en mai 2000. Par conséquent, elle ne peut pas avoir connaissance et mentionner des travaux et publications postérieurs relatives aux nouvelles technologies ou même sur la mise à jour de la classification (1E dans la RFS, catégorie A, B, et C pour la CEI, F1a, F1b, F2 pour le European Utility Requirements).

L'évolution rapide des exigences des autorités de sûreté n'est pas suivie au niveau des différents textes réglementaires et propose donc une forte connaissance tacite et assez peu formalisée. En conséquence, il y a une rupture du point de vue de la traçabilité dans la pyramide réglementaire. Cette connaissance tacite, si elle est maîtrisée dans le contexte français par les industriels français pose néanmoins un véritable défi lorsqu'il s'agit de la transposer dans d'autres pays où la réglementation et les pratiques diffèrent. Nous abordons cet aspect dans la prochaine section.

1.3 Le contrôle-commande face à l'internationalisation du marché

Dans cette section, nous abordons la motivation des travaux de la thèse sous la perspective du "renouveau du nucléaire", expression consacrée au milieu des années 2000, avec le regain d'intérêt des pays pour l'énergie nucléaire et qui mène aujourd'hui des industriels comme EDF à s'intéresser aux exigences réglementaires dans différents pays, désormais dès la phase de conception de leurs systèmes.

Nous présentons tout d'abord un historique de l'ouverture de ces marchés et l'implication des acteurs français EDF et Areva à travers les projets de différents EPRs à travers le monde.

1.3.1 Depuis 2000, renouveau et rechute du nucléaire

1.3.1.1 Renouveau du nucléaire

Depuis le début des années 2000, l'intérêt pour l'énergie nucléaire est de nouveau remonté avec notamment la volonté du gouvernement américain de relancer la filière avec le "Nuclear Power Plant 2010 program" (2002) et l'Energy Policy Act" (2005) qui vise à trouver de nouveaux sites d'installation et la conception de centrales de nouvelles génération. Depuis 1979 et l'accident de Three Mile Island en 1979, les Etats-Unis n'ont plus connu de projet de nouvelle centrale, la plupart ayant été mise en service au plus tard en 1974. Depuis l'accident de Tchernobyl en 1986, à l'exception de quelques pays comme la France, la Corée du Sud et le Japon, cette tendance s'était généralisée au niveau mondial. Au delà de l'objectif de construire au moins une centrale de nouvelle génération aux Etats-Unis, il y a aussi la volonté d'éprouver de nouvelles approches de conception et de qualification pour la sûreté et de se mettre à jour après plusieurs décennies sans démarche complète sur des projets neufs. C'est ainsi qu'apparaît le programme AP1000 de Westing House, concurrent du programme EPR d'Areva et EDF.

Cette démarche a été imitée en Grande-Bretagne avec un livre blanc sur l'énergie (Energy White Paper - 2003) dès 2003, remis à jour en 2007, mettant l'accent sur la nécessité d'avoir des sources d'énergies fiables et à faible bilan carbone, parmi lesquelles la filière nucléaire tient une place importante (le livre blanc rappelant que *"Nuclear power is currently an important source of carbon-free electricity ... This white paper does not contain specific proposals for building new nuclear power stations. However we do not rule out the possibility that at some point in the future new nuclear build might be necessary if we are to meet our carbon targets."*). Contrairement, aux Etats-Unis, le programme nucléaire britannique n'a pas connu d'arrêt majeur même si le plus récent des 16 réacteurs, Sizewell-B, a été démarré en 1995 (contre 1999 pour le dernier réacteur français). Pas moins de 6 consortiums industriels, dont EDF, se sont déclarés intéressés et ont démarré les consultations dès 2007.

Sur ces deux marchés et dans la généralisation de ce mouvement que les analystes ont baptisé "renouveau du nucléaire" [Nut04], EDF et Areva se sont positionnés avec l'EPR et ont entamé les démarches de certification dès 2007 pour la Grande Bretagne et en 2008 pour les Etats-Unis. En 2009, on a compté jusqu'à vingt-six projets différents pour les Etats-Unis. Dans le reste du monde, on peut observer des démarches similaires durant la décennie avec des projets de nouveaux réacteurs en Italie (projet pour quatre nouveaux réacteurs type EPR), un EPR en construction en Finlande depuis 2005, un accord de coopération franco-émirati et américano-émirati en 2008 et 2009. On retrouve aussi un intérêt croissant de la part des économies émergentes avec quatre réacteurs en cours de construction au Pakistan et deux autres en projet depuis 2009, deux EPRs en

cours de construction en Chine, un contrat signé en 2010 pour un réacteur en Argentine ou encore des accords passés avec l'Inde pour deux EPR, etc.

1.3.1.2 Crise économique et accident de Fukushima, rechute du nucléaire

Si on pouvait croire à un renouveau du nucléaire au début des années 2000, portés par les préoccupations de l'après-pétrole, de dépendance énergétique vis-à-vis d'états tiers pour les uns et les objectifs de réductions d'émissions de dioxyde de carbone pour d'autres, ce renouveau a été complètement remis en question avec la crise financière mondiale depuis 2009, période durant laquelle nombre de projets ont été retardé ou annulés. Cela a été le cas, par exemple, aux Etats-Unis où nombre des consortiums candidats se sont retirés à partir de 2010. Il ne reste aujourd'hui que deux consortiums en course en Grande-Bretagne. Au Etats-Unis, la NRC qui prévoyait des besoins pour une trentaine de nouveaux réacteurs n'a plus en ligne de mire que quatre à cinq nouveaux réacteurs. Ce fut aussi la fin du programme EPR sud-africain et la prévision de construction de trois EPRs et qui s'est arrêté en 2009, ou enfin les reports successifs du deuxième EPR français à Penly en 2011 et 2012.

Le second évènement marquant la rechute du nucléaire est lié à l'accident à la centrale de Fukushima Daishi en mars 2011. Conséquence directe de l'accident, l'Italie, par référendum, décide sa sortie du nucléaire. On constate une démarche similaire en Suisse où les deux projets en cours ont été suspendus, ou ou Allemagne avec la décision de l'arrêt définitif des réacteurs d'ici à 2022. Si l'accident à la centrale de Fukushima Daishi a significativement ralenti le renouveau de la filière, et conduits les différentes autorités de sûreté à considérablement durcir leurs exigences de sûreté et soumettre les réacteurs sous leur responsabilité à des stress test [ASN11, ONR11, NRC11], ce mouvement n'est pas arrêté pour autant puisque la Turquie vient de signer un contrat avec Areva/Mitsubishi pour un réacteur de type ATMEA plus petit que l'EPR.

1.3.2 EPR dans 5 pays ; du produit sur mesure vers une ingénierie de ligne de produits

Dans cette période de renouveau du nucléaire, on a vu qu'EDF et Areva se sont positionnés avec succès sur le marché des centrales de nouvelle génération avec l'EPR. En plus des constructions en cours en Finlande, Chine et France, des discussions pour la certification sont en cours aux Etats-Unis et en Grande Bretagne. Comme nous le disions dans la section 1.1, les textes réglementaires et les approches de sûreté sont différentes dans les pays. Cela se traduit non seulement par des exigences différentes mais également par des interprétations différentes des normes internationales dont l'application est, à des degrés divers, exigée, recommandée ou acceptée selon les pays.

Cette hétérogénéité est reconnue dans la communauté nucléaire internationale. Elle a donné lieu à une analyse et une tentative de comparaison entre les les normes de la CEI et de l'IEEE par Gary Johnson en 2001 [Joh01]. Des efforts d'harmonisation sont en cours depuis plusieurs années. Il s'agit d'un des objectifs du Nuclear Power Plant 2010 Program américain et fait l'objet d'une démarche commune au niveau des pays

européen au sein de la WENRA (Western Europe Nuclear Regulators Association) avec des travaux menés en ce sens depuis le milieu des années 2000 et un rapport publié en 2006 [RHW06] ainsi qu'un état d'avancement en 2011 [RHW11].

Un des exemples les plus explicites de cette hétérogénéité tient dans la classification de sûreté des fonctions et des matériels et qui est représentée dans la figure 1.5.

National or international	Classification to the importance to safety (excerpt from IAEA-TECDOC-780)			
IAEA	Systems important to safety		System not important to safety	
	Safety systems	Safety related systems		
IEC 1226	Category A	Category B	Category C	Unclassified
France N4	1E	2E	IFC/NC	
European utility requirements	F1A (auto)	F1B (auto & manual)	F2	Not classified
UK	Category 1	Category 2	Not classified	
USA(IEEE)	Class 1E	Non-class 1E		

FIGURE 1.5 – Hétérogénité de la classification de sûreté

Non seulement le système de classification varie entre les pays (classification France, UK, USA), entre les organismes (AIEA, CEI) mais également dans un même pays puisque la classification a évolué en France entre les paliers 900 (classification de par la conception originelle WestingHouse), N4 (qui est une extension de la certification Westinghouse faite par EDF) et l'EPR qui a adopté la classification de l'european Utility Requirements et non la mise à jour de la classification dans la norme CEI 61226 (dont la dénommination précédente était la CEI 1226). De la même façon les périmètres de la classification sont différents.

Cette hétérogénéité dans les exigences de sûreté est la cause directe des difficultés rencontrées par EDF et Areva dans la certification de l'EPR dans les différents pays où ce dernier a été proposé. Ainsi, depuis 2008, sur les cinq projets d'EPR les plus avancés (construction en Finlande, France et Chine, certification en cours aux Etats-Unis et en Grande-Bretagne), EDF et Areva en sont désormais à quatre architectures différentes pour le contrôle-commande et à cinq processus de certification propres à chaque pays. EDF et Areva, capitalisant sur leur expertise en France ont mis en avant la certification française initiale acceptée par l'ASN. Alors que le processus de certification s'étale sur plusieurs années, celui-ci s'est vu rallongé et les efforts pour la certification augmentés pour chaque pays. Ces efforts démarrés depuis les années 2007 et 2008 pour les EPRs britanniques et américains ne sont toujours pas arrivés à terme depuis et ce, alors que les discussions en sont au stade de l'architecture de haut niveau, avec des exigences de haut niveau.

Faisant écho aux objectifs du Nuclear Power Plant 2010 Program et aux travaux

d'harmonisation menés par la WENRA, l'histoire de la certification de l'EPR dans différents pays montre à quel point il est nécessaire pour les industriels de trouver des similarités dans les exigences de sureté et de proposer un socle d'architecture de haut niveau qui puisse convenir à la plupart des exigences réglementaires exprimées par les autorités de sûreté. Une telle base permettrait de réduire les coûts et le temps de la certification des projets. L'objectif est donc de se diriger vers une ingénierie de ligne de produits avec des similarités et des variants tant du point de vue des exigences que de l'architecture et de la qualification que ces exigences impactent afin de faire face à ces multiples nouveaux interlocuteurs.

1.4 Une approche hybride pour un problème à plusieurs dimensions

1.4.1 Capitaliser les exigences écrites et la connaissance tacite, un problème de formalisation et de modélisation

Nous pouvons faire les observations suivantes. **L'industrie du nucléaire possède un large corpus de documents nationaux et internationaux écrits à plusieurs niveaux et qui évoluent**. Outre les texte nationaux, les différents référentiels d'exigences se basent sur un nombre relativement important de normes et recommandations internationales. L'ensemble de ces documents présente des informations avec des niveaux d'importance, de précision et/ou d'interprétation hétérogènes. Si les exigences sont maîtrisées au niveau local, elles le sont beaucoup moins une fois transposées au niveau international. De la même manière, il est difficile de pouvoir les comparer entre elles.

Les relations explicites ou implicites qui existent au sein des différents référentiels représentent un enjeu important pour la compréhension de l'organisation de ces exigences. Ces relations sont plus que des simples liens reliant deux artéfacts et font partie intégrante de la formalisation **des pratiques et de la connaissance tacite à capitaliser** [SS05, SGN11].

Un métamodèle pour abstraire, formaliser et capitaliser. Plusieurs questions se posent alors. Modéliser à quelle granularité ? Au niveau de la norme comme les travaux de Gary Johnson [Joh01] ? Au niveau de l'exigence ? Faut-il également représenter l'environnement de l'exigence ? Comment modéliser le texte de ces documents ? Quelles informations représenter ? Quelles sont les approches de modélisation d'exigences proposées ?

Aujourd'hui, le langage naturel est le média le plus commun, le plus simple et le plus utilisé pour communiquer ces exigences dans ces industries multipliant les domaines d'expertise et pour des projets à longue durée de vie. Prise unitairement, la nature de ces exigences de haut niveau les rendent difficilement interprétables et vérifiables de manière programmatique sur une architecture ou dans un processus de certification. Quelles sont les propriétés et les interactions de ces exigences ? L'objectif en cours n'est pas d'adresser

la vérification ou la certification vis-à-vis d'éléments particuliers parfaitement identifiés dans un document. L'objectif est bien d'obtenir **un panorama global, dans le large, de l'ensemble du référentiel d'exigences** et de l'utiliser au niveau d'abstraction dans lequel il est exprimé.

Une manière d'apporter un cadre homogène à ces exigences et de les formaliser est d'en faire un modèle, perspective de plus haut niveau de ce domaine. L'apport d'un langage de modélisation permet de manipuler ses concepts avec une sémantique précise et ainsi d'homogénéiser le domaine dans un canevas unificateur. Deuxième propriété, on peut modéliser et donc expliciter les liens pour l'instant implicites. Ce qui permet à la fois de formaliser cette connaissance du domaine mais aussi de la capitaliser. Troisième apport, c'est la possibilité de raisonner non pas document par document mais de manière globale au niveau du modèle.

1.4.2 Exigences, architecture, qualification : un problème de traçabilité

La traçabilité comme moyen de comprendre les relations dans un domaine. La traçabilité des exigences [GF94] est couramment exigée dans les milieux industriels pour les systèmes critiques. Dans le cas présent, il est important de considérer la traçabilité selon deux facettes. La traçabilité classique, des exigences vers l'architecture et la qualification. Il s'agit, dans ce cas, de mettre en oeuvre **la traçabilité "comme mécanisme pour maintenir la cohérence en présence du changement"** telle que défini par Lamsweerde [vL09]. La traçabilité est donc particulièrement utile pour comprendre les exigences impactées par un changement de référentiel pour un système, l'analyse d'impact étant une activité courante. Le second niveau de traçabilité à prendre en compte est bien évidemment la traçabilité dans le référentiel pour **expliciter l'organisation du domaine et les relations entre ses éléments**.

Recherche d'information pour la traçabilité des exigences. S'il est possible de capturer des métriques sur les modèles d'exigences [MBC+13] ou, en ingénierie des modèles, de calculer et d'opérer sur le modèle [JCV12], la forte nature textuelle des informations et des propriétés à analyser limite de telles approches. En particulier, se pose la question de la construction des relations entre éléments du modèle. Si les relations explicites relèvent d'une analyse de la langue naturelle et sont détectables et instanciables, l'explicitation de liens implicites relève d'une tâche plus complexe que la simple analyse de clause au niveau unitaire.

Il est donc nécessaire d'apporter une touche supplémentaire au seul problème de modélisation afin de permettre la traçabilité entre les éléments du modèle. A partir de ce constat, il nous faut nous poser les questions suivantes : Quelles sont les techniques manuelles ou automatiques de traçabilité ? Quels éléments sont tracés ou à tracer ? Comment reconstruire les liens ?

1.4.3 Formalisation et traçabilité de textes dans le large, un problème hybride

Les techniques de modélisation des exigences d'un côté, et celles de traçabilité des exigences de l'autre sont deux courants relativement différents, bien qu'ils ne soient pas orthogonaux. Se pose cependant la question de la prise en compte dans un seul environnement de ces deux préoccupations.

Cependant si la modélisation et le langage naturel non contraint ne vont guère de paire, les techniques de recherche d'information et de traitement de la langue naturelle ont montré depuis plusieurs années de bonnes dispositions pour la traçabilité des exigences [CHBC+07] avec des approches comme REVERE [SRG02] ou des outils comme Poirot [CHSDZ05] ou RETRO [HDS+07]. Cependant, ces techniques (semi)automatiques de traçabilité ont tendance à considérer l'ensemble des éléments de manière plane, sur des ensembles finis et déjà déterminé d'exigences. Elles ne tiennent donc pas compte de la granularité des documents ou de la diversité des éléments qui composent un référentiel pris dans sa globalité et où ces éléments sont à identifier. Par conséquent, ces techniques, qui génèrent déjà un nombre de faux positifs important dans la proposition de candidats, vont générer encore plus de faux positifs dans notre cas. Si cet effet est intéressant pour obtenir des ensembles de résultats quand il n'existe pas une solution unique, comme dans le cas des référentiels d'exigences de sûreté, cela les rend moins pratiques du fait du grand nombre de faux positifs remontés.

Si on voit que l'une et l'autre de ces approches apportent une réponse partielle aux problèmes de formalisation et de traçabilité que nous avons énoncés, on peut alors s'interroger sur la faisabilité et la viabilité d'une approche hybride combinant les bonnes propriétés de la (méta)modélisation (aspects structurels du domaine) et de la recherche d'information pour apporter une réponse globale au sein d'un même environnement.

La section suivante s'intéresse donc en particulier à ces deux aspects : modélisation des exigences et traçabilité des exigences.

Chapitre 2

Etat de l'art

There are known knowns. These are things we know that we know. There are known unknowns. That is to say, there are things that we know we don't know. But there are also unknown unknowns. There are things we don't know we don't know.
Donald Rumsfeld

Les questions qui nous intéressent et que nous traiterons dans cette section sont directement liées aux problématiques que nous avons définies dans la section précédente, à savoir : un problème de formalisation et de représentation du domaine et un second problème de traçabilité.

Le chapitre se découpe en quatre temps. La première partie du chapitre présente d'un point de vue global l'ingénierie des exigences comme un sous ensemble d'activités dont nous faisons un panorama (2.1) avec un traitement particulier sur la modélisation d'exigences (2.2). Le second temps du chapitre traite de deux aspects de l'ingénierie des exigences qui nous intéressent particulièrement pour la thèse. Nous abordons tout d'abord la question des exigences réglementaires et de la conformité à celles-ci (2.3). Le troisième temps est consacré à la traçabilité des exigences (2.4) et des techniques de recherche d'information pour la traçabilité des exigences (2.5). La dernière partie (2.6) présente une discussion autour de l'état de l'art et des problèmes que nous avons identifiés dans le chapitre 1.

2.1 Ingénierie des exigences

Avant de parler d'ingénierie des exigences, il convient de définir une exigence. Selon le glossaire IEEE [IEE90], une exigence est une condition ou une capacité (fonctionnalité) requise par un utilisateur pour résoudre un problème ou réaliser un objectif, satisfaire un contrat, une norme, une spécification ou un autre type de document formellement imposé.

Définition 2.1 *(a) A condition or capability needed by a user to solve a problem or achieve an objective. (b) A condition or a capability that must be met or possessed by a system to satisfy a contract, standard, specification, or other formally imposed document*

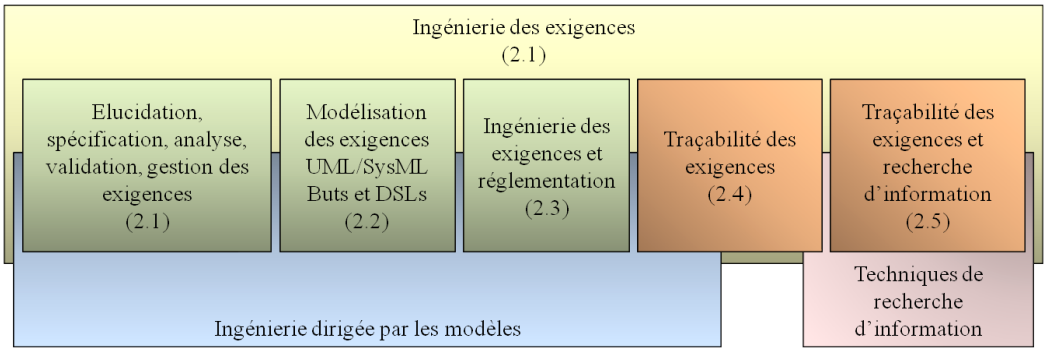


FIGURE 2.1 – Organisation de l'état de l'art

On peut définir l'ingénierie des exigences comme étant la branche du génie logiciel qui se préoccupe des buts, fonctions associées, contraintes posées sur les systèmes logiciels dans le monde réel. Elle s'intéresse également aux relations entre ces facteurs qui affectent les systèmes, la spécification précise du comportement du logiciel et des évolutions de l'ensemble dans le temps et dans les familles de logiciel.

Définition 2.2 *Requirements Engineering is the branch of software engineering concerned with the real-world goals for, functions of, and constraints on software systems. It is also concerned with the relationship of these factor to precise specifications of software behavior, and to their evolution over time and across software families [Zav97].*

Amyot et al, de manière plus informelle dans leurs supports d'enseignement, proposent une définition sous la forme d'activités : "L'ingénierie des exigences est l'ensemble des activités d'élucidation, spécification, analyse et gestion des exigences qui sont à satisfaire dans un système nouveau et évoluant". "L'ingénierie des exigences se préoccupe de l'identification des objectifs d'un système logiciel et du contexte dans lequel celui-ci est mis en oeuvre et utilisé". De la même manière Lamsweerde [vL08] définit l'ingénierie des exigences comme un ensemble d'activités, de l'élucidation, l'analyse et l'évaluation, la spécification, la consolidation et la validation et enfin l'évolution des objectifs, fonctionnalités, contraintes auxquels un système logiciel doit répondre dans un cadre physique et organisationnel défini. Dans leur roadmap de 2000, Nuseibeh et Easterbrook [NE00], ainsi que Cheng et Atlee dans l'article qui leur fait suite en 2007 [CA07], définissent l'ingénierie des exigences autour des activités d'élucidation des exigences, de modélisation et d'analyse des exigences, de communication et de négociation des exigences (uniquement pour Nuseibeh et Easterbrook), de validation des exigences, de leurs évolutions et leur gestion.

Plusieurs éléments sont à observer dans ces définitions. La première observation se porte sur une description qui ferait consensus sur les activités de l'ingénierie des exigences et que nous détaillerons par la suite : élucidation, spécification, modélisation et analyse, validation et gestion des exigences. La seconde est l'aspect fonctionnel de l'exigence, même si les deux grandes normes IEEE sur la spécification d'exigences mettent

l'accent sur les aspects à la fois fonctionnels et non fonctionnels des exigences et parmi ceux-ci les aspects réglementaires comme des contraintes (section 2.4 d'une spécification générique proposée par la norme) portant sur la spécification. La deuxième observation vient de la séparation relative des différentes activités qui définissent l'ingénierie des exigences. Si ces activités sont séparées d'un point de vue recherche académique, dans le cycle de vie d'un projet, ses activités s'entrelaçent et se réitèrent dans un processus continu tel que présenté par Lamsweerde [vL09].

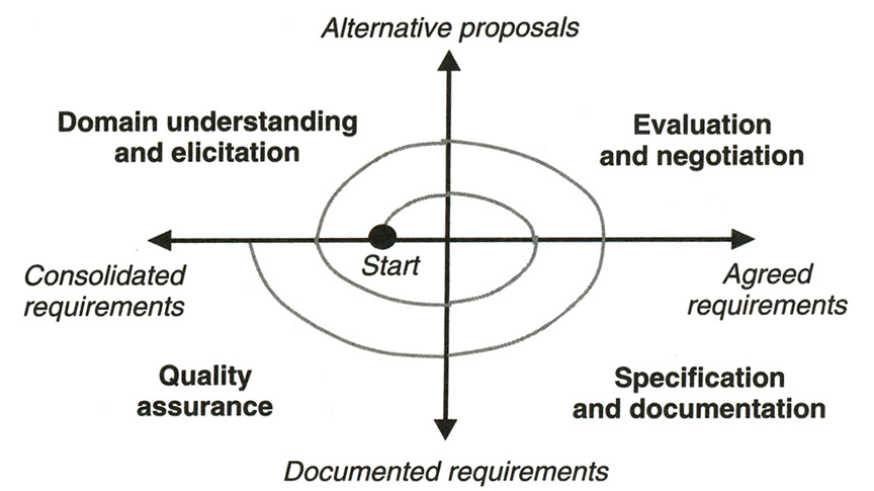


Figure 1.6 *The requirements engineering process*

FIGURE 2.2 – Le processus d'ingénierie des exigences par Lamsweerde

2.1.1 Ingénierie des exigences : un bref panorama

Pour s'assurer que la solution logicielle développée résolve le problème donné, il est indispensable de comprendre quel est le problème à résoudre. Bien que cette affirmation soit triviale, définir le problème est beaucoup plus compliqué qu'il n'y paraît. Il faut découvrir "quel" est le véritable problème qui doit être résolu, "pourquoi" il doit être résolu et "qui" est impliqué ou doit être impliqué dans la résolution de celui-ci. Or ce problème à résoudre n'est pas que de l'ordre du logiciel, il vient d'un contexte beaucoup plus large prenant en compte des problématiques organisationnelles ou/et techniques mais aussi des contraintes de l'environnement autour du système logiciel et du monde réel. Cette rencontre du "monde" et de la "machine" a été développée par Jackson [Jac95] dans laquelle le système logiciel (développé et installé dans une "machine") vient "améliorer le monde" en résolvant un problème du monde. Le monde et la machine ont leurs phénomènes propres mais partagent aussi des phénomènes, ce qui forme l'interface et donc le lieu et les modalités interactions entre le système et le monde.

Bien que du point de vue académique, l'ingénierie des exigences soit une branche du

génie logiciel, celle-ci se préoccupe de comprendre et d'exprimer, analyser, gérer dans le temps les problèmes à résoudre et non pas comment résoudre les problèmes d'un point de vue logiciel. Dans cette section, nous présentons un bref panorama académique de l'ingénierie des exigences et le présentons vis-à-vis de la définition en activité proposé précédemment. Nous y rajoutons les travaux autour de la spécification qui n'étaient pas pris en considération. La partie modélisation et traçabilité (qui est une forme d'analyse) seront discutées de manière plus spécifique.

2.1.1.1 Elucidation des exigences

L'élucidation des exigences comprend un éventail d'activités qui permet la découverte et la compréhension des buts, objectifs et les motivations pour la construction ou la modification d'un système. Elle permet également de découvrir quelles seront les exigences auquel le futur système aura à répondre pour satisfaire les buts/objectifs fixés. L'élucidation des exigences couvre un large spectre qui va de la compréhension des évolutions d'un système dans un environnement maîtrisé, à des problèmes nouveaux qui émergent dans un environnement, ou d'exigences relativement souples, sans contraintes fortes de la part des parties prenantes. Comme le soulignent Cheng et Atlee [CA07] la plupart des travaux en élucidation des exigences visent à fournir des environnements ou des méthodologies pour améliorer les informations sur les exigences. Ces travaux peuvent concerner l'identification des parties prenantes pour s'assurer que toutes les personnes impactées par le système en-cours et à venir soient consultées et leurs objectifs pris en compte durant la phase d'élucidation [SFG99], l'emploi de métaphores [Pot01], scénarios[ABKO04], séances de brainstorming, de théâtre ou autres techniques de créativité [MMH12, MR05] pour aider les parties prenantes à comprendre leurs exigences [Ber02], découvrir des exigences innovantes, proposer des exigences non essentielles mais qui vont améliorer l'acceptabilité, la désirabilité ou la satisfaction de l'utilisateur envers le système ([KSTT84]).

2.1.1.2 Spécification des exigences

Suivant l'activité d'élucidation dans le processus d'ingénierie des exigences en spiral proposé par Lamsweerde (2.2), la phase de spécification vient assoir et documenter / formaliser les résultats de la phase d'élucidation. En fait, ces deux phases s'entrelacent plus qu'elles ne se suivent. C'est une étape clé car elle prend en entrée un ensemble hétéroclite et peu formalisé d'objectifs généraux, d'exigences systèmes, logicielles, contextuelles, de définitions de concepts du domaine, de propriétés de l'environnement. A l'issue de l'étape cet ensemble hétéroclite a été transformé à l'issue de plusieurs itérations en un document d'exigences ou une spécification, qui structure et organise l'ensemble des exigences selon des critères qualité comme par exemple ceux définis dans les différentes normes IEEE 1233 ou 830 sur la spécification d'exigences système ou logicielles.

Nous parcourons brièvement les différentes approches pour la spécification de ces exigences.

L'utilisation de la langue naturelle sans contrainte est l'option la plus simple et la plus évidente pour la spécification des exigences. Cette approche possède de nombreux avantages, parmi lesquels sa simplicité de mise en œuvre, sa richesse d'expressivité, l'absence de barrières techniques ni d'obligation de formation entre les parties prenantes. Néanmoins, cette approche est particulièrement susceptible d'amener des nombreuses ambiguïtés, du bruit, des erreurs de référencement, de l'opacité dans le discours et qui rendent les exigences d'autant plus difficiles à comprendre, mettre en œuvre et vérifier par la suite via des approches automatisées. La langue naturelle est ambigüe par nature [Kam05, Poh10] et n'est clairement pas une bonne approche pour la bonne expression des exigences dans un système. Pohl va même jusqu'à décrire 5 types d'ambigüité : lexicale (le sens des mots), syntaxique (la structure de la phrase), sémantique (le sens de la phrase), référentielle (problème de différence de domaine), approximative (emploi de termes vagues ou approximatifs).

L'utilisation de langages naturels contraints permet de limiter les défauts de l'expression libre. Ces contraintes peuvent porter sur des règles dites "locales" pour la bonne formation de la clause, ou "globales" qui visent à l'organisation du document dans son ensemble. Les règles locales peuvent fournir des canevas pour l'écriture avec l'emploi de règles ou patrons d'écriture (emploi de verbe modaux, utilisation de la forme active, restrictions des synonymes, interdiction des termes génériques, etc.). De nombreux patrons de spécification d'exigences ont ainsi vu le jour pour permettre l'expression systématique d'exigences sous la forme de remplissage de champs. Ces patrons permettent d'exprimer des exigences avec différents niveaux de granularité et niveaux d'analyse automatique pour leur traitement ultérieur ou la vérification et la validation. Les patrons ne se concentrent pas uniquement sur le verbatim de l'exigence mais sur toute l'exigence, la définition d'acteurs, la proposition de métriques pour l'évaluation, la mise en œuvre de la traçabilité, etc.

Ainsi les cellules de Volere de Robertson [RR99] fournissent-elles un ensemble complet de propriétés, au format textuel et hypertexte, autour du simple verbatim de l'exigence même si l'écriture de l'exigence reste à la charge de son auteur, avec les règles locales mentionnées précédemment.

A la même période, Konrad et al. [KC02], ou Dwyer et al. [DAC99] ont présenté d'autres patrons pour les exigences mais avec des vocations différentes. Les patrons de Konrad sont inspirés par les patrons de conception du GOF [GHJV93] et visent à la spécification d'exigences dans un langage formel interprétable (Hydra), à leur analyse par un moteur de validation de modèles (Spin) et à leur visualisation avec le framework Minerva sous la forme de diagrammes UML. A partir de l'exigence textuelle, l'analyse choisit le pattern qui lui convient le mieux afin d'entreprendre de manière assistée la génération et la modélisation des exigences.

Les patrons de Dwyer et al [DAC99] sont quant à eux tournés vers la spécification d'exigences vérifiables par opérations sur des machines à états. A partir du texte d'une exigence, celle-ci est réécrite en déterminant manuellement le type de patron à mettre en œuvre, les propriétés à vérifier de l'exigence, la ré-expression de l'exigence en une expression logique. Les différents types de patrons, et la réécriture d'une exigence sont

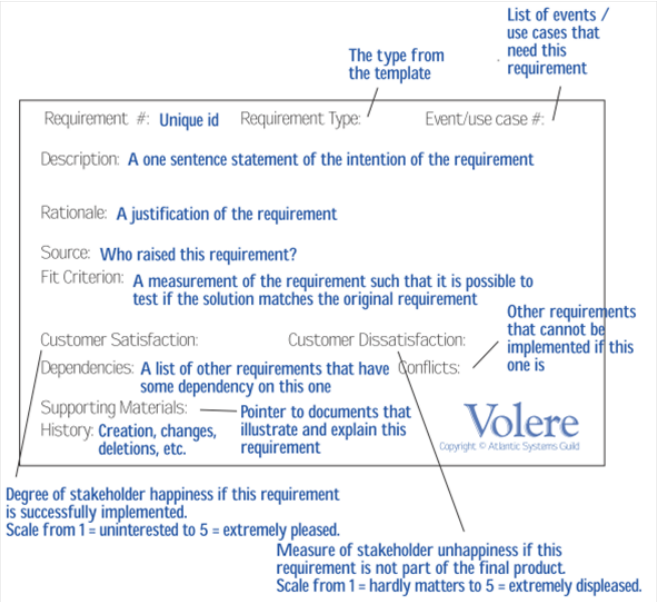


FIGURE 2.3 – Cellules de Volere par Robertson

TABLE 2.1 – Patrons d'exigences de Konrad [KC02]

Actuator-Sensor	How to specify various kinds of sensors and actuators and their relationships to a controller in an embedded system.
Controller Decompose :	How to decompose an embedded system into different components according to their responsibilities.
Monitor-Actuator	How to increase safety and reliability by monitoring actuator behavior for errors.
Fault Handler	How to integrate a fault handler into an embedded system.
Channel	How to arrange communication between two components.
Watchdog	How to monitor a device or system conditions and initiate corrective action(s) if a violation is found.
Examiner	How to monitor a device and store occurring errors.
User Interface	How to specify a user interface that is extensible and reusable.
Mask	How to reduce the burden placed on the computing component when many sensors and actuators are present, whose values need to be sorted or filtered into single values for the computing component.
Moderator	How to provide an interface to support decoupling of complex subsystems.

illustrée dans la figure 2.4

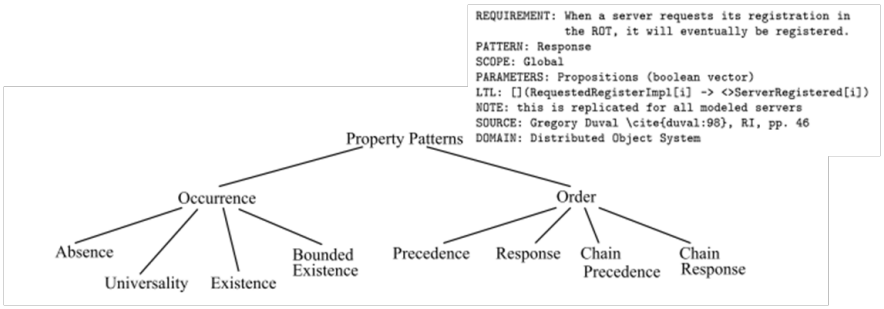


FIGURE 2.4 – Patrons de Dwyer

Depuis 2009, Mavin et al ont développé la méthodologie EARS (Easy Approach to Requirements Syntax) chez Rolls Royce [MWHN09]. Dans EARS, les exigences sont définies selon deux types. Les exigences de fonctionnement normal et les comportements non désirés pour décrire toutes les déviations du domaine de fonctionnement normal.

Dans sa syntaxe, EARS propose une articulation générique autour de blocs : <préconditions (optionel)> <déclenchement (optionnel)> le <nom du système> doit <comportement / réponse du système>. Cette syntaxe définit quatre périmètres d'application d'exigences et sont illustrés dans la figure 2.5 :

- ubiquitaire : valable tout le temps
- dépendant d'un évènement (avec le mot-clé quand/when) : valable au déclenchement d'un évènement particulier
- dépendant d'un état (avec le mot-clé alors que / while) : valable dans un état particulier du système
- dépendant d'une option (avec le mot-clé où / where) : valable si une fonctionnalité du système est présente

Ces différents éléments peuvent se combiner pour former des exigences complexes.

Ces patterns sont présentés comme apportant plus de rigueur et de cohérence dans la spécification, sans être particulièrement difficiles à apprendre et mettre en oeuvre puisqu'ils ne nécessitent aucun outil particulier. En contre partie, les relations entre exigences sont difficiles à mettre en oeuvre puisque la structure de l'exigence ne permet aucune référence extérieure. Si les blocs sont combinables, l'écriture d'exigences particulièrement complexes n'est pas facilitée.

D'autres patrons ou approches de spécifications sont disponibles dans la littérature avec des objectifs encore différents. Par exemple, RELAX [WSB+09, CSBW09] adresse ainsi la question de la spécification des exigences "sur le logiciel" pour les systèmes adaptatifs. RELAX se base sur une approche à partir de langage naturel contrôlé, et propose des opérateurs spécifiques pour l'expression d'incertitudes dans les exigences.

RELAX se base comme les patrons précédents sur les verbes modaux ("shall", "will", "may") pour exprimer les comportements ou les fonctionnalités offertes par le système. Il inclut également des opérateurs classiques de logique temporel comme "eventually",

Ubiquitous Requirement
The <system name> shall <system response>
The laptop shall have a minimum battery life of XXX hours.

Event-driven Requirement
When <trigger> the <system name> shall <system response>
If <optional preconditions> <trigger>, then the <system name> shall <system response>
When the laptop is running and the laptop is closed, the laptop shall enter "powersave" mode.
If the incorrect password is entered, then the laptop shall display XXX warning message.

State-driven Requirement
While <in a specific state> the <system name> shall <system response>
While the laptop is running on the battery and the battery is below XXX % charge, the laptop shall display "low battery".

Option
Where <feature is included> the <system name> shall <system response>
Where the laptop is a "lightweight" model, the laptop shall have a mass of no more than XXX grams.

FIGURE 2.5 – Quatre types d'exigences EARS et leur patron

"until / before / after", ainsi que des opérateurs spécifiques pour l'incertitude comme "as early/late as possible, as many/few as possible [frequency / quantity]", ce qui permettent de préciser si les exigences exprimées sont des invariants, ou doivent être relâchées sous certaines conditions.

RELAX définit aussi des facteurs d'incertitudes autour de l'environnement du système (ENV), des propriétés qui peuvent être observées par le système (MON), les relations entre l'environnement et les propriétés observées par le système (REL) et identifient les dépendances entre les exigences (DEP).

2.1.1.3 Analyse et validation des exigences

Les travaux concernant les deux activités d'analyse des exigences se concentrent autour de l'analyse de la qualité des exigences enregistrées, la détection d'incertitude ou d'ambiguïté [YRG+12, MAS12a]. Certaines analyses portent sur les règles de bonne formation des exigences quant aux critères qualités de précision, de non ambigüité, ou de complétude. D'autres analyses portent sur la découverte d'anomalies [CCMS02], l'analyse de risques [DLR13] ou de conflits dans les modèles de buts ou la comparaison d'exigences dans le cadre de réglementations. Cette considération sera présentée par la suite. Ces analyses incluent également les analyses de risques ou les études d'impact [FKMT05] qui aident à mieux comprendre les exigences et leurs interrelations dans le projet. Il en est de même pour la gestion des priorités dans les exigences pour aider les managers à coordonner et sélectionner au mieux l'ensemble des exigences à implémenter

ensemble ou en priorité [RKH03, WCR10, WRS09].

En ce qui concerne la validation et la vérification des exigences, celle-ci concerne deux aspects particuliers autour des exigences. Il s'agit tout d'abord de vérifier et valider que les exigences, dans leur expression, c'est-à-dire qu'elles reflètent correctement les problèmes et les besoins des parties prenantes. Dans le second cas de figure, il s'agit de validation telle qu'on l'entend en génie logiciel, c'est à dire l'assurance que le système fait ce pour quoi il a été conçu, répond aux exigences exprimées.

2.1.1.4 Gestion des exigences

Changing requirements is as certain as death and taxes ...
variation de la lettre de Benjamin Franklin à Jean-Baptiste Leroy en 1789

Leffingwell et Widrig ont proposé une définition de la gestion des exigences [LW00].

Définition 2.3 *A systematic approach to eliciting, organizing, and documenting the requirement of the system, and a process that establishes and maintains agreement between the customer and the project team on the changing requirements of the system.*

La gestion des exigences regroupe également un ensemble d'activités qui vont de la gestion des exigences elles-même dans leur cycle de vie, leurs évolutions successives dans le temps ou dans les familles de produits. Pour cette activité, il existe de nombreux outils commerciaux pour la gestion du cycle de vie des exigences ou des systèmes (on parle d'outils d'ALM, "Application Lifecycle Management") telles que DOORS chez IBM Rational, ou RequisitePro, Reqtify, Caliber, etc. et qui sont très largement utilisés par les industriels même pour les projets d'ingénierie système de très grosse taille.

En dehors de ces outils, les travaux de recherche s'articulent autour d'environnement ou d'outils pour faciliter ou automatiser la documentation des exigences, mettre en œuvre la traçabilité des exigences, traçabilité entre les exigences et avec les différents artéfacts de développement ou déterminer la variabilité et la maturité [WRK09] ou la propension à évoluer des exigences [MAS12a, WRS09].

2.2 Modélisation d'exigences

Il existe différents types de formalismes ou de représentations pour les exigences, dépendantes de la sémantique que l'on souhaite associer à ces dernières, aux objectifs poursuivis, etc. Nous n'avons pas la prétention de proposer un tour exhaustif de tous les moyens possibles mais ceux qui nous semblent pertinents et proches de nos préoccupations. Avant de démarrer, il est important tout d'abord de signaler qu'il existe différents types d'exigences, en plus des différents objectifs poursuivis en termes de représentation.

2.2.1 Distinctions entre exigences

Il y a eu par exemple la différenciation entre exigences fonctionnelles et non fonctionnelles connues dans la communauté académique et faite, par exemple, dans le standard

IEEE 830 sur la spécification d'exigences logicielles. Une exigence fonctionnelle concerne le service effectif rendu par le système. Le pendant non fonctionnel étant un aspect qualité de service mais pas uniquement. Le standard IEEE 830 distingue ainsi les aspects performances, contraintes de conceptions, d'interfaces ... Cette distinction est assez importante car, dans la plupart des travaux, les auteurs travaillent essentiellement dans l'une ou l'autre des catégories, rarement dans les deux. Cette distinction n'est pas la seule même si elle est prédominante. D'autres catégorisations peuvent (co)exister selon que l'on souhaite "typer" des exigences en fonction des considérations qu'elles adressent (exigences fonctionnelles, de performance, de fiabilité, etc), par exemple la décomposition de Sommerville et Kotonya [SK98] proposée dans la figure 2.6. Cette distinction est faite entre le côté binaire de la satisfaction d'une exigence (si elle est satisfaite ou non), ou le fait qu'il existe une zone grise où l'exigence est plus ou moins satisfaite. Le cas de figure se présente avec des exigences en opposition et devant être négociées.

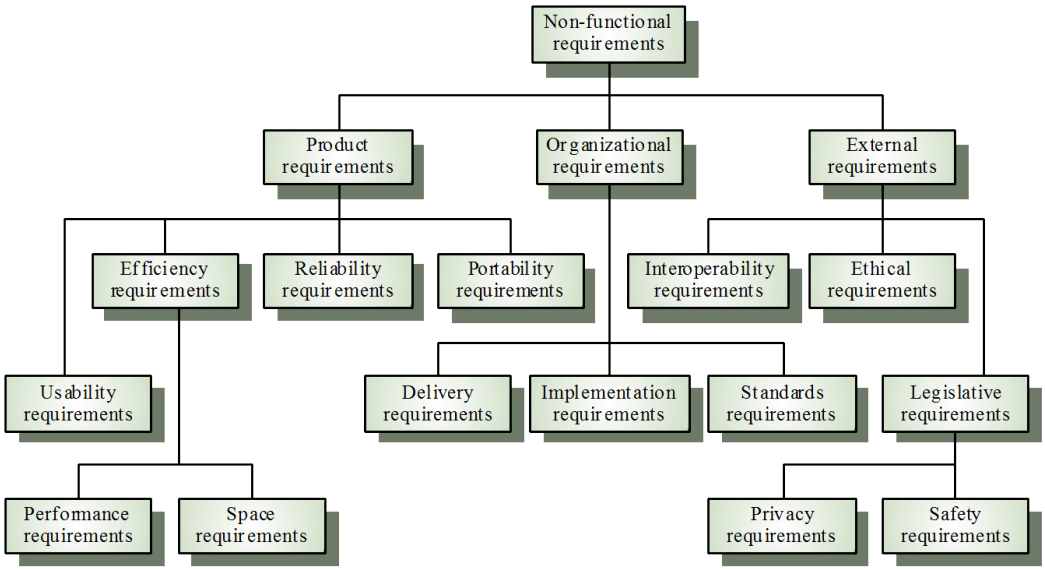


FIGURE 2.6 – Différents types d'exigences non fonctionnelles par Sommerville et Kontonya [SK98]

En termes de représentation des exigences, ces distinctions amènent différentes représentations possibles. Certaines permettent, par exemple, de représenter des exigences fonctionnelles comme des cas d'utilisation dans une modélisation UML. C'est un mode de représentation pratique car il permet de représenter les différents acteurs qui vont être en interaction avec une fonctionnalité/service mais aussi représenter l'élément du système concerné par ce service. Inconvénient d'une telle représentation, elle ne permet pas de représenter les exigences non fonctionnelles. Il existe des approches qui visent à représenter des exigences non fonctionnelles avec ce même mécanisme ou à annoter les cas d'utilisation pour y associer leurs éventuelles contraintes non fonctionnelles.

Nous balayons un ensemble de représentations et de formalisations possibles pour les exigences.

2.2.2 Modélisation des exigences en ingénierie système

L'AFIS [1] est l'association Française d'Ingénierie Système, représentant au niveau français l'INCOSE (International Council on Systems Engineering) [2]. Sa vocation est de promouvoir l'ingénierie système au niveau français et regroupe un certain nombre de membres venant essentiellement du monde industriel. Un de ces groupes de travail (GT) s'est penché sur la problématique de la formalisation et de la traçabilité des exigences en proposant un modèle de données en 2001 (dernière modification 2010).

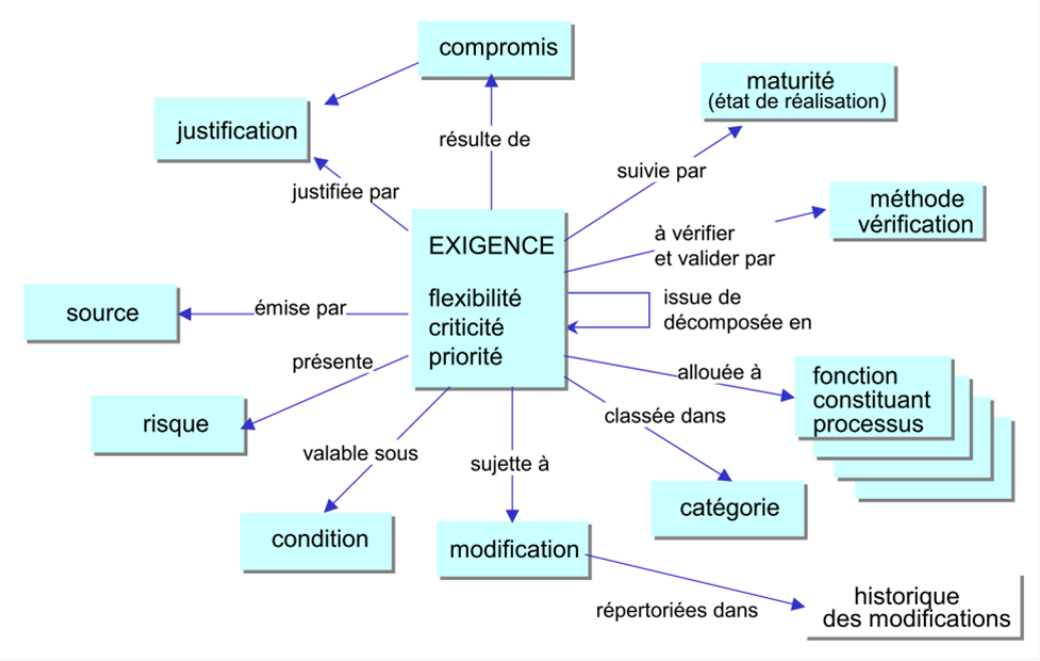


FIGURE 2.7 – Modèle de données des exigences proposé par l'AFIS

Ce modèle de données met l'exigence au centre des préoccupations en plus de lui donner un certain nombre de propriétés largement manipulées dans les projets industriels (criticité, maturité, flexibilité, catégorie, V&V associée). Il fait également la part belle aux questions de traçabilité amont avec une association "supportée_par" vers une justification, ce qui s'apparente dans la communauté académique aux *Requirements rationales* ; mais également traçabilité aval, par des associations "vérifiée_par" ou "allouée_à". L'AFIS reconnaît également la difficulté de manipuler des exigences textuelles et les nécessaires négociations qui amènent les exigences à être ce qu'elles sont ("ré-

1. http://www.afis.fr
2. http://www.incose.org

sulte_de", "émise_par"). En ingénierie système, le système comprend également l'environnement de l'artéfact en devenir. Ces prises en compte peuvent être des contraintes sur le système ("valable_sous"), ou des risques sur l'environnement ("présente") ou des risques de l'environnement sur le système.

Le modèle de données AFIS permet donc une représentation assez riche, même si elle reste au niveau textuel. Une telle représentation ne permet pas cependant une grande souplesse dans la différenciation ou une éventuelle hiérarchisation des exigences. On reste à un niveau d'abstraction relativement bas puisque l'on attaque très rapidement les cas de test et les éléments d'architecture. Une légère souplesse est apportée par l'attribut *Catégorie* dans l'exigence mais dont l'énumération d'échantillons de valeurs possibles laisse plutôt à penser que l'on reste au niveau d'exigences techniques pour l'utilisation du modèle.

2.2.3 UML et la modélisation d'exigences

UML propose deux familles de modèles dans sa spécification : les diagrammes structurels et les diagrammes comportementaux. Les diagrammes structurels, parmi lesquels le diagramme de classe et le diagramme de cas d'utilisation permettent de décrire les concepts du domaine sous la forme de classes, leurs propriétés sous forme d'attributs, le périmètre du système et ses fonctionnalités sous la forme de cas d'utilisation. Comme Yue et al le soulignent [YBL11], le diagramme de cas d'utilisation est de loin le plus utilisé. Le diagramme de cas d'utilisation permet de manipuler dans une représentation les notions d'acteurs, de rôles, des interactions entre acteurs et système, du périmètre du système. Ces cas d'utilisations représentent des comportements attendus du système par certains acteurs. Il y a donc dans UML des mécanismes puissants pour modéliser les exigences et le processus d'ingénierie des exigences [GRS+97, RSA98, ARSM99, NFTJ03a, YBL13] même si l'on se restreint ici aux exigences fonctionnelles. Ces cas d'utilisation peuvent-être organisés et séquencés dans des scénarios ou au travers de use cases maps [AMGK11].

Les scénarios, les diagrammes d'activités, diagrammes de séquence permettent de représenter des intentions, de mettre les exigences dans leur contexte et de permettre de cette façon, un premier niveau d'analyse (détection et résolution de conflit par exemple) [RD11, SAYD12, AGI+13], ou de spécifier par l'exemple, en conjonction des cas d'utilisation ou d'autres approches comme KAOS [DLVL06]. Dans les scénarios, on peut alors visualiser les concepts de ressources, de pré et post-conditions. On peut également dissocier les scénarios positifs des scénarios négatifs, de scénarios alternatifs ... Une fois encore, la perspective fonctionnelle est très majoritairement la seule prise en compte de ce genre de représentation.

2.2.4 SysML et la modélisation d'exigences

UML est perçu comme trop tourné vers le génie logiciel avec notamment un lien très fort avec le paradigme objet et pas assez vers l'ingénierie système. Avec SysML [3], l'OMG

3. http://www.omgsysml.org/

souhaite offrir un langage de modélisation de haut niveau aux ingénieurs système en remplaçant notamment le diagramme de classe par les diagrammes de blocs (BDD) et les diagrammes de blocs internes (IBD - internal block diagram), en refondant un certain nombre de représentations et en en proposant deux nouvelles : le diagramme d'exigences et le diagramme paramétrique ainsi qu'un ensemble de relations pour permettre la traçabilité entre exigences, artéfacts système et éléments de vérification. SysML est diffusé dans le tissu industriel via le support de différents outils commerciaux ou open source. Il est notamment intégré dans TopCased avec le modeleur Eclipse Open Source Papyrus [LTE+09, LST+11]. Ces derniers permettent d'importer des exigences depuis des documents pour les formaliser sous formes d'exigences SysML et d'utiliser les éléments du standard pour supporter la traçabilité dans le cycle de vie du projet.

Le diagramme d'exigences permet de décrire à minima les exigences sous la forme d'un identifiant unique et d'une description textuelle, le reste étant à personnaliser. Il est possible de les raffiner, les dériver, décrire leurs origines à travers des commentaires. La traçabilité peut être assurée via des matrices d'exigences, gérant les raffinements successifs. Une approche forte dans SysML est de permettre dans un même diagramme une approche multi-vues qui permet d'associer une exigence à un cas de test qui viendrait vérifier cette dernière, associer une exigence à un block (une forme d'allocation de l'exigence) qui viendrait la satisfaire ; l'associer à un cas d'utilisation pour un raffinement particulier, ou être décomposée en sous exigences.

L'une des limitations principales de ce standard réside dans l'aspect très abstrait des définitions, de l'ambigüité et des interprétations possibles sur les modèles. Contrairement d'autres approches multi-vues à la KAOS, SysML ne permet pas une construction à la fois simple et claire d'une décomposition hiérarchique d'exigences capable de proposer des alternatives simples. De même elle permet un choix relativement limité de relation et de représentation d'exigence. Par exemple, la relation *refine* entre une exigence et un cas d'utilisation ne peut se faire qu'entre une exigence qui a un caractère fonctionnel et sa représentation sous forme de cas d'utilisation qui , par définition, n'aborde que le côté fonctionnel. La cohérence entre les vues n'est pas assurée de même qu'il est nécessaire de fortement personnaliser SysML pour atteindre les objectifs que l'on se fixe. De même, ce type de représentation, s'il présente l'avantage d'offrir une perspective sur le projet, reste uniquement dans cette perspective descriptive.

SysML, dans la pratique, a donc été peu utilisé tel quel et a été étendu pour prendre en compte les préoccupations ou les analyses relatives aux modèles. C'est le cas, par exemple, chez Goknil et al. avec une extension des exigences et des liens de traçabilité [GKvdB08], ou pour les aspects temporels [GPF12], chez Laleau et al. [LSM+10] ou Bousse et al. [BMC+12] pour une extension vers le langage B, chez Ahmad et al [ABLG12] avec un langage SysML étendu avec KAOS et RELAX [WSB+10, CSBW09] pour la prise en compte d'exigences non fonctionnelles pour les système adaptatifs.

2.2.5 Langages de modélisation dédiés

Il existe une grande famille de langages de modélisation spécifiques dédiés à la modélisation d'exigences. Parmi ceux là figurent les langages orientés buts et populaires dans

la communauté des exigences comme KAOS, i* ou URN et que nous développerons dans la section suivante. Il y a aussi des langages plus spécifiques, dédiés aux problématiques qu'ils adressent.

Parmi ces langages dédiés, on peut noter, un certain nombre de profils UML et SysML. Ainsi, Gocknil et al. présentent un *core metamodel* [GKvdB08] qui vient spécialiser le diagramme d'exigences de SysML en y ajoutant de nouvelles relations et en specialisant les exigences. Par la suite, ils ont proposé une extension à SysML pour la définition et l'analyse d'exigences avec des propriétés temporelles [GPF12].

Panesar et al. dans leur approche CRESCO, proposent un profil UML pour assister la certification et a été développé autour de la norme généraliste de sûreté CEI 61508 [PWSB11]. Ils y définissent ainsi les différentes activités liées à la certification.

Dans un contexte différent, ils proposent SafetyMet, un métamodèle basé sur Ecore [VPW13] et qui se veut plus généraliste. SafetyMet se base sur l'analyse de quatre normes de sûreté de différents domaine (aéronautique avec la DO-178C, automobile avec la norme ISO 26262, ferroviaire avec la norme EN 50128, et la norme CEI 61508). Ces deux métamodèles ont la particularité d'être orientés autour de la modélisation d'activités et des éléments en entrée et en sortie de ces activités pour la certification.

Vicente-Chicote et al. [VCMÁ07] visent la spécification avec REMM-Studio. Ils proposent ainsi un métamodèle généraliste, définissant deux types d'exigences, systèmes et logicielles mais spécialisent ces exigences via un attribut. Si les exigences de REMM ne possèdent que des liens de dépendances entre elles. Le métamodèle ne considère que l'aspect spécification des exigences.

Helming et al. dans Unicase [LNHK11, HK10] visent, quant à eux, la perspective gestion des exigences. Dans Unicase, l'accent est mis sur l'intégration des exigences dans le processus métier et dans la gestion de version. Les exigences sont représentées sous la forme de de cas d'utilisation ou de scénarios et sont stockés dans un dépôt et sont par la suite rappelés dans différents modèles. Les auteurs mettent ainsi en avant EMFStore qui leur permet de gérer leurs modèles en version.

Brottier et al. proposent à travers la plate-forme R2A, un outil pour la spécification formelle d'exigences à partir de spécifications partielles d'exigences. Ces spécifications peuvent s'exprimer sous la forme de langue naturelle contrainte avec le langage RDL, ou de modèles conformes à leur métamodèles. Ces différents modèles sont ensuite fusionnés en un modèle conforme au métamodèle RM qui propose différentes vues sur l'ensemble de la spécification [BBT+07]. R2A s'est dérivé sur plusieurs cas d'application avec, par exemple, la détection d'incohérences dans la spécification et l'analyse formelle d'exigences [PBBT09], la validation de modèles d'exigences via la simulation [BNT07], ou encore la génération de cas de test pour les transformations de modèles [BFS+06], ou encore pour le test de lignes de produits [NFTJ03a].

2.2.6 Modèles de buts et approches orientées buts

2.2.6.1 Modèles KAOS

L'approche KAOS (Knowledge Analysis in autOmated Specification) résulte des travaux des universités de Louvain et de l'Oregon. [vL09] C'est une approche multi-vues (modèle de buts, de responsabilité, d'objets, de comportement, etc) qui permet d'élucider, spécifier, un modèle d'exigences sous une forme structurée et hiérarchique. Dans KAOS, les buts sont raffinés successivement en sous-but dans un graphe ET/OU jusqu'à être assignés à un/des agent(s). Un but assigné à un seul agent du système se définit comme une exigence. Un but assigné à un agent de l'environnement du système est définie comme une attente. Un sous-but participe (achieve, maintain, avoid) de manière positive ou négative à l'accomplissement de son but parent. Tout cet ensemble permet de mener des spécifications, des analyses de risque, d'analyse d'alternatives, etc. Il est donc important de dissocier les buts des exigences qui sont un moyen d'accomplir les premiers.

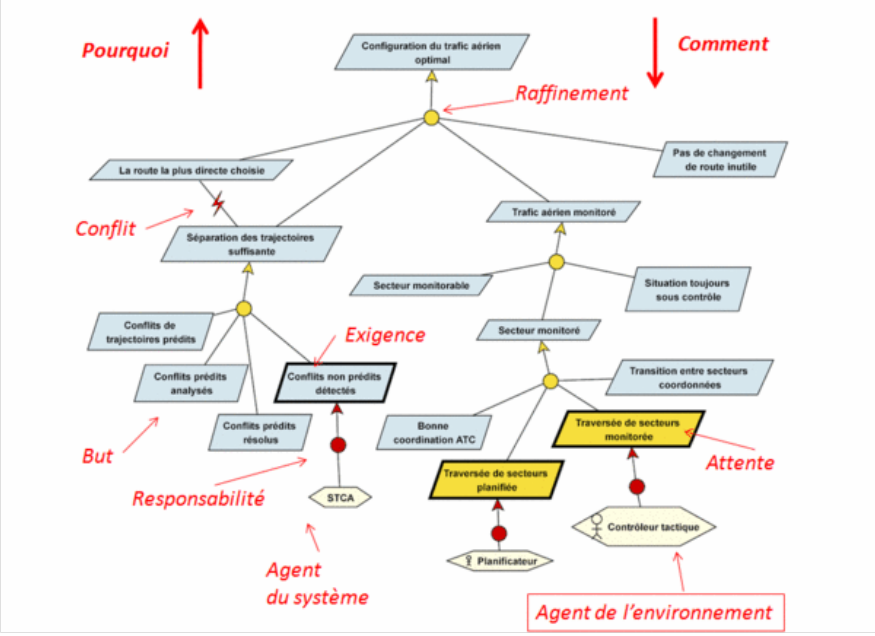


FIGURE 2.8 – Hiérarchie de buts et relations dans un modèle de buts

Dans KAOS, les buts sont les principales entités pour la découverte, l'élaboration et l'analyse des exigences. Les buts sont des « déclarations d'intention » dont la satisfaction demande la coopération d'agents tant au niveau du logiciel que dans son environnement. Tout le système est représenté en de multiples points de vue (responsabilité, opérationnalisation, comportement . . .) avec les liens de traçabilité correspondants, afin de couvrir l'ensemble du système dans son environnement. La méthode KAOS a été im-

plémentée dans une plate-forme, Objectiver[4], qui a été utilisée dans plusieurs projets industriels.

2.2.6.2 Modèles i*

i* est une approche concurrente développée par l'université de Toronto [Yu97]. Sa vocation principale est de comprendre le domaine du problème (et son périmètre) et s'articule essentiellement autour de la description des dépendances entre les acteurs du système. Les modèles i* utilisent principalement quatre éléments : les buts, les "qualités" (*soft goals* qui ne sont pas forcément à rapprocher de la dichotomie fonctionnel / non fonctionnel des exigences), les tâches et les ressources. L'idée principale d'i* tourne autour de l'acteur et de ses intentions. Les acteurs organisationnels sont vus comme ayant des intentions, des buts, des croyances, des compétences et des obligations. Les acteurs dépendent les uns des autres pour les buts à accomplir. Les possibles solutions pour y parvenir sont appelées des stratégies, et comprennent les tâches à réaliser et les ressources à requérir ou fournir. Grâce à cette dépendance envers les autres, un acteur peut résoudre un but qui lui était difficile ou impossible à résoudre seul. D'un autre côté, un acteur devient vulnérable si l'acteur dont il dépend pour résoudre son but ne délivre pas sa contribution. Cette notion d'acteur est donc centrale puisqu'elle montre les intentions et les interactions (positives et négatives) dans l'environnement.

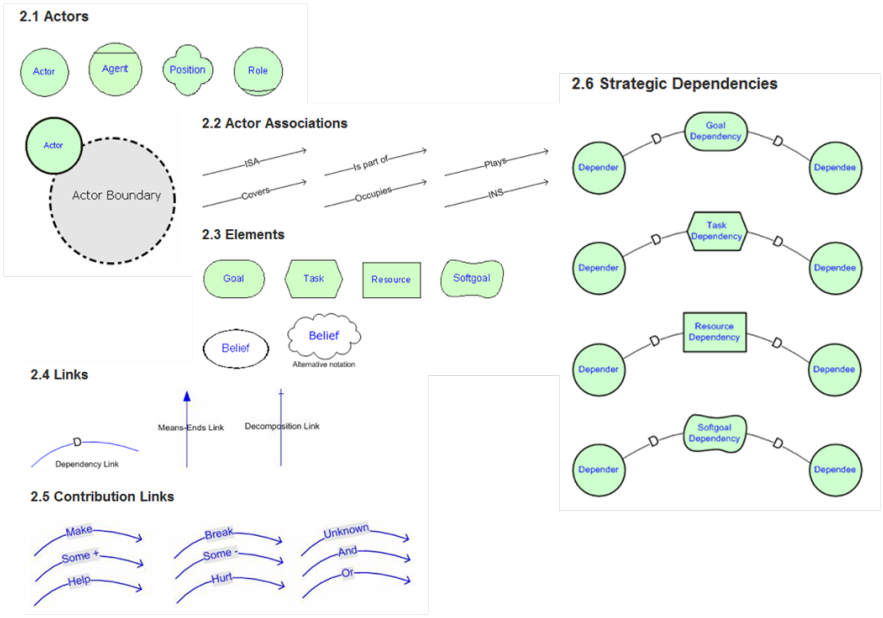


FIGURE 2.9 – Syntaxe du langage de modélisation i*

i* se divise en deux modèles particuliers : un modèle de dépendances stratégiques

4. http ://www.objectiver.com/

qui explicite le contexte organisationnel et les dépendances entre les acteurs, et un modèle de justification (Strategic Rationales) qui contient le premier modèle et exprime les raisons (*rationales*) des liens de dépendances entre acteurs ainsi que des informations sur la façon dont les acteurs accomplissent leurs buts, ou sur les éléments impactant la résolution des buts. Comparés aux modèles de dépendances, les modèles de justification donnent une vue plus détaillée en prenant aussi en compte les relations et les intentions internes, propre à chaque acteur. Ces éléments sont reliés par des liens de type "moyens pour une fin" et de décomposition de tâches. Ces liens indiquent les raisons pour lesquelles un acteur s'engagerait dans une tâche, poursuivrait un but, utiliserait ou aurait besoin d'une ressource, ou rechercherait une qualité à atteindre. Ces modèles permettent de décrire les intérêts des parties prenantes, leurs préoccupations, leurs interactions avec l'environnement et ainsi d'adresser la compréhension de l'environnement et la phase d'élucidation des exigences.

2.2.6.3 User Requirements Notation(URN)

URN (User Requirements Notation) est un langage de modélisation développé à l'université d'Ottawa par Amyot et al [AM11] et normalisé par l'ITU (International Telecommunication Union) depuis 2008. URN se base à la fois sur GRL (Goal-oriented Requirements Language, lui même basé sur i*) et sur le framework UCM (Use Case Maps).

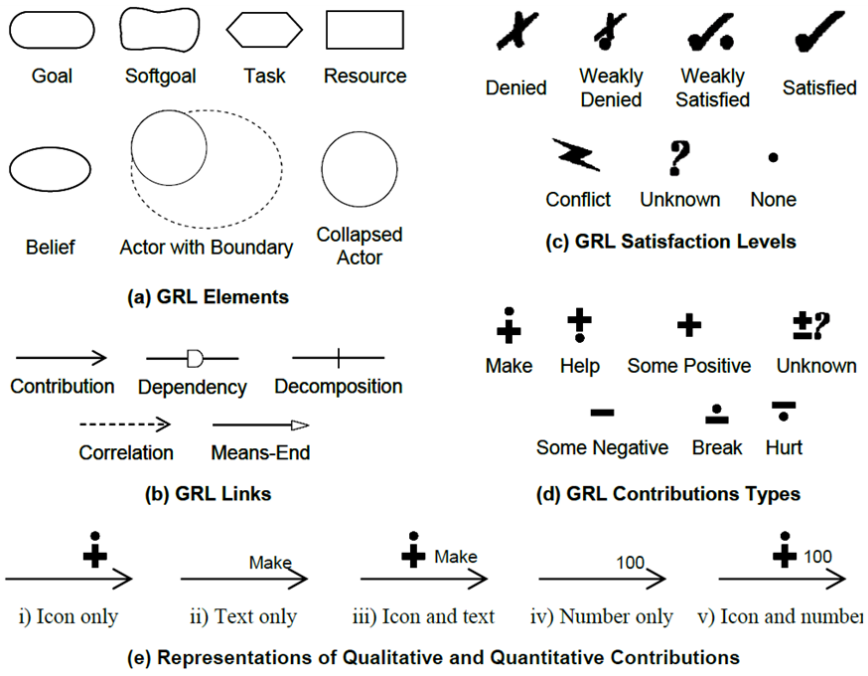


FIGURE 2.10 – Syntaxe du langage de modélisation GRL

Les *use case maps* permettent quant à eux, d'organiser et de scénariser les différents cas d'utilisation, tout en conservant une vue sur les acteurs du système.

2.2.6.4 Discussion autour des approches orientées but

Une approche systématique d'utilisation d'une approche orientée buts à la KAOS pour la modélisation d'exigences réglementaires serait d'utiliser la structure documentaire dans laquelle les exigences s'organisent comme hiérarchie de buts et de structurer les exigences/attentes dans les feuilles de l'arbre construit. La principale limitation d'une telle construction est la relative décorrélation entre la hiérarchie de buts et la logique structurelle du document et dont les préoccupations sont différentes. Ces approches sont définies pour les phases d'élucidation des exigences et pour la proposition d'un document de spécification. Dans notre cas, les documents réglementaires sont déjà écrits et forment la base de départ de notre travail.

Si on prenait en considération la famille i* et URN, la transposition entre les acteurs et les systèmes sur lesquels portent les exigences serait une possibilité. Or, comme nous le mentionnions précédemment, les exigences de sûreté expriment des objectifs ou des moyens. Leurs transpositions en terme de tâches et de ressources voire de scénarios dans des use case maps, si elle est possibles n'est pas forcément intuitive ni possible de manière systématique à l'échelle d'un corpus de documents ou même d'une norme ou d'un document réglementaire. Encore une fois, la décomposition et le périmètre de chaque acteur entre en conflit avec la structure documentaire d'un texte déjà écrit.

La notion d'alternative serait assez restreinte. De la même manière, la notion de contribution (ou de participation) positive ou négative commune à toutes les approches orientées buts ou la notion de conflit est peu évidente à utiliser.

2.3 Ingénierie des exigences et réglementation

2.3.1 Nature des exigences réglementaires

Toutes les exigences dont il est question pour l'instant n'expriment aucune propriété fonctionnelle sur les systèmes mais des contraintes importantes sur ceux-ci. Il ne s'agit pas non plus d'exigences de performance ou de qualité au caractère non fonctionnel classique car exprimées à un niveau d'abstraction beaucoup plus élevées, mais plutôt d'objectifs ou des moyens à mettre en oeuvre pour atteindre un objectif particulier.

Par exemple, les exigences liées à la diversité ou à l'indépendance entre les lignes de défense sont des exigences relativement génériques, puisqu'elles s'appliquent aux systèmes qui ont besoin de vérifier ces propriétés, et ont un impact majeur sur l'architecture du système sans pour autant avoir une influence particulière du point de vue fonctionnel. De la même manière, les processus d'assurance qualité ou de validation et vérification ou de documentation revêtent une importance pour la sûreté alors qu'elles n'ont aucun impact sur le comportement du système en terme de fonction réalisée, de performance, de maintenabilité, ou de disponibilité. Par contre, elles assurent un certain niveau de fiabilité dans la démarche de conception et de validation du système.

Dans la pratique, toutes ces clauses sont considérées comme des exigences et sont à traiter comme telles. Ces exigences sont donc relativement éloignées des définitions des exigences et des spécifications d'exigences que l'on retrouve par exemple dans les normes IEEE 830 (spécification d'exigences logicielles) ou 1233 (spécification d'exigences systèmes), avec des critères attendus de clarté, précision, non ambiguïté, vérifiabilité, testabilité, etc. Elles le sont d'autant moins que ces "anti-propriétés" sont parfois involontaires mais aussi parfaitement volontaires pour laisser la place à l'interprétation de l'exigence [BVA06, Kam05].

Ces exigences sont "subies" par les industriels dans le sens où il ne s'agit pas d'exigences qu'ils ont exprimées mais qui ont été écrites par des tiers. A la différence des textes réglementaires, les exigences des normes sont d'application volontaire (sauf lorsque l'application d'une norme est exigée par une autorité) mais sont également publiées par des organismes tiers. Par conséquent, les industriels n'ont pas la maîtrise de ces exigences. A la différence des spécifications qui peuvent être revues et modifiées, les exigences réglementaires suivent un cycle de vie indépendant des projets.

2.3.2 Ingénierie des exigences et conformité avec la réglementation

Les travaux autour de la conformité des systèmes vis-à-vis d'exigences réglementaires, ou de la loi, se situent dans un périmètre assez restreint. On les retrouve notamment autour de la réglementation américaine dans le domaine médical et notamment autour de *HIPAA* (Health Insurance Privacy Accessibility Act - 1996) et de ses évolutions successives comme *HITECH* (Health Information Technology for Economic and Clinical Health act 2007), ou encore les mesures incitatives de mise en œuvre de la réglementation par les éditeurs logiciels (*Meaningful Use stage 1* - 2010, stage 2 - 2012) et enfin le *HIPAA omnibus Regulations* de 2012 qui compile l'ensemble de ces textes pour la communauté nord américaine. Ces travaux sont majoritairement portés par les travaux de Antón et al, Breaux et al et Amyot et al [BABD09, BAD08, BVA06, MOA09, MOHA10, MSOA11, MA09, MAS12a, MAS11a, MAS+12b, YA10, GB12, GB13, BG13, GAP07]. Dans ces travaux, les analyses portent sur la conformité vis-à-vis de points précis de la réglementation, l'élaboration de méthodologies ou de taxonomies pour la comparaison d'exigences et, pour les travaux les plus récents, la prise en compte de l'évolution de la réglementation ainsi que la comparaison des différences dans la réglementation américaine et des différences inter-états. L'évaluation de la conformité elle-même reste essentiellement une tâche manuelle ou se base sur les approches à contraintes [BVA06, MLHS+11].

Les travaux autour de la conformité ont aussi été entrepris par Mylopoulos et al [KZB+07, KZB+08] mais avec un outillage supplémentaire basé sur l'extraction d'information lexicale sur les exigences et la constitution de graphes pour l'analyse de ces exigences. Ingolfo et al [ISM11] se basent sur des exigences déjà modélisées en i* et avec le framework NFR (Non Functional Requirements) de Chung et al [MCN92] et le framework Nomos [SJI+12] pour revoir les exigences, préparer la justification et démontrer la conformité vis-à-vis de la réglementation.

D'autres secteurs d'activités sont concernés comme la réglementation du transport au Canada [BCS+12] ou la réglementation nucléaire finlandaise [URMT11, RMTV11].

Dans ces cas de figure, les objectifs, à travers la modélisation avec GRL [AMGK11] ou de patrons de spécification comme EARS [MWHN09] est d'orienter la réécriture de ces textes réglementaires dans le processus de mise à jour de ces textes par les autorités ou institutions concernées.

2.3.3 Conformité des exigences et modélisation

En plus des travaux sur les exigences et la loi, un certain nombres de travaux s'articulent autours des approches MDE et la certification vis-à-vis de normes internationales. Des parties de la norme CEI 61508 (sûreté de fonctionnement dans le sens général) ou de la RTCA DO 178-B pour la sûreté de fonctionnement du logiciel dans l'aéronautique, pour lesquelles la conformité est exigée, ont des propriétés qui peuvent être représentées de manière informatique et vérifiées de manière plus automatique. Cet axe de travail reste peu abordé dans la littérature. On peut noter les travaux de Panesar et al autour de la CEI 61508 [PWSB11] et Zoughbi et al [ZBL07, ZBL11] autour de la DO 178-B. Ces deux travaux portent sur la même démarche de proposer un profil spécifique à la norme et d'assister un expert dans la démarche d'instantiation du profil pour modéliser les activités prescrites par les normes.

Panesar et al [PWKSB11] proposent l'outil CRESCO pour la génération de dépôts de "preuves" de sûreté prenant en entrée des modèles conceptuels de domaines sous la forme de profils UML. Les éléments du profil constituent alors un schéma de base de données , schéma utilisé pour la génération du dépôt, de l'outil graphique pour la manipulation des objets, et pour la collecte des différentes preuves pour les tâches de certification. Dans le projet OPENCOSS, de la Vara et Panesar définissent les contours du metamodèle SafetyMet [VPW13] pour proposer un framework de certification autour d'un ensemble de normes de domaines différents. Ce métamodèle est présenté dans la figure 2.11.

Ce métamodèle a été construit à partir de la collecte de concepts de plusieurs normes majeures pour l'aéronautique (DO-178C), l'automobile (ISO 26262), le ferroviaire (EN 50128), la norme généraliste pour les exigences sur le logiciel classé de sûreté CEI 61508.

Toutes les réglementations ne se ressemblent pas. Si OPENCOSS promeut une diversité de domaines, il en reste que les normes utilisées sont très proches les unes des autres. La CEI 61508 et la DO 178-B (et sa nouvelle édition DO-178C) sont deux normes relativement similaires, définissant des niveaux de sûreté (safety integrity levels), et s'organisant autour d'activités, de documents en entrée et en sortie. Il en va de même avec les normes ISO 26262 et EN50128 qui sont des spécialisations de la norme CEI 61508 pour leur domaine. De plus, en dehors de la norme CEI 61508 générique, les trois autres normes sont les seules "grandes" normes de leur domaine, tandis que le monde nucléaire possède toute une hiérarchie de normes et traitent spécifiquement de certains aspects de la sûreté dans des documents spécifiques.

L'OMG étudie actuellement deux recommandations autour de deux métamodèles pour la certification de logiciels importants pour la sûreté. Ces deux métamodèles s'articulent autour des questions de "*preuves*" (SACM [OMG13]- SAEM [OMG10]) et de "justification". Ces métamodèles se concentrent sur la définition de "*claims*" (une pro-

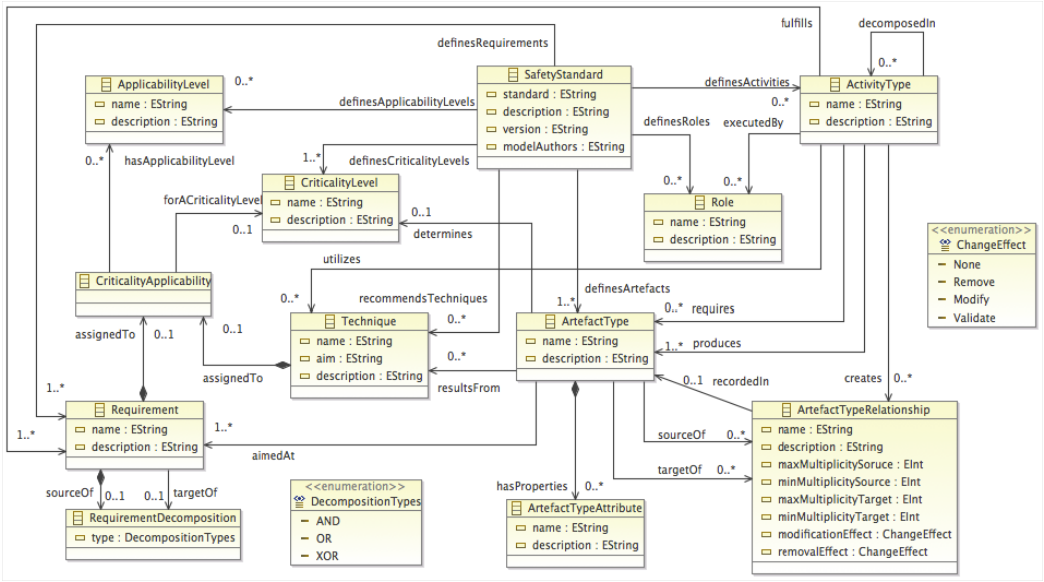


FIGURE 2.11 – Métamodèle SafetyMet

position à propos de la satisfaction d'une exigences), "d'*evidences*" (preuves) venant étayer ("*argumentation*") la proposition. Ils explicitent entre autres, le cycle de vie, les acteurs, ou des critères comme la confiance, le niveau de sûreté ou des propriétés plus classiques pour la gestion du cycle de vie de ces éléments. S'ils présentent des éléments utiles à la certification et à la compréhension des mécanismes pour la certification, ces deux métamodèles ne prennent pas en compte les exigences ni leur organisation.

2.4 Traçabilité des exigences

La définition la plus souvent rencontrée dans la littérature sur les exigences est celle proposée par Gotel et Finklestein [GF94].

Définition 2.4 *Requirements traceability refers to the ability to describe and follow the life of a requirement, in both a forwards and backwards direction (i.e., from its origins, through its development and specification, to its subsequent deployment and use, and through all periods of on-going refinement and iteration in any of these phases)."*

La traçabilité des exigences consiste à suivre l'évolution des exigences de manière descendante, vers leur implémentation et leur vérification, mais aussi de manière ascendante, du système vers ses exigences, et d'identifier les raisons qui ont conduit à la spécification de ces exigences.

Le glossaire IEEE [IEE90], un peu plus ancien (1990) propose quant à lui deux définitions de la traçabilité mais au niveau du développement logiciel.

Définition 2.5 *(1) The degree to which a relationship can be established between two or more products of the development process, especially products having a predecessor-successor or master-subordinate relationship to one another ; for example, the degree to which the requirements and design of a given software component match. See also : consistency.*

Définition 2.6 *(2) The degree to which each element in a software development product establishes its reason for existing ; for example, the degree to which each element in a bubble chart references the requirement that it satisfies.*

Ces définitions peuvent être complétées par la proposition de Gotel et Morris [GM11] qui, en capitalisant plusieurs approches de domaines très différents, définissent la traçabilité comme la possibilité d'interpréter et d'associer des signes distinctifs laissés par un individu. Il faut donc voir la traçabilité en termes d'entité à tracer, de signes permettant d'identifier cette entité dans un environnement particulier (par exemple au sein du corpus d'exigences réglementaires pour les systèmes de contrôle-commande), et d'associer cette collection de signes dans une trace cohérente. Et ce, aussi dans ses raffinements successifs vers la réalisation effective du but recherché (traçabilité aval) qu'en direction des justifications de cette exigence (traçabilité amont).

La traçabilité des exigences est une tâche difficile [CA07, vL08, GCHH+12] de par l'ambigüité des exigences [BVA06] : ambigüité intentionnelle (comme dans le cas des standards et des normes) ou involontaire, de part la complexité intrinsèque de la manipulation du langage naturel [Kam05]. Les liens de traçabilité peuvent également être explicites (références explicites à d'autres standards, allocation d'une exigence à un système, etc) ou implicites, comme dans le cas des pratiques spécifiques à un domaine dans un contexte donné. Elle est également difficile de par les différentes transformations qui s'effectuent entre les raffinements des exigences, et le passage entre le domaine du problème, c'est-à-dire l'exigence ; et le domaine de solution, c'est-à-dire ce qui a été mis en œuvre pour répondre à cette exigence.

Il y a donc plusieurs aspects à considérer pour la traçabilité :

- La représentation de la sémantique de la traçabilité, d'un point de vue générique ou au niveau d'un langage de modélisation avec des primitives de type *refinement / deriveReq / satisfy* dans SysML [OMG12] ; des raffinements ET/OU et des liens dans des systèmes multi-vues comme KAOS [vL09] avec des liens d'allocation, de responsabilité, de conflit, etc qui introduisent une sémantique dans les traces.
- La traçabilité en elle-même, c'est-à-dire, les moyens que l'on met en œuvre pour déterminer les traces et les exploiter.

2.4.1 Sémantique et représentation de la traçabilité

Dans les projets, la traçabilité des exigences est souvent abordée à travers la gestion des exigences et des outils de gestion d'exigences ou de gestion de cycle de vie de projet (Application Lifecyle Management ALM) tels que IBM Rational Doors, Reqtify de Geensoft, IBM Rational RequisitePro, Borland Caliber, Polarion Requirements, Unicase [49], etc. Il s'agit essentiellement de mécanismes de référencement d'exigences

(avec des identifiants uniques), de liens hypertextes, ou encore d'allocations à travers des matrices de traçabilité telles que celles utilisées dans des langages comme SysML [OMG12] par exemple. Paradoxalement, ces outils tiennent assez peu compte de la diversité des exigences manipulables ou des éléments à tracer ni du type de ces relations. Cette traçabilité ne porte donc que peu d'information. De même, elle reste limitée dans le temps, offre peu de réutilisabilité comme le soulignent Mäder et Cleland [MCH10]. Ces techniques ne concernent pas les liens implicites de traçabilité et notamment les questions de conformités vis-à-vis de réglementations, écrites en langue naturelle et pour lesquelles la traçabilité doit être démontrée. Nous développons cet aspect par la suite.

2.4.1.1 Matrices de traçabilité

Les matrices de traçabilité sont des tableaux à deux dimensions qui représentent les liens entre deux ensembles d'artéfacts, qu'il s'agisse d'exigences, d'éléments de conception, de cas de test, de blocs d'architecture. Les lignes et les colonnes représentent un type d'artéfact et les cellules vides ou marquées qui marquent l'intersection entre ces lignes et colonnes dénotent l'absence ou la présence de lien de traçabilité entre ces deux éléments.

Cette représentation relativement sommaire est facilement compréhensible par tous ses utilisateurs, même non techniques, et peut être enrichie en diversifiant les informations contenues dans les cellules pour exprimer une variété de liens ou fournir des informations additionnelles à la seule absence ou présence effective d'un lien. C'est aussi la forme la plus traditionnellement employée pour visualiser la traçabilité entre éléments d'un point de vue assez général. La principale limitation de cette approche est qu'elle passe difficilement à l'échelle dans des projets importants avec une matrice qui augmente d'autant et qui perd en lisibilité. De plus les informations supplémentaires sur la traçabilité se retrouvent en dehors de la matrice elle même. Cette séparation et l'aspect bidimensionnel de la matrice imposent dès lors l'emploi et le maintien en cohérence de plusieurs matrices pour relier les différents éléments du cycle de vie : une matrice pour les liens entre artéfacts d'une même famille (par exemple pour le raffinement d'exigences), et entre différentes familles (exigences vers l'architecture, exigences vers les éléments de validation et de vérification, etc.) Suivre de manière récursive le cycle de vie d'une exigence et les éléments qui la concernent est rendu par conséquent extrêmement compliqué dès que le projet grossit en taille. Il n'y a pas de notion de hiérarchie ou de raffinement ou alors de manière indirecte dans la mise en place de règles pour la dénomination des éléments, mais qui n'ont rien à voir avec la matrice elle même. De la même manière les liens n-aires sont difficiles à interpréter [WP10].

2.4.1.2 Références croisées

Les liens de traçabilité peuvent aussi s'exprimer sous la forme de références croisées entre éléments. Dans sa forme la plus simple, il peut s'agir d'un renvoi simple dans le texte vers la partie concernée (par exemple : "voir la section 2.2.1 du document") ou

s'enrichissant de liens hypertextes fournis par les outils de traitement de texte afin de naviguer le long du lien. Une alternative au lien dans le texte peut être d'ajouter des metadonnées sur l'artéfact (dans une menu contextuel) ou en encapsulant les artéfacts dans des objets et en fournissant des propriétés additionnelles telles que les identifiants et les références croisées. De la même manière, la navigation se fait dans un seul sens ou demande un mécanisme de réflexivité. Comparées aux matrices, les références croisées offrent des informations d'un point de vue local (l'élément concerné), la matrice offrant des informations d'un point de vue global. S'il est possible de savoir quels sont les liens qui entrent ou qui sortent pour une traçabilité amont ou avale, la visualisation et la navigation récursive sont difficiles voire impossible à mettre en œuvre [WP10].

2.4.1.3 Arcs/arêtes de graphes

Dans les approches dirigées par les modèles, les liens de traçabilité sont exprimés comme des éléments du modèle définis par leur métamodèle. Les différents éléments du modèle peuvent alors former un graphe où les éléments à tracer forment les nœuds du graphe et les liens de traçabilité sont les arcs ou arêtes (selon que le liens soit directionnel ou non) du graphe. Il est dès lors possible de définir des sémantiques différentes selon les types de liens de traçabilité définis mais aussi faciliter la navigation en suivant les arcs du graphe et proposer à la fois des informations globales et locales. Ces liens de traçabilité sont eux-mêmes des éléments à part entière des modèles et posséder leurs propres propriétés.

En ingénierie des exigences, ce type de représentation a donné lieu à la définition de différents types de liens, par exemple pour la création de différents types d'interactions entre exigences représentées dans des features par Zhang et al [ZMZ05] ou une taxonomie de références croisées par Maxwell et al [MAS11b] pour l'epxlicitation de liens de conflits entre exigences lors de la comparaison de réglementations. On retrouve également les différents liens hiérarchiques ou de contributions / participations / allocations que l'on retrouve dans la syntaxe des approches de modélisation. C'est le cas des différents liens dans le diagramme d'exigences du langage SysML, ou alors dans les approches orientées buts à la KAOS, i*, ou URN que nous avons développé précédemment. On retrouve encore d'autres types de liens dans les modèles de variabilité comme le diagramme de fonctionnalités de Kang [KCH+90] dont les arcs représentent à la fois la hiérarchie de fonctionnalités mais aussi une sémantique dans le regroupement et la sélection des fonctionnalités.

Ces graphes se retrouvent également dans un certain nombre d'outils tels que PROART de Pohl [Poh97] qui propose une vue en étoile qui montre les différentes dépendances d'un élément et de naviguer progressivement parmi ces dépendances. De manière plus sommaire, certains outils considèrent juste la visualisation d'identifiants, d'icônes ou de "boîtes" et de leurs dépendances. On les retrouve dans TOOR [PG96], dans Process Knowledge Tracer (PKTracer) pour la compréhension des influences et des éléments de connaissances dans le cadre de développement collaboratif distribué chez Ramesh et al [MR07, Ram02] ou dans EGRET (Eclipse-based Global REquirements Tool) de Sinha et al [SSC06]. Chez ces derniers, les liens permettent, à la façon de i*, de comprendre

en plus des dépendances, les relations organisationnelles et sociales dans le projet dans des contextes collaboratifs.

2.5 Recherche d'information pour la traçabilité des exigences : solutions semi-automatiques

2.5.1 Introduction à la recherche d'information

Une façon de réduire le coût de la traçabilité manuelle est de considérer l'extraction des liens de traçabilité entre éléments à partir de leur description textuelle en utilisant les méthodes de recherche d'information. Ces approches ont notamment été développées dans le monde des moteurs de recherche sur internet et ont atteint un certain degré de maturité.

Les approches de recherche d'information se basent généralement sur des extraits de textes et ne reposent pas sur les propriétés structurelles des documents d'entrée [BYRN+99]. Les résultats d'une recherche d'information consiste à retourner un ensemble de *documents* pertinents, relativement à une requête, parmi un ensemble de documents constituant un *corpus*. On y parvient en extrayant les termes clés de chaque document du corpus lors d'une première phase *d'indexation* et de calculer la similarité entre les termes de la requête et ces mots clés de chaque document du corpus. Une liste de documents *candidats* est alors présentée à l'utilisateur à qui il revient de valider au final, les documents pertinents vis-à-vis de sa requête.

Appliquée à la traçabilité des exigences, le corpus est construit autour des éléments que l'on souhaite tracer. L'extraction des mots-clés dépend du type et du format des artéfacts. Dans le cas d'artéfacts textuels comme pour les exigences, ces mots sont extraits en éliminant les mots génériques des documents (*stop words* en anglais ou mots vides), c'est à dire les mots communs, les déterminants, les pronoms, etc.). Dans le cas de programmes ou de modèles, les identifiants, les signatures de méthodes, les noms de classes peuvent être utilisés comme mots-clés.

2.5.2 Recherche d'information et traçabilité

La traçabilité des exigences basée sur la recherche d'information a été proposée comme une méthode a postériori pour la détection de liens de traçabilité. Le principe derrière cette stratégie est de retarder autant que possible les efforts des ingénieurs dans la mise en œuvre manuelle de la traçabilité et de réduire les risques d'une sur-traçabilité dans des activités où celle-ci ne serait finalement pas requise. Ainsi, au lieu de réaliser manuellement les liens de traçabilité entre artéfacts, il suffirait seulement d'analyser les ensembles de *liens candidats* (c'est à dire, les documents remontés par l'algorithme de recherche d'information sur la requête) et de les valider ou les rejeter. Cette activité reste manuelle et ne peut être automatisée [DHH05].

Néanmoins, un facteur important à prendre en compte est l'effort dévolu à l'humain dans cette dernière étape de validation. Par conséquent, la question est aussi d'être capable de proposer une information de qualité à l'ingénieur. Lucia et al. [LFOT07],

Lormans et Deursen [LVD06] ou encore Mahmoud et Niu [MN11, NM12] ont analysé différentes stratégies d'analyse de la liste des liens candidats pour réduire le nombre de ces derniers. Ces différentes stratégies ont également été implémentées dans des outils comme Poirot [CHSDZ05], ADAMS [DLFOT05] ou RETRO [HDS+07].

Si l'on considère l'effort pour la traçabilité, ces techniques se comportent de manière honorable. L'explicitation manuelle des liens a un coût proche de zéro puisque les activités manuelles ne sont nécessaires que pour l'évaluation des candidats au moment de la requête, donc a posteriori. Cependant, ce résultat est à mettre en perspective vis-à-vis de la qualité des données à tracer a priori, mais aussi au coût lié à la connaissance du domaine, nécessaire pour la validation et l'exploitation de ces liens. De plus, il est difficile de pouvoir affirmer la complétude des liens candidats, car certains liens entre artéfacts peuvent exister alors que ceux-ci ne partagent aucune similarité d'un point de vue syntaxique, base sur laquelle se fonde la recherche d'information, même si les questions de synonymie ne semblent pas avoir un impact important dans les domaines techniques, où le vocabulaire est plus fortement restreint [HDS06].

Parmi les travaux utilisant les techniques de recherche d'information pour la traçabilité, on retrouve les travaux de Cleland et al. [MSCHc12, DGH+11, CHBC+07, CHSDZ05]. Parmi ces travaux, on retrouve notamment les question de traçabilité et de conformité vis-à-vis de la réglementation (HIPAA) [CHCGE10], les systèmes industriels dans le domaine de la mécatronique [CCHcB12] ou les systèmes importants pour la sûreté dans un cadre plus général [CHHH+12]. Dans ces cas cités, il s'agit d'une traçabilité entre les exigencesd'un côté et le code du logiciel développé ou des éléments de modélisation (use cases, test cases) de l'autre [MCH10]. Chez Leuser et Ott [LO10], la traçabilité basée sur la recherche d'information trouve une application dans l'automobile chez Daimler. Les artéfacts à tracer sont issues de document de spécification écrits en langue naturelle contrainte. Ces derniers utilisent donc particulièrement les restrictions sur les termes du domaine et la construction des exigences pour formaliser leur données en entrée. Chez Chen et Grundy, celle-ci est mise à profit pour la traçabilité entre le code et la documentation [CG11].

Hormis les questions de traçabilité entre éléments, on retrouve également, les approches qui visent à regrouper ces derniers au sein de groupes thématiques (algorithmes de clustering et de topics detection). Parmi ces techniques, Henßet al. utilisent des techniques d'apprentissage autour de l'algorithme LDA (Latent Dirichlet Allocation) [BNJ03] pour produire automatiquement des FAQs (Frequently Asked Questions) [HMM12]. Ascuncion et al. [AAT10] utilisent également l'algorithme LDA pour la détection et la modélisation de thèmes pour la découverte de liens de traçabilité. Ces algorithmes basés sur les techniques d'apprentissage sont aussi mis en œuvre pour la définition de systèmes de recommandations pour les liens entre fonctionnalités [DGH+11].

Malgré un compromis intéressant dans le rapport coût bénéfice, le problème commun à toutes les approches de traçabilité est la sémantique, à la fois de l'information traitée dans les documents que dans les ensembles de liens candidats. D'une manière général, il est nécessaire que chaque terme soit employé avec une sémantique unique, qu'il existe un thésaurus ou un glossaire permettant les liens entre synonymes. Heureusement dans ce genre de domaines, le vocabulaire est relativement limité et les termes sont employés

spécifiquement. La deuxième limitation vient de la sémantique du lien qui est remonté. La plupart des approches ne sont capables de remonter que les pairs ou un ensemble de documents qui partagent une similarité, mais ne permettent pas d'établir la sémantique de cette similarité, et qui reste à la charge de l'ingénieur dans son activité de validation. Néanmoins, elle se comporte honorablement lorsqu'il n'y a pas de sémantique dans les liens de traçabilité.

2.5.3 Fondamentaux de la recherche d'information

Dans cette section, nous proposons un bref panorama autour des aspects théoriques autour de la recherche d'information. Ce panorama a pour but d'apporter les fondamentaux sur les techniques utilisées au cours de la thèse pour la traçabilité et n'a pas pour vocation d'établir un état de l'art autour de ces approches.

2.5.3.1 Le document, un citoyen de première classe

L'unité de traitement dans une recherche d'information est le document. Par document, nous entendons un fragment de texte, peu importe sa longueur, son contenu ou sa structure. Un document peut ainsi avoir un verbatim extrêmement long comme le texte d'une norme dans son entier mais sans sa structure, ou se limiter à un paragraphe, une phrase ou à un ensemble de mots. Ce document peut avoir des attributs (ou des champs) comme des métadonnées l'accompagnant.

2.5.3.2 Indexation dans un corpus documentaire

Le point de départ de toute recherche d'information et de traitement d'un corpus de document consiste en un certain nombre de pré-traitements et à l'indexation des documents.

Traitement des documents du corpus. Le but de cette étape est de retirer des documents toute information inutile ou qui va nuire au traitement des données. Tout d'abord, chaque document est découpé en unités de sens (la plupart du temps, cette unité est constituée autour du mot).

Par la suite, on procède à l'élimination des mots vides (stop words), c'est à dire les mots trop courants pour avoir une valeur pour l'étude. Ces mots font généralement parti d'une liste personnalisable au besoin pour ajouter ou retirer des mots. Pour passer outre la conjugaison des verbes ou les préfixes/suffixes associés aux mots, les mots restants sont par la suite *lemmatisés* et *racinisés*.

La *lemmatisation* consiste à trouver un lemme en partant de ses *flexions*. Les flexions sont les différentes formes qu'un même mot peut prendre. Un lemme est la forme non conjuguée et non accordée d'un mot. La *racinisation* consiste à transformer un lemme en sa racine. Un racinisateur (ou stemmer) cherche la racine d'un mot en fonction de sa forme et de la langue souhaitée. Il existe plusieurs algorithmes de racinisation, celui que nous avons utilisé étant celui de Porter [VRRP80].

modélisation vectorielle du document Il existe trois grandes familles de modèles pour la représentation des documents : les modèles VSM (Vector Space Models) [SWY75], LSI (Latent Semantic Indexing) et PN (probabilistic Network). La plupart des travaux en recherche d'information utilisent l'une de ces trois représentations sans pour autant montrer de réelles différences entre elles. Nous nous concentrons sur les approches à base de VSM puisqu'il s'agit de la modélisation majoritairement utilisée dans les outils Poirot [CHSDZ05], ADAMS [DLFOT05] et RETRO [HDS+07] ou dans les librairies open source comme Apache Lucene [5] ou l'analyseur du langage naturel développé à Stanford [DMMM+06].

Une fois le traitement du document effectué, celui-ci est représenté sous la forme d'un vecteur [SWY75]. La dimension de ce vecteur est égale au nombre total de termes du corpus. Chaque composante du vecteur représente la fréquence du terme (nombre d'occurrences du terme dans le document) correspondant dans le document. Rapportée à l'ensemble des documents du corpus, on obtient une matrice dont les colonnes représentent les documents et les lignes les termes de l'index.

2.5.3.3 Score de similarité et pondération TF-IDF

Au lieu d'utiliser la seule fréquence des termes, Salton a introduit la notion de pondération sur les termes pour tenir compte de la rareté d'un terme dans le corpus de documents [SM86, SB88]. Ce poids tient compte de la fréquence (nombre d'occurrences) du terme dans le document mais aussi de sa fréquence dans l'ensemble des documents. Plus un document possède une fréquence élevée vis-à-vis d'un terme, plus il a de chance de répondre à une requête contenant ce terme. Cependant, si ce terme est présent couramment dans le corpus, moins il sera discriminant pour la recherche.

Définition 2.7 *On note TF (term frequency), le nombre d'occurrences (fréquence) d'un terme dans un document.*

$$tf(t_i, d) = \frac{freq(t_i, d)}{|d|}$$

où $freq(t_i, d)$ *est la fréquence du terme* t_i *dans le document* d
$|d|$ *le nombre de termes dans le document*
On note IDF (inverse document frequency) la rareté d'un terme.

$$idf_i = \log \frac{|D|}{|\{d_j : ti \in d_j\}|}$$

où $|D|$ *: nombre total de documents dans le corpus*
$|\{d_j : ti \in d_j\}|$ *: nombre de documents où le terme* t_i *apparaît*
Finalement, le poids du terme w *ou score TF-IDF est le produit de ces deux scores.*

$$w_{id} = tf(t_i, d) * idf_i$$

Le score de similarité entre le document d*, par rapport à une requête* q *étant finalement le cosinus de l'angle entre les deux vecteurs (vecteur du document et vecteur de la requête).*

5. http://www.apache.lucene.org

2.5.3.4 Algorithmes de clustering

Les algorithmes de regroupement (clustering) visent la division automatique d'un corpus de données en sous-ensembles ou clusters cohérents. Avec l'utilisation massive d'internet, ce domaine a connu un essort important avec des cas d'utilisation pour la recherche de textes, la fouille de données, la détection de thèmes,l'organisation des résultats de recherche. La grande majorité des exigences étant représentées sous format textuel, il est naturel que les approches de clustering aient été utilisées pour l'analyse d'exigences.

Les algorithmes effectuent des appariements ou des divisions basés sur la similarité (ou la différence) entre documents les plus proches (ou les plus éloignés) les uns des autres et constituent de manière itérative les clusters jusqu'a atteindre le nombre de clusters désiré. Cette similarité est évaluée en fonction de critère comme la nature des documents ou vis-à-vis de leur contenu. Dans le cas d'un algorithme comme Average-Link Hierarchical Clustering (AHC), chaque exigence ferait initialement parti d'un cluster spécifique, puis serait fusionné avec son plus proche cluster et ainsi de suite jusqu'à former le nombre de clusters désiré. Dans le cas de l'algorithme K-Mean (K-moyennes), on définit K *c*entroïdes (un pour chaque cluster) avec K documents représentatifs des K clusters. Le corpus est analysé relativement à leur similarité vis-à-vis de ces centroïdes et placés dans les clusters. On recalcule les centroïdes des clusters jusqu'à ce qu'aucun artéfact ne soit réassigné à un cluster. L'algorithme de bissection est une forme de K-moyennes avec K = 2 ou chaque cluster est coupé de manière itérative en deux nouveaux clusters jusqu'à atteindre la granularité voulue.

Le clustering d'exigences est cependant très différent des approches de clustering pour les documents généralement utilisés dans les travaux de ce domaine. Premièrement, le nombre de clusters à établir et donc la granularité du clustering n'est pas une donnée facilement quantifiable en ce qui concerne les exigences alors que ce nombre est une donnée d'entrée de ces approches. Alors que le clustering des documents a plus largement pour but de trier les documents pour affiner les recherches en catégorisant les documents, il s'agit, pour les exigences, de les regrouper par thématique, et non pas par nature. De plus, la plupart des algorithmes les plus utilisés dans la communauté (ceux que nous avons cité précédemment), font la supposition que les documents n'appartiennent qu'à un seul cluster alors que cette distinction n'est absolument pas systématique pour les exigences. Cependant, certains algorithmes de clustering, comme Lingo [OW04], proposent des clusters pouvant se recouvrir.

2.5.3.5 Distribution probabiliste de thèmes

Une autre préoccupation autour des ces approches concerne le regroupement de documents ou d'exigences autour de thèmes (Topic Modeling, Topic Detection) et peut être envisagée à travers les mécanismes d'apprentissage. Dans ces approches, les documents d'un regroupement sont similaires s'ils partagent la même distribution de termes. Cette approche consiste donc à identifier ces modèles de distributions de termes et classifier les documents relativement à leur similarité par rapport à ces distributions.

Ces approches probabilistes peuvent se porter sur un grand ensemble de données et ont montré de manière empiriques de bonnes performances et prennent de l'importance dans le monde de la recherche d'information. Plusieurs types de distribution existent dans la littérature, distinguant notamment les modèles où les documents peuvent appartenir à un ou plusieurs thèmes. Dans le cas qui nous intéresse, l'algorithme LDA (Latent Dirichlet Allocation)[BNJ03] propose de multiples recouvrements de thèmes pour les documents.

L'idée de base des approches du type LDA est que l'information peut être dérivée d'une matrice de co-occurence terme/document. Les documents peuvent être décrits comme un mélange de plusieurs thèmes ou sujets où les thèmes peuvent se décrire autour de la distribution de probabilité de certain termes ou de mots-clés dans les thèmes. Ainsi, ces modèles constitués en un certain nombre d'itérations sur un index de documents proposent non seulement la liste des thèmes et de leur distribution de mots-clés mais également la distribution de ces thèmes sur le corpus. Nous décrirons les principes de l'algorithme LDA dans la section 4.4.3, au moment où nous l'utiliserons.

2.5.3.6 Métriques pour l'évaluation des approches de recherche d'information

Deux mesures standards sont utilisées pour l'évaluation des approches de recherche d'information : le rappel et la précision. Le rappel mesure la proportion de liens corrects remontés parmi l'ensemble des liens corrects. La précision mesure la proportion de liens corrects parmi les liens remontés. Ils sont calculés de la façon suivante :

$$\text{rappel} = \frac{\text{nombre de liens corrects retrouvés}}{\text{nombre de liens corrects}}$$

$$\text{précision} = \frac{\text{nombre de liens corrects retrouvés}}{\text{nombre de liens retrouvés}}$$

En traçabilité des exigences, la précision est généralement largement dévaluée au profit d'un fort score de rappel (80 à 90%). Du point de vue de la traçabilité, il est préférable d'obtenir un certain nombre de liens faux positifs parmi le plus grand nombre possible de liens valides, quitte à augmenter l'effort de validation plutôt que de manquer des liens (faux négatifs) avec une tâche manuelle de recherche de liens supplémentaire.

Cette approche présente cependant un défaut majeur. Il est facile d'obtenir un score de rappel de 100%. Il suffit de retourner l'ensemble des liens possibles. Cette stratégie rend cependant l'approche inutile et perd tout bénéfice. Dès que les corpora possèdent des tailles conséquentes, le compromis rappel/précision devient donc un enjeu important afin de proposer des ensembles suffisamment petit et complets pour des analyses manuelles a postériori.

2.6 Discussion et synthèse autour des manques à combler

1. **Les exigences réglementaires ne sont pas des exigences comme les autres**. Elles sont de haut niveau, exprimées en langue naturelle non contrainte. D'un point

de vue global, il est difficile de les formaliser. Elles sont déjà formulées et, s'il est possible d'exprimer des propriétés sur ces exigences, d'identifier des ambigüités sur ces exigences ou de les interpréter, il n'est pas question d'en modifier le texte a priori mais plutôt de l'analyser et dériver une ou plusieurs autres exigences, du côté de l'ingénierie, pour répondre à cette exigence réglementaire. Ainsi la réexpression de ces exigences avec l'aide de patrons de spécification n'est pas appropriée.

2. Très peu de travaux observés dans la littérature se concentrent sur **les exigences réglementaires dans leur globalité**. Une très large majorité des travaux sur la conformité vis-à-vis des exigences réglementaires menés sur HIPAA le font sur des points très spécifiques de la loi (violation des données, protection des informations des patients, comparaison d'une même exigence dans différents états). Les travaux autour de la DO 178-B et de la CEI 61508 ne concernaient que ces deux normes avec la certification en perspective. Nos travaux à l'échelle de corpora entiers sont donc des précurseurs et ont été suivis plus récemment par des projets européens comme OPENCOSS [VPW13].
3. Les approches orientées buts comme KAOS, GRL ou i* sont particulièrement utiles en phase d'élucidation et pour préparer la spécification. De plus elles proposent des approches à multiples vues/perspectives qui sont particulièrement intéressantes, elles conviennent moins en terme **d'analyse ou de représentation du domaine**. Quand aux approches généralistes de modélisation à la UML ou SysML, elles ne sont pas utilisables directement et il est indispensable de les étendre par un mécanisme de profil pour prendre en compte la diversité des informations (nature des objets à manipuler, nature des associations) que nous souhaitons capturer dans le modèle ([LSM+10, WSB+10, GKvdB08, GPF12]. En particulier, le diagramme d'exigences de SysML se révèle particulièrement pauvre en terme d'expressivité.
4. **Les exigences ne sont pas l'élément de départ des analyses dans la littérature**, mais leur forme modélisée a priori et la transformation de l'exigence textuelle vers un élément de modèle est le plus souvent occultée [YBL11]. Le choix de l'utilisation de telle ou telle forme ou approche de modélisation n'est pas évoqué et est plus abordé comme un état de fait que comme un véritable paramètre de l'étude. Or, si on s'aperçoit que la plupart des exigences modélisées concernent des exigences "classiques", la plupart du temps fonctionnelles et donc représentées sous la forme de cas d'utilisation de scénario, la vérification de propriétés s'opèrent le plus souvent sur des diagrammes comportementaux mais avec des propriétés très précises qui ont été exprimées et qui souhaitent être validées.
5. Les outils actuels de gestion des exigences sont essentiellement conçus pour la gestion de version et la vérification de liens d'allocation des exigences à des systèmes ou à leur vérification via les cas de tests qui leur sont associés. Par construction, ces environnements visent donc des exigences fonctionnelles / non fonctionnelles, foncièrement tournées vers leur développement. Ils ne visent pas la représentation ni l'analyse des exigences dans leur domaine ni leur comparaison. Bien que des environnements comme Pure : :Variants [6] proposent des intégrations aux outils

6. http://www.pure-systems.com/

d'ALM pour modéliser les similarités et variabilités dans l'élaboration de ligne de produits, ces exigences portent sur des fonctionnalités particulières au sein d'une même organisation et non entre plusieurs référentiels différents.

6. La traçabilité est prise en compte comme un problème spécifique. Qu'il s'agisse de proposer un langage pour visualiser la traçabilité dans les modèles ou de techniques pour mettre en œuvre la traçabilité, les travaux académiques concernent souvent la traçabilité vers les artéfacts de développement. La traçabilité entre exigences est prise en compte via la formalisation des interactions notamment pour la décomposition des exigences, plus rarement pour mettre en évidence des références explicites dans la documentation [MAS11b, MAS+12b], des contraintes ou des interactions entre exigences [ZMZ05].
7. La traçabilité des exigences, basée sur la recherche d'information, favorise le rappel au détriment de la précision. La génération de faux positifs, si elle est préférable à la perte d'éléments (faux négatifs), tend à rendre difficile la validation des candidats à mesure que ces ensembles de candidats grossissent. Pour des ensembles de milliers de documents, l'approche se retrouve très limitée avec un score de rappel haut et un score de précision trop bas.

Deuxième partie

INCREMENT, une approche hybride pour la représentation et l'analyse des exigences réglementaires de sûreté

Introduction de la partie

Résumé de la partie I

Dans la partie précédente, nous avons présenté, dans le chapitre 1, une première contribution d'ordre méthodologique avec un état de la pratique industrielle autour du contrôle-commande nucléaire et ses évolutions depuis le milieu des années 1980. Ces évolutions ont amené ce domaine à l'utilisation de systèmes programmés qui posent un certain nombre de problèmes pour la qualification de sûreté et amené à la multiplication des règlementations pour encadrer la sûreté de fonctionnement de ces systèmes.

La première partie du chapitre 2 a été consacrée à l'état de l'art académique autour de l'ingénierie des exigences avec, dans un premier temps, un panorama dans le large de l'ingénierie des exigences. Nous avons ensuite mis l'accent sur deux aspects particuliers que sont la modélisation d'exigences, et les questions de conformité à la réglementation.

Nous avons présenté dans la seconde partie du chapitre 2, un état de l'art sur la traçabilité des exigences, avant de mettre l'accent sur les techniques de recherche d'information pour la traçabilité et de présenter une introduction à ces techniques.

Introduction de la partie II

La seconde partie de cette thèse se consacre à la contribution technique de la thèse, à savoir l'approche INCREMENT. Celle-ci s'articule autour des trois autres contributions de la thèse et en trois chapitres qui les adressent respectivement :

- les travaux liés aux problèmes de formalisation du domaine à travers la contribution INCREMENT-MDE que nous présentons dans le chapitre 3 ;
- les travaux liés aux questions de traçabilité des exigences et d'organisation du domaine dans des thèmes (chapitre 4) à l'aide de techniques de recherche d'information et la contritbution INCREMENT-IR ;
- la proposition, dans le chapitre 5, d'une approche hybride, INCREMENT-Hybrid, mélangeant les approches de modélisation et de recherche d'information pour une utilisation conjointe.

Chapitre 3

INCREMENT-MDE : une approche d'Ingénierie Dirigée par les Modèles

3.1 Introduction : un métamodèle pour les exigences réglementaires du contrôle-commande

Dans cette section, nous présentons la première partie de l'approche INCREMENT, à savoir l'aspect métamodélisation de la formalisation du domaine. Outre la définition des différents concepts manipulés dans le domaine, celui-ci permet de proposer une organisation de ceux-ci dans un ensemble cohérent.

Cette présentation est vue à travers l'évolution en quatre temps du métamodèle au long de la thèse (section 3.2). Même s'il y eu d'autres métamodèles, ces quatre étapes illustrent le processus de transformation entre la vision académique du problème et une vision partagée entre chercheurs et industriels dans le cadre du projet CONNEXION. Cette transformation, opérée sur le temps de la thèse, illustre également la difficulté de la recherche en collaboration avec l'industrie et la nécessité d'une approche incrémentale pour capturer précisément les besoins de modélisation des partenaires et mener une recherche pertinente.

Dans un second temps, nous abordons l'outillage qui accompagne le métamodèle et les travaux menés autour de celui-ci. Cet outillage porte sur l'acquisition systématique d'éléments de modèles instance du métamodèle ainsi que les facilités de navigation et de manipulation de modèle (section 3.3).

3.2 Histoire d'un métamodèle en milieu mixte académique et industriel

Dans cette section, nous abordons les quatre étapes de la vie d'un métamodèle, de ses prémices qui correspondent à la vision que nous avions au début des travaux de la thèse, à ses évolutions majeures dans le temps et au fil de la thèse avec EDF puis dans le cadre du projet CONNEXION qui ont nourri la réflexion autour du métamodèle.

3.2.1 Episode I : RAQM un triptyque <exigence – architecture – qualification>

A partir des problèmes soulevés dans le chapitre 1, trois concepts fondamentaux émergent rapidement et naturellement. Ce sont les concepts d'exigences, d'architecture de systèmes sur lesquelles portent ces exigences et enfin la qualification de ces systèmes vis-à-vis de ces exigences. Ce triptyque de concepts <exigence, architecture, qualification> peut être présenté et organisé dans un métamodèle que nous avons dénommé RAQM (pour Requirements - Architecture - Qualification Metamodel) 3.1.

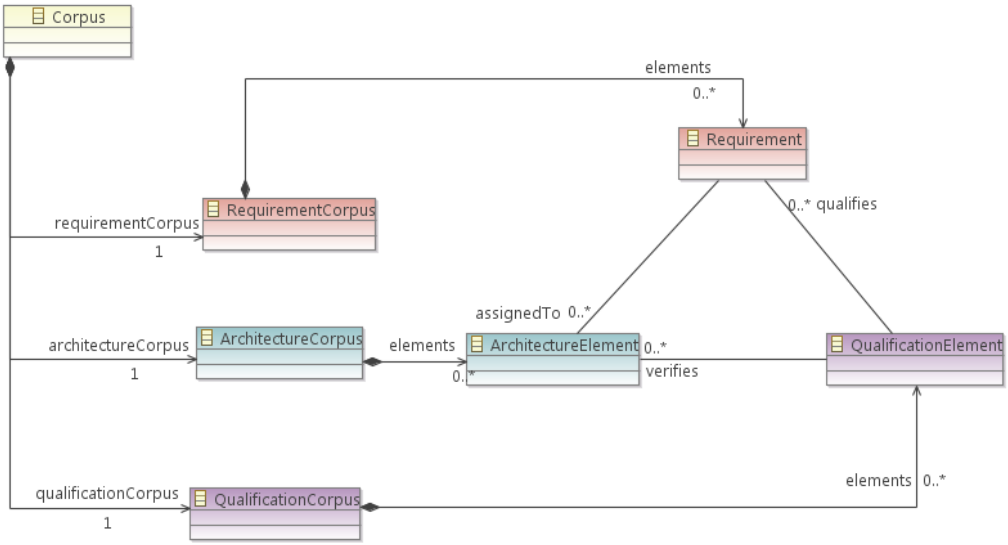


FIGURE 3.1 – Métamodèle RAQM

Ce petit langage de modélisation permet rapidement de créer des instances des différents éléments et, dans une vision idéalisée, d'avoir une organisation où chaque exigence du modèle peut ainsi être associé à des éléments d'architecture et de qualification. De la même manière architecture et qualification sont également reliés. Les cardinalités sont relâchées, ce qui permet d'obtenir des modèles conformes rapidement. Cela permet également d'analyser si certaines allocations sont instanciées ou non et ainsi de lever des anomalies quand les exigences ne sont pas associées à des éléments d'architecture ou à des éléments de qualification.

Cependant, cette vision est simpliste et très limitée, car elle ne tient pas compte de la diversité et de la nature des exigences, de leurs provenance ou même du fait qu'elles interagissent également entre elles.

3.2.2 Episode II : Réification d'exigences, variabilité et interactions

L'évolution suivante suit plusieurs dimensions. Tout d'abord l'organisation des exigences au sein de documents comme nous l'avons montré dans le chapitre 1, et la notion d'interprétation et de redéfinition des exigences. Ensuite, la définition de différents concepts pour les liens de traçabilité et qui se divisent en trois catégories d'interactions ou de dépendances. Cette augmentation du métamodèle est présentée dans la figure 3.3.

Cette transformation s'amorce avec l'approfondissement de nos échanges avec les experts chez EDF et qui permettent de préciser certains des concepts majeurs que nous allons définir dans le métamodèle.

L'organisation des exigences au sein de la structure documentaire prend le contrepied d'une organisation à plat des exigences au sein d'un document de spécification. Outre la notion d'exigence, l'analyse des documents d'exigence manipulés par les ingénieurs EDF (et comme nous l'avons présenté dans le chapitre 1) montre que la nature même des documents varie, et que le contenu de ces documents est important. Il doit donc être modélisé à partir de concepts particuliers du métamodèles (par exemple, des définitions, des recommandations, du texte descriptif, la valeur normative ou informative d'une annexe, etc.) et ne doit pas se limiter aux seules exigences.

Dans les documents, l'identification explicite d'une exigence par un identifiant n'est pas systématique. Elle n'est pas non plus garante que la clause identifiée n'exprime qu'une seule exigence. Ainsi, les Safety Assesment Principles (textes réglementaires britanniques) ont une identification linéaire de chaque paragraphe / clause mais ces paragraphes peuvent aussi bien contenir une que plusieurs ou aucune exigence. De la même façon, les relations que nous avions initialement définies ne suffisent pas à exprimer les liens entre exigences. La sémantique de ces liens peut varier entre les liens explicites exprimant une seule référence ou l'obligation de suivre de nouvelles exigences.

La figure 3.2 présente un extrait de la norme CEI 60880 sur les aspects logiciels pour les systèmes de contrôle-commande réalisant des fonctions de catégorie A. Dans le cas présent, les clauses sont énumérées (6.2.A à 6.2.F). Cependant ces clauses n'expriment pas toutes des exigences. Les clauses 6.2.C et .2.F expriment des recommandations. 6.2.C exprime une énumération d'erreurs à détecter. Ces erreurs peuvent être considérées de manière individuelle, formant trois recommandations, ou collective et définir une seule recommandation. De la même manière, la clause 6.2.D est accompagnée d'une note additionnelle, complétant l'information de l'exigence mais sans en faire partie. L'exigence 6.2.A fait également référence à une référence interne (l'annexe A.2.2) et l'exigence 6.2.D à la norme CEI 61513. Cette dernière relation entre ces deux normes est cependant plus forte puisqu'elle impose que l'implémentation des réponses suite à une erreur suivent les règles de conception énoncées dans la norme CEI 61513.

Bien que nous ayons montré que les exigences réglementaires manipulées étaient ambigües, difficilement vérifiables, peu claires, ni systématiquement identifiés, les documents qui les contiennent sont cependant bien formés et organisés. Il y a donc un intérêt particulier à reprendre la structure documentaire dans le modèle. La structure permet de tenir compte du périmètre des exigences et de naviguer à différents niveaux de granularité dans les textes (à l'échelle du document, de la section ou du fragment

6.2 Self-supervision
6.2.A The software of the computer-based system shall supervise the hardware during operation within specified time intervals and the software behaviour (A.2.2).
This is considered to be a primary factor in achieving high overall system reliability.
6.2.B Those parts of the memory that contain code or invariable data shall be monitored to detect unintended changes.
6.2.C The self-supervision should be able to detect to the extent practicable :

– Random failure of hardware components ;
– Erroneous behavior of software (e.g. deviations from specified software processing and operating conditions or data corruption) ;
– Erroneous data transmission between different processing units.

6.2.D If a failure is detected by the software during plant operation, the software shall take appropriate and timely response. Those shall be implemented according to the system reactions required by the specification and to IEC 61513 system design rules.
This may require giving due consideration to avoiding spurious actuation.
6.2.E Self-supervision shall not adversely affect the intended system functions.
6.2.F It should be possible to automatically collect all useful diagnostic information arising from software self-supervision.

FIGURE 3.2 – Extrait de la norme CEI 60880 pour l'identification de concepts et de relations

unitaire (*TypedFragment*) dans le référentiel d'exigences.

En ce qui concerne les liens de traçabilité rencontrés, ceux-ci vont largement au delà des seules relations d'allocation ou de vérification entre éléments d'exigences, d'architecture et de qualification, car ces éléments ont également une organisation au sein de leur propre domaine. C'est le cas pour les exigences et les documents qui se référencent mutuellement, s'impliquent, se requièrent. Nous avons appelé ces liens de traçabilité des *interactions*. Les interactions classiques qui existaient à l'état de références entre éléments du triptyque <exigence, architecture et qualification> sont cette fois-ci perçues comme des liens de traçabilité explicites entre éléments de domaines différents (*InterAreaInteraction*). Nous avons également introduit des interactions dans le domaine des exigences (*DomainInteraction*) qui couvrent quant à elles les liens de références, d'implication, de couverture. Ces liens indiquent une dépendance d'un élément d'exigence vers un autre Second type de *DomainInteraction*, nous avons introduit des liens pour la comparaison entre éléments d'exigences afin de répondre à l'un des objectifs de la thèse pour la représentation de similarités/différences entre exigences. Ces relations comprennent les notions d'équivalence totale ou partielle et celle de conflit. Si certains liens, comme les références, peuvent être retrouvés de manière automatique, la comparaison de deux exigences relève de l'expertise humaine. Nous ne fournissons ici que le mécanisme pour exprimer le lien.

Enfin, nous avons abordé le volet interprétation des textes sous la forme d'interprétations et de raffinements. Ces interprétations permettent de prendre une exigence

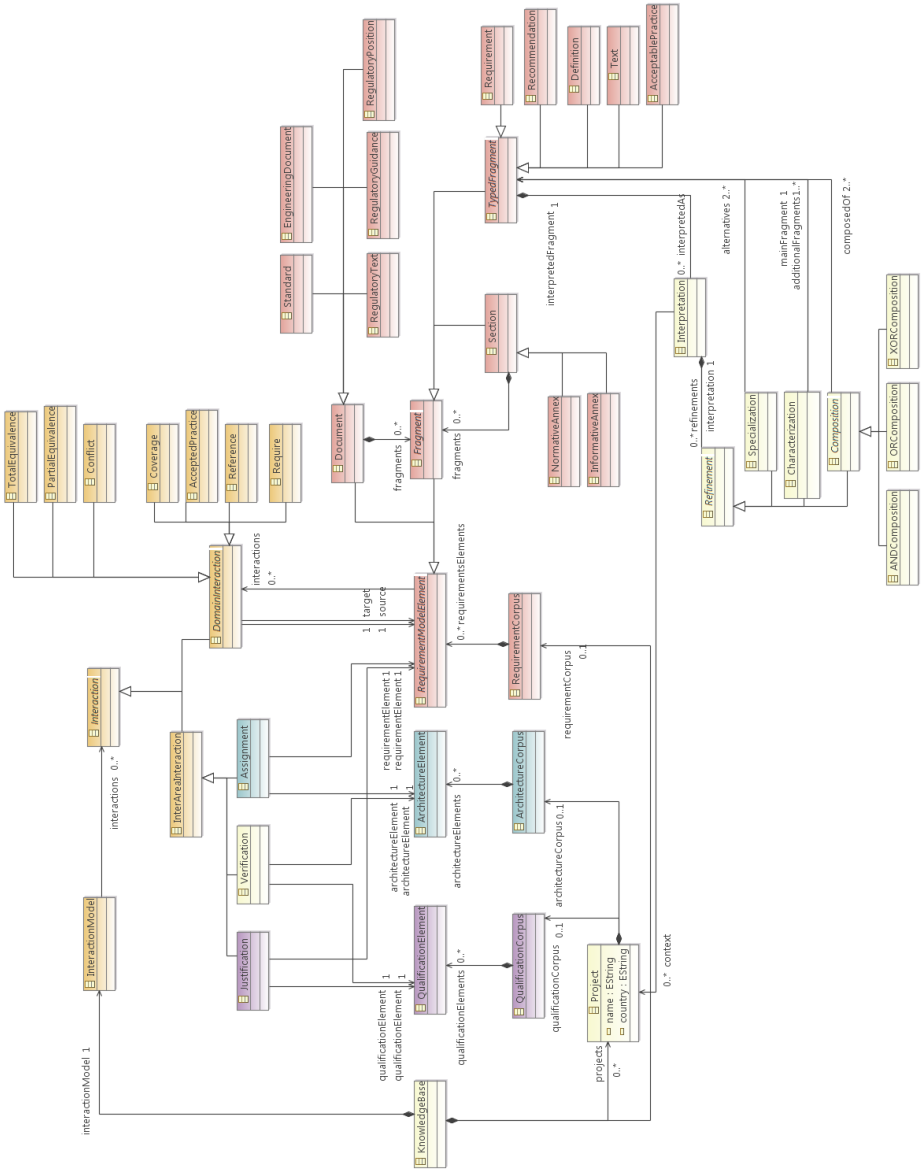


FIGURE 3.3 – Métamodèle KVT

d'un document et de la raffiner en une ou plusieurs autres avec des raffinements spécialisés comme les caractérisations (séparation entre exigence principale et exigences secondaires), spécialisations (propositions de plusieurs alternatives pour satisfaire une même exigence, ou des mécanismes de composition plus classiques. Ce mécanisme de composition va au delà de la simple notion de *containment* qu'on retrouve dans SysML car cette dernière ne reprend que le mécanisme de composition d'UML (avec la notion d'appartenance à un parent et de propagation de la suppression du parent à ses enfants). Nous avons augmenté nos *Refinements* en proposant trois possibilités (caractérisation, spécialisation et décomposition) et une organisation particulière des éléments. Cette décomposition reprend la proposition de métamodèle de modèle de features de Perrouin et al. [PKGJ08] pour l'expression de la variabilité au niveau de ces exigences. Ces mécanismes peuvent être intéressant dans le cas de clauses contenant plusieurs phrases ou plusieurs exigences et que l'on souhaite les analyser séparément.

3.2.3 Episode III : « Theme » ou la nécessité de regrouper des éléments

Dans la littérature, les exigences sont essentiellement considérées d'un point de vue atomique ou dans des perspectives de raffinements dans les approches orientées buts. Avec les nombreuses références croisées entre les normes ou les guides réglementaires, la vision globale des exigences concernant une préoccupation particulière (par exemple, la modification du logiciel, la vérification et validation, la communication entre lignes de défense) nécessite de parcourir un nombre important de documents différents. A la différence d'une approche orientée but qui propose un raffinement en profondeur vers les exigences, nous nous retrouvons avec une description à la fois en profondeur mais aussi en largeur, transverses à plusieurs documents.

Cet aspect est né à la fois de nos échanges avec EDF et des échanges dans le cadre du projet CONNEXION. En effet, l'allocation des exigences dans des thèmes (ou Topic) est apparue immédiatement comme une pratique des ingénieurs et a fait l'objet d'une des toutes premières tâches du projet.

Il est donc nécessaire d'avoir une vue différente des exigences, qui soit décorrélée de la seule vue hiérarchique d'un ou plusieurs documents. Le principal apport du métamodèle *"knowledge"* vient donc de la définition de la notion de topic, et l'approche *"theme"* que nous avons présenté dans [SB12a] et que nous présenterons dans le chapitre 4.

"Theme" est proche, dans l'idée du regroupement sémantique de données, de la proposition *Theme/Doc* de Baniassad et Clarke [BC04] même si celle-ci est orientée séparation des préoccupations et développement des exigences sous la forme d'aspects, et non leur seul expression. Gotel et Morris [GM11] redéfinissent la traçabilité à travers les notions d'individus à tracer, de signes distinctifs (ou de signature) permettant d'identifier un individus et de traces, c'est à dire l'ensemble des signatures retrouvées dans l'environnement.

Définition 3.1 *Nous définissons ainsi la notion de* ***Topic*** *ou de* ***thème*** *comme la préoccupation à tracer dans le référentiel, la notion de signature étant la liste des mots-clés associés au topic, et les traces, comme étant les références aux éléments du modèle*

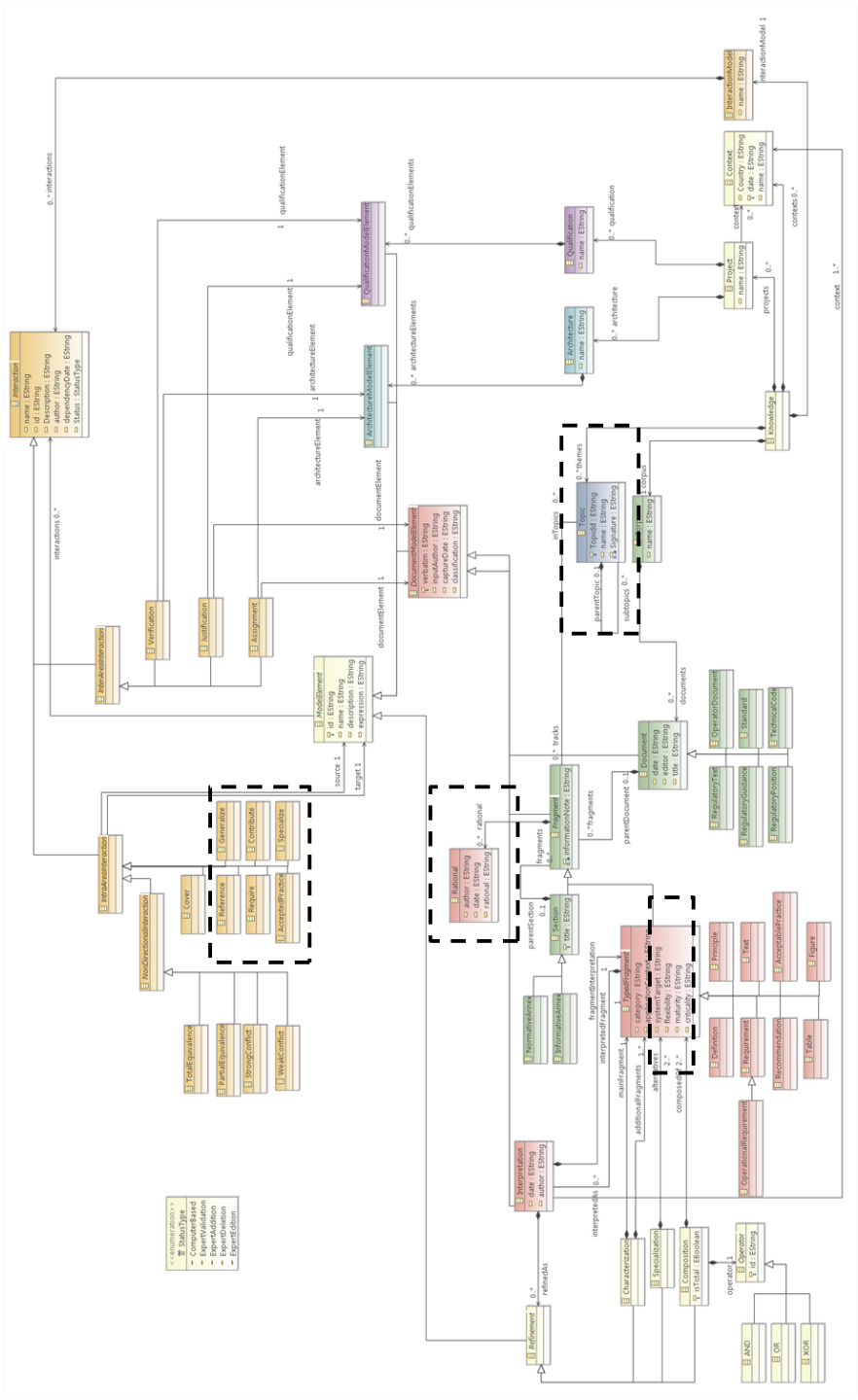


FIGURE 3.4 – Métamodèle Knowledge, les zones encadrées représentent de nouveaux attributs pour les "exigences", ou de nouveaux concepts comme le concept de thème (Topic) ou de nouvelles interactions

où l'on retrouve le topic. A travers les topics, nous proposons une vue dans le large des différents éléments de modèles relatifs à la préoccupation et regroupés dans le même ensemble.

3.2.4 Episode IV : Connexion, la dissociation entre exigences et textes dans les documents

Si nous avions défini des mécanismes d'interprétation et de variabilité, ceux-ci se sont avérés trop complexes, en tout cas trop peu intuitifs, et peu utilisés dans les modèles que nous avions instanciés pour la validation du métamodèle. Le métamodèle *Knowledge* aura été stable (en tout cas, sans modification significative) pendant un peu plus d'une année. La réflexion des partenaires du projet CONNEXION s'est portée sur une approche différente de l'organisation des exigences. Si les exigences sont bien écrites dans les différents documents, les exigences manipulées gardent ce contexte de document, d'un point de vue structurel, mais sont désormais vues de manière atomique. Nos expérimentations passées se limitaient à prendre le paragraphe comme unité la plus petite pour le traitement des exigences. De cette façon, une "exigence" dans le texte pouvait en fait en contenir plusieurs, mais avait le bénéfice de pouvoir acquérir les exigences de manière systématique.

En lieu et place des interprétations, jugées trop complexes, les partenaires au sein du projet CONNEXION ont préféré une séparation des exigences des textes dont elles sont issues. Dans le métamodèle *Knowledge*, les *TypedFragment* (c'est-à-dire les fragments atomiques des documents) étaient directement contenus dans les documents. Ce n'est plus le cas dans le métamodèle *Connexion* ou les *TypedElement* sont contenus "à plat" dans un corpus tiers (*TypedElementCorpus*) et peuvent être associés à des éléments de documents.

Les documents possèdent toujours leur structure composite, mais ceux-ci ne contiennent plus que des fragments de texte. Cette séparation permet de dissocier le verbatim des textes des documents, de celui des exigences, qui peuvent être identiques mais aussi être exprimées de manière différente tout en conservant un lien de traçabilité avec le fragment textuel d'origine.

De la même manière, dans les échanges, nous avons identifié le besoin d'ajouter le concept d'exigence non écrite. Ce sont des exigences ne provenant pas de documents, mais exprimées de manière formelle ou informelle au cours des projets. Il est donc nécessaire de distinguer la provenance des exigences. La séparation des exigences de la documentation concourt à la facilitation de cette différence entre exigences écrites et non écrites.

Si les volets qualification et architecture ont conservé une description sommaire, car secondaire aux travaux de la thèse, le contexte du projet CONNEXION est légèrement différent. Les exigences que nous manipulons sont de haut niveau et peu formalisées. Il est clair qu'une relation directe entre les exigences et les deux autres domaines de départ du métamodèle se révèle très difficile. Dans la pratique, pour la conception à haut niveau des architectures, les industriels se reposent sur des règles de conception (*DesignRule*) qui font office de proxy avec les exigences de plus haut niveau auxquelles elles répondent.

Dans le cadre du projet CONNEXION, les volets exigence et architecture de haut niveau font l'objet de deux métamodèles séparés et dont le rapprochement s'effectue par le biais de ces règles de conception.

De la même façon, les éléments de qualification et les exigences ne peuvent être directement liés et font également l'objet d'un proxy avec ce que nous avons défini comme Justification. A haut niveau, les deux concepts sont aussi génériques que les éléments d'architecture et de qualification qu'ils remplacent mais explicitent la nécessité de passer par ces éléments intermédiaires. Pour la justification, elle s'imprime dans la démarche de l'OMG et de son Safety Assurance Evidence Metamodel (SAEM) [OMG10, OMG13] encore à l'état de draft aujourd'hui pour la justification du logiciel à base de "preuves" (evidence en anglais), c'est à dire d'éléments venant démontrer le respect vis-à-vis d'une ou de plusieurs exigences. Dans notre cas, les "éléments de démonstration" peuvent varier de la simulation, de l'analyse probabiliste de sûreté, au simple argumentaire textuel et sont laissés comme un champ textuel.

Ces deux notions, *DesignRule* et *Justification*, sont encore à l'état de maturation au sein du projet CONNEXION et seront à développer par la suite.

3.2.5 Un exemple d'instanciation

Un exemple pratique d'instanciation de ce modèle est présenté dans les figures 3.7 et 3.8. A partir du texte réglementaire américain 10 CFR 50 qui suit (figure 3.6), nous présentons quelques éléments de transformation tout d'abord en fragment de textes pour la volet document (figure 3.7), et la traçabilité vers les exigences qui sont créées et qui seront les éléments manipulés (3.8). Si les deux verbatims sont pour l'instant identiques, ils peuvent être amenés à diverger. De plus, l'exigence créée possède un certain nombre de propriétés supplémentaires qui ont été acquises soit manuellement, soit automatiquement par analyse du texte.

3.2.6 Cycle de vie d'un métamodèle pour la représentation d'un domaine

Dans cette section, nous avons abordé la construction et l'évolution du métamodèle support à l'approche INCREMENT. Les évolutions que nous avons décrites sont dues, dans un premier temps à la meilleure compréhension du domaine. Dans un second temps, elle sont liées à la prise en compte de la volumétrie des exigences et de la nécessaire prise en compte de regroupements sémantiques autour de *topics*. Cette évolution marque également le début de la participation active des acteurs industriels dans le projet CONNEXION. Stable pendant une année, l'avant dernier métamodèle a évolué vers une séparation entre les exigences et les documents dont elles sont issues. Ce dernier métamodèle reprend plus largement la notion de regroupement et étend encore plus la notion d'exigences, pour faire ressortir la dimension écrite de la dimension tacite, la nature normative ou réglementaire des exigences, etc. Aujourd'hui les réflexions se portent également autour d'une meilleure définition de la notion de pratiques, que nous ne présentons pas ici.

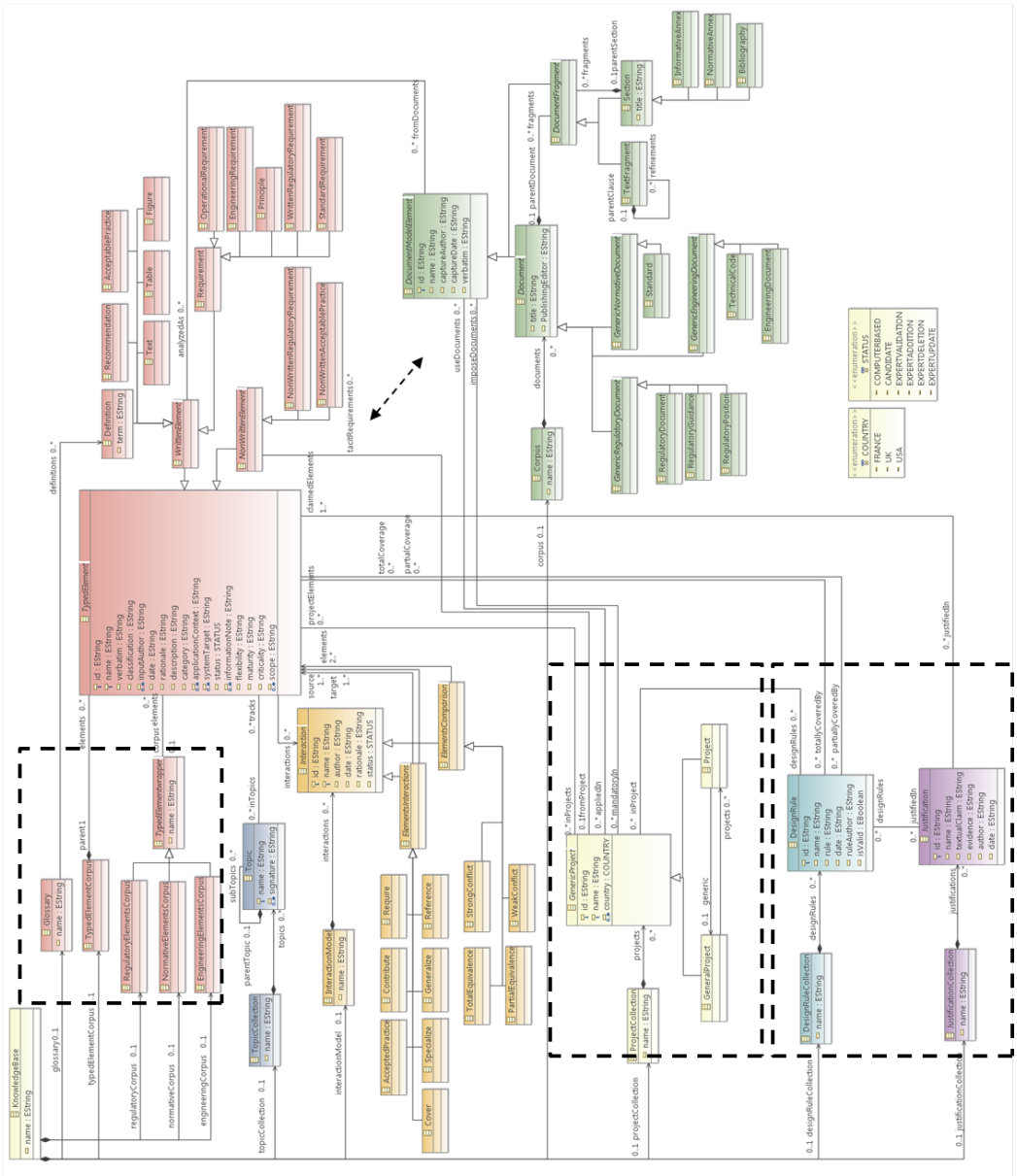


FIGURE 3.5 – Métamodèle Connexion Les zones encadrées présentent les nouveaux concepts de regroupements thématiques, le changement de définition des éléments d'architecture et de justification

III. Protection and Reactivity Control Systems
Criterion 20—Protection system functions. The protection system shall be designed (1) to initiate automatically the operation of appropriate systems including the reactivity control systems, to assure that specified acceptable fuel design limits are not exceeded as a result of anticipated operational occurrences and (2) to sense accident conditions and to initiate the operation of systems and components important to safety.
Criterion 21—Protection system reliability and testability. The protection system shall be designed for high functional reliability and in-service testability commensurate with the safety functions to be performed. Redundancy and independence designed into the protection system shall be sufficient to assure that (1) no single failure results in loss of the protection function and (2) removal from service of any component or channel does not result in loss of the required minimum redundancy unless the acceptable reliability of operation of the protection system can be otherwise demonstrated. The protection system shall be designed to permit periodic testing of its functioning when the reactor is in operation, including a capability to test channels independently to determine failures and losses of redundancy that may have occurred.
Criterion 22—Protection system independence. The protection system shall be designed to assure that the effects of natural phenomena, and of normal operating, maintenance, testing, and postulated accident conditions on redundant channels do not result in loss of the protection function, or shall be demonstrated to be acceptable on some other defined basis. Design techniques, such as functional diversity or diversity in component design and principles of operation, shall be used to the extent practical to prevent loss of the protection function.
Criterion 23—Protection system failure modes. The protection system shall be designed to fail into a safe state or into a state demonstrated to be acceptable on some other defined basis if conditions such as disconnection of the system, loss of energy (e.g., electric power, instrument air), or postulated adverse environments (e.g., extreme heat or cold, fire, pressure, steam, water, and radiation) are experienced.

FIGURE 3.6 – Extrait du texte réglementaire américain 10CFR0

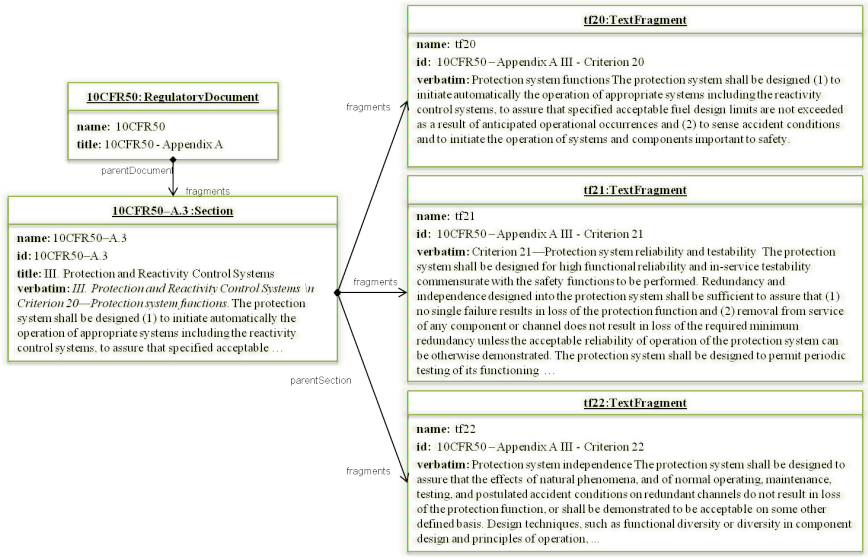


FIGURE 3.7 – conversion du texte à des éléments de modèles (première partie : conversion en TextFragment)

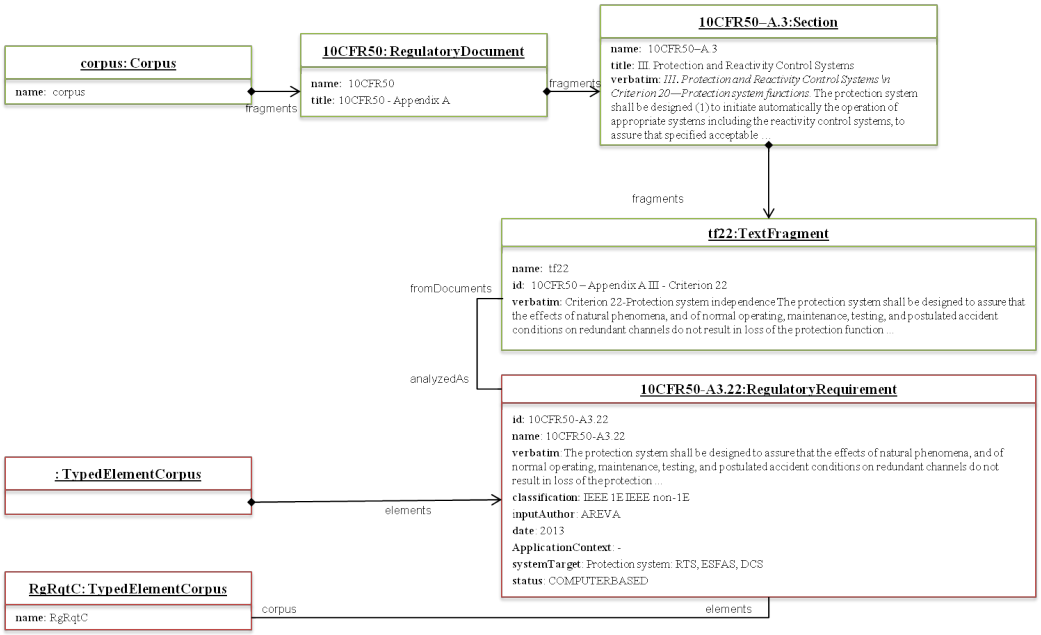


FIGURE 3.8 – conversion du texte à des éléments de modèles (première partie : création d'exigences et traçabilité vers le texte)

Il y a un nécessaire temps d'acclimatation et d'appropriation du métamodèle en tant que tel (comprendre les concepts de base de la métamodélisation et comprendre ensuite la définition des concepts du domaine et leur manipulation). Ce temps est nécessaire avec des partenaires qui ont une expérience en ingénierie système, qui peuvent connaître ou non UML, mais n'ont aucune expérience en ingénierie dirigée par les modèles. Se sont écoulés pratiquement 9 mois entre la première présentation du métamodèle aux partenaires CONNEXION (mai 2012), présentation détaillée (plusieurs discussions courant octobre 2012), prise en main par les partenaires via une base de données, modification, instanciation sur exemples concrets et note de description (à partir de mars 2013).

Il y a donc deux fils d'évolution qui suivent la logique de l'élucidation des exigences telle que décrite par Lamsweerde dans les itérations d'un processus d'ingénierie des exigences [vL09], où les parties prenantes académiques et industrielles se découvrent, découvrent voire redécouvrent le problème sous des angles différents. La question des règles de design, ou la séparation exigence-documents sont des illustrations de ce phénomène.

Entre la présentation du métamodèle et son appropriation par les industriels, il est important de noter que ceux-ci ont mené une activité parallèle pour déterminer les documents à analyser, les attributs importants des exigences, les liens à envisager. Il s'agit également d'une activité de métamodélisation bien qu'elle s'est traduite par la conception d'une base de données. Cette base de données couvre un périmètre plus petit que notre métamodèle mais sa conception s'est effectuée dans un environnement "maîtrisé", en tout cas plus familier pour les partenaires. Les résultats de cette activité se traduisent dans le métamodèle que nous avons construit au niveau de certains attributs des TypedElements et de liens de traçabilité supplémentaires.

3.3 Instancier un modèle Connexion

3.3.1 IncrementParser : un analyseur configurable pour instancier automatiquement un modèle Connexion

Afin de ne pas créer manuellement des instances complètes des documents, nous avons développé un analyseur syntaxique, IncrementParser, permettant de typer dynamiquement les entrées textuelles pour en extraire les éléments typés mais également la structuration des documents. De tels analyseurs existent pour la traduction d'exigences textuelles en éléments de modèles. Celui de Rauf et al [RAC11] transforme par exemple des spécifications en cas d'utilisation. Il en est de même pour les outils commerciaux qui utilisent les styles des documents pour effectuer l'extraction. Rauf et al, se basent sur des documents de spécifications bien formés, en langage naturel non contraint. La difficulté n'est pas au niveau de l'identification des exigences en elles même mais dans la conversion des différents éléments textuels vers les concepts du diagramme de cas d'utilisation.

Dans notre cas, il y a hétérogénéité dans la structuration des documents et le niveau d'abstraction choisi ne peut pas s'exprimer comme des exigences fonctionnelles. Pour

palier à cette hétérogénéité, notre analyseur est donc configurable et repose essentiellement sur l'utilisation d'expressions régulières pour la reconnaissance des concepts.

Cet analyseur a été développé pour générer des modèles conformes au métamodèle KVT et Knowledge et a été adapté pour le passage à des instances conformes au métamodèle Connexion. Celui-ci a permit l'acquisition de normes CEI, d'une recommandation de l'AIEA ainsi que de documents normatifs issus de l'IEEE.

L'analyseur ne propose qu'une fonctionnalité de typage et ne permet pas de déterminer les liens de traçabilité explicites que peuvent contenir les documents qui constitue une étape d'analyse supplémentaire sur le modèle. L'élément le plus fin à analyser est constitué du paragraphe. L'analyse à grain plus fin tel que la phrase ou un ensemble déterminé de phrase reste manuelle.

L'ensemble des documents présente des écarts dans leur organisation : différents types d'énumération pour la structure des documents, énumération ou absence d'énumération des clauses, énumération selon un style identique / différent de la hiérarchie du document.

La validation des modèles extraits des documents a été réalisée par vérification d'échantillons sur les documents.

3.3.1.1 Préparation initiale des documents

Les documents que nous analysons varient assez largement de l'un à l'autre dans leur format d'entrée. Même si la plupart d'entre eux sont accessibles au format pdf, l'encodage des fichiers varient sur les métadonnées, la gestion des index, le format des images et des tableaux, etc. Afin d'avoir une approche homogène dans le traitement de l'information, un certain nombre de tâches de pré-traitement est requis.

1. Il est nécessaire de convertir les documents d'entrée vers des fichiers txt. Si ce procédé a l'inconvénient de perdre les styles appliqués aux documents, il est à noter que les styles ne sont pas appliqués / analysables de manière homogène, indifféremment du format initial des documents, word ou pdf. Ils ne sont pas non plus systématiquement accessibles. Pour les documents établis à base de technologies à partir de photocopies ou de numérisation de documents papier, l'accès au style est impossible. La réalisation de ce traitement est réalisé via l'outil Tika, projet du consortium Apache et qui permet d'obtenir un texte, formaté proprement, avec la contrainte que chaque paragraphe est représenté sur la même ligne. Pour les listes, les éléments sont sur des lignes séparées.
2. A partir du texte au format .txt, il est nécessaire de retirer toutes les marques qui proviennent des fournisseurs et distributeurs du document, telles que les marques de copyright de l'IEEE, les marques de distributeurs comme SAGAWEB qui fourni l'accès à de nombreux documents normatifs, etc, les paginations, titrages, etc.
3. A partir du texte au format .txt, il est important de remettre en cohérence les figures et tableaux qui ont également été importés et qui rendent plus difficile une analyse systématique de chaque ligne du fichier. Chaque figure et tableau devant être analysé de manière ad hoc, nous proposons d'étiqueter (#figure# ou

#table#) ces éléments afin de conserver leurs informations textuels, et de les typer comme texte informatif ou tableau.

4. Certains documents tels que la recommandation NS-G-1.3 de l'AIEA ou les SAPs du HSE définissent, sans les numéroter, de grandes sections, tandis que les clauses sont numérotées sans pour autant présenter de hiérarchie particulière. Il peut être nécessaire de reconstruite « artificiellement » cette hiérarchie pour pouvoir proposer différents niveaux de granularité et prendre en compte ces titres dans les documents.

3.3.1.2 Configuration de l'analyseur

La configuration de l'analyseur pour chaque document se réalise au moyen d'un fichier tiers qui produit l'ensemble des expressions régulières nécessaires au typage des différents fragments textuels rencontrés. Le fichier de configuration contient également un jeu de métadonnées telles que le titre, l'année, qui permet d'apporter un ensemble d'informations complémentaires par le biais de ce seul fichier de configuration. Nous présentons dans la figure 3.9 un exemple de fichier qui a servi pour l'acquisition de la norme CEI 60880, publiée en 2006.

```
title = IEC60880 - Nuclear power plants - Instrumentation and control ...
tag = IEC60880
date = 2006
type = standard
section = ([0-9]+)(.[0-9]+)*(?!\))(?!.*(shall|should|may| )).*
sectionSeparator = .
informativeAnnex = ^Annex.*\(informative\).*
normativeAnnex = ^Annex.*\(normative\).*
annexSection = [A-Z](.[0-9]+)+(?!.*(shall|should|may| )).*
definition = 3.[0-9]+.* .*
acronym = [A-Z]+.* .*
requirement = (([0-9]|[A-Z])+(.[0-9]+)*|[a-z]\))(?=.*shall).*
recommendation = ([0-9]|[A-Z])+(.[0-9]+)*(?!.*shall.*)(?=.*should).*
text = !([A-Z]((.[0-9])+|[0-9](.[a-z])*))(?!.*(shall|should|may| )).*
textNote = (?= (NOTE|Note|note).*)
figure = (?=// figure //).*
table = (?=// table //).*
listMarker = .*:( )?
list = -.*
enumList = ([0-9]+\)|[a-z]+\)).*
```

FIGURE 3.9 – Fichier de configuration pour l'acquisition de la norme CEI 60880

Ce fichier permet de distinguer à travers les expressions régulières des éléments à grain fin qui peuvent partager la même organisation mais qui sont différents (telles que

la différence entre exigence et recommandation) ou des éléments de structure entre eux (sections de chapitres, sections d'annexes) et qui n'ont pas forcément la même valeur, normative ou informative. Une expression du fichier de configuration s'écrit et s'analyse de la manière suivante <item>␣=␣<expression> où ␣représente le caractère d'espacement (un seul et unique espace). L'item est soit une métadonnée ("date", "tag", "listMarker") soit une métaclasse du métamodèle ("definition", "section", "requirement"). Expression est soit la propriété dans le cas d'une métadonnée, soit une expression régulière pour la reconnaissance de type.

3.3.1.3 Traitement et génération d'éléments de modèle

La génération d'éléments se déroule en deux étapes :

1. Regroupement de fragments dans une table. Il s'agit ici de regrouper, dans une seule entrée d'une table, les éléments susceptibles d'être dispersés sur plusieurs lignes. Cela concerne les listes, dont les énumérations sont sur plusieurs lignes, ainsi que les figures et tableaux. Si l'on reprend l'exemple de l'exigence 6.2.C de la norme CEI 60880 (figure 3.10), cette étape consiste en la fusion de la ligne contenant la clause 6.2.C et des trois lignes concernant chaque élément de la liste.

6.2 Self-supervision 6.2.C The self-supervision should be able to detect to the extent practicable : – Random failure of hardware components ; – Erroneous behavior of software (e.g. deviations from specified software processing and operating conditions or data corruption) ; – Erroneous data transmission between different processing units.

FIGURE 3.10 – Extrait de la norme CEI 60880

2. Typage et génération d'éléments de modèle. Chaque ligne de la table est ensuite analyser pour déterminer s'il s'agit d'un élément de structure (section) ou d'un élément typé. En utilisant les règles de style, il est possible d'établir la hiérarchie et la structure du document et ainsi d'organiser les différents fragments au sein du document. Chaque fragment est aussi typé comme un élément et est créé en fonction de son parent.

3.3.2 Instancier automatiquement un modèle avec des documents du monde nucléaire

L'utilisation de cet analyseur a permis l'acquisition de huit documents normatifs issus de la CEI (CEI 60880, 60987, 61500, 61513, 61226, 62138, 62340, 62566), de la norme de sûreté NS-R1.2 issue de l'AIEA ainsi que des documents issus de l'IEEE. Dans la pratique, l'analyseur acquiert dynamiquement un modèle Connexion contenant une norme, il ne vient pas enrichir un modèle Connexion existant et contenant des documents déjà acquis. La validation des modèles s'effectue par échantillonnage, l'acquisition

nécessite par conséquent quelques itérations et ne peut être directement ajoutée au modèle. L'évolution des métamodèles a nécessité la modification du parser dans le sens où la modification des concepts définis dans le métamodèle change également la création et l'organisation des éléments de modèles à créer et organiser.

Les métriques menées sur l'étude des 8 normes CEI, ont montré la génération de plus de 4000 éléments de modèles pour le métamodèle Knowledge et d'un peu moins du double pour le métamodèle Connexion du fait de la séparation entre fragments textuels et éléments d'exigences. Parmi ces 4000 éléments, nous avons identifiés 1339 exigences (5.2. Ces chiffres sont à mettre en comparaison avec les efforts manuels et hétérogènes des partenaires du projet CONNEXION pour acquérir manuellement un peu moins de 800 exigences mais dont les critères et les sources d'acquisition divergent des nôtres.

3.3.3 Instancier manuellement et manipuler un modèle avec un environnement graphique

Acquérir de manière automatique un modèle Connexion avec l'analyseur est une possibilité, mais il est aussi nécessaire de fournir un environnement pour la manipulation et l'édition de modèles. Une partie des informations du modèle ne peut pas être acquis automatiquement, ou de manière systématique avec l'assurance d'une totale correction. Comme nous l'avons mentionné, les verbatim textuels et les textes des exigences tels qu'ils seront exploités peuvent évoluer, des liens de similarité être établis, etc. Il est donc nécessaire de fournir de primitives pour la manipulation et l'édition des éléments de modèle.

Un premier prototype d'environnement utilisant Obeo Designer [1] propose une vue diagrammatique des modèles et se base sur le métamodèle KVT. Le principal avantage d'une vue en diagramme est d'expliciter les liens de traçabilité entre les éléments de modèles. La principale limitation est sa difficulté à passer à l'échelle sur des modèles de taille moyenne voire importante sans mécanisme de filtre [BCBB11]. Les modèles que nous construisons possèdent pour une norme plusieurs centaines d'éléments. De plus les hiérarchies se représentent mal dans une représentation de diagrammes de classes qui présente les choses "à plat".

Le second prototype propose une vue très différente et met l'accent sur les grands ensembles du métamodèle *Connexion*, les contenus textuels ainsi que les propriétés additionnelles des éléments. Une vue du prototype est présentée dans la figure 3.12. Cette représentation fait donc la part belle aux différents regroupements de données (partie gauche), tandis que la partie centrale se concentre sur les verbatim et la partie droite sur les propriétés additionnelles ainsi que sur la traçabilité. Contrairement à la vue diagrammatique précédente, cette perspective a reçu un accueil plus favorable et sera développée dans la suite du projet.

Cette approche à base de modèles permet également d'obtenir des données intéressantes sur ces derniers et d'offrir certaines fonctionnalités supplémentaires en plus de la seule navigation. Ces fonctionnalités vont du calcul de taux de couverture pour les exigences vis-à-vis de règles de conception qui leurs sont liés ou des calculs sur les thèmes

1. http://www.obeo.fr/pages/obeo-designer

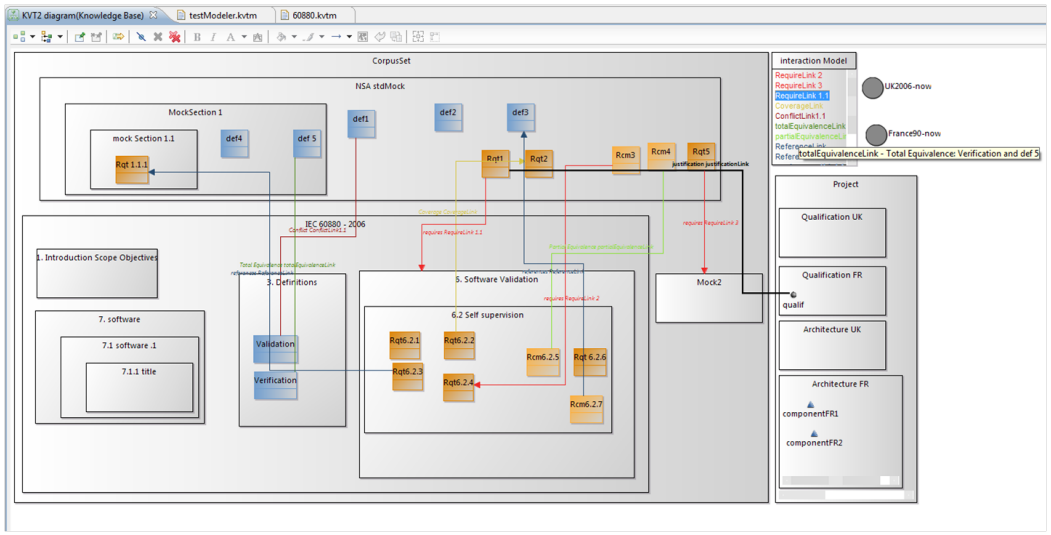


FIGURE 3.11 – environnement de modélisation de modèles KVT basé sur Obeo Designer

associés, à la validation statique de contraintes de bonne formation ou la propagation de propriétés sur les modèles. Ces éléments ont fait l'objet d'un développement propre en Kermeta [MFJ05, JCB+13] et se retrouvent au niveau de la contribution sous la forme de la brique IncrementTools. Cet aspect est appelé à être développé dans le futur du projet CONNEXION et est resté très secondaire dans le cadre de la thèse.

3.4 Discussion et synthèse

Nous avons présenté dans cette section la première partie de la contribution INCREMENT, c'est-à-dire sa perspective modélisation. Nous avons présenté le processus de construction et les différentes évolutions du métamodèle en milieu industriel qui nous a permis de proposer une réponse au premier problème qui nous était posé dans la thèse, à savoir un problème de formalisation du domaine.

En plus de cette formalisation sous la forme d'un métamodèle, nous avons proposer des illustrations d'instanciations théoriques de modèles d'exigences de sûreté mais également un premier ensemble outillé pour l'acquisition automatique et la manipulation de tels modèles. L'analyseur *IncrementParser* permet l'acquisition automatique de modèles. La proposition d'un prototype d'environnement graphique (*IncrementGui*) propose, quant à lui, la manipulation de ces éléments de modèles. Tous les deux permettent la construction du référentiel d'exigences, sorte de base de connaissance et qui était l'un des objectifs de la thèse. Cependant, s'il est possible d'obtenir des métriques sur les modèles [MBC+13], d'effectuer des opérations sur les modèles ou naviguer dans ce dernier via des langages d'action comme *Kermeta* (avec la brique *IncrementTools*) [MFJ05, JCB+13], ces analyses ne répondent pas à l'ensemble des défis qui sont posés, du fait de la nature textuelle des exigences.

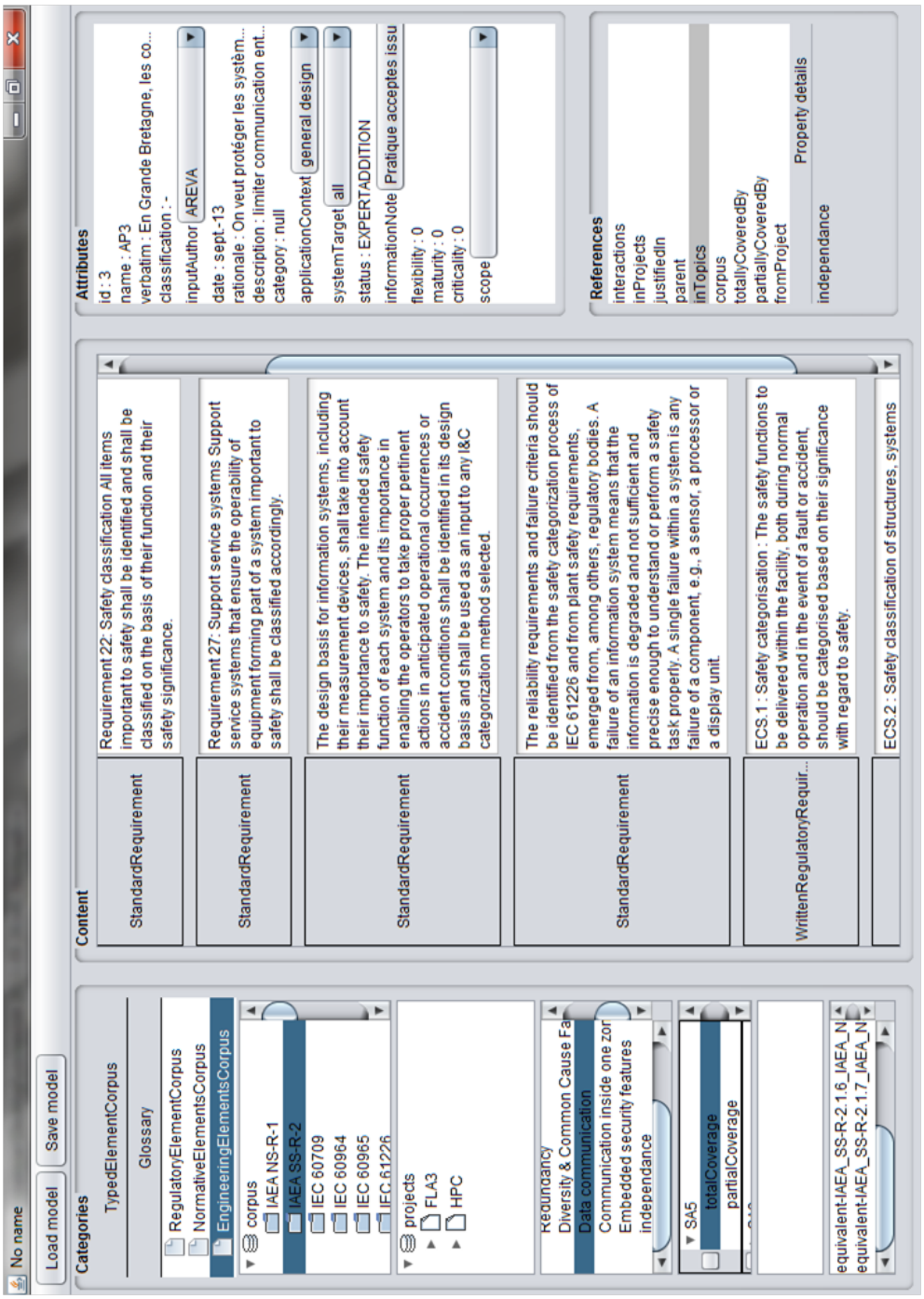


FIGURE 3.12 – environnement INCREMENT GUI pour la navigation et la manipulation de modèles Connexion

Cette contribution INCREMENT-MDE peut-être décrite par la figure 3.13, autour des différentes itérations autour du métamodèle pour décrire le domaine des exigences réglementaires de sûreté nucléaire et des travaux connexes et des outils développés autour de ces métamodèles.

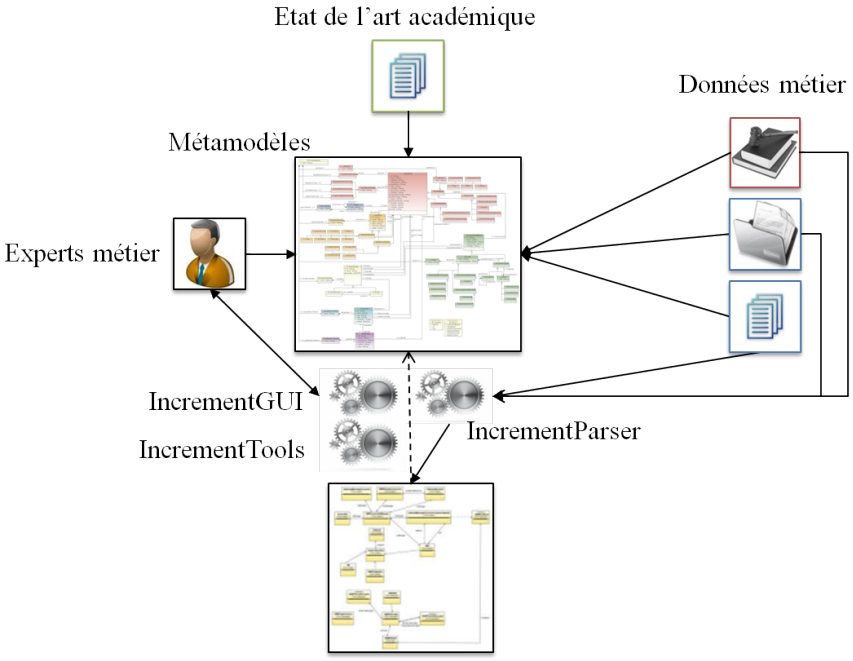


FIGURE 3.13 – INCREMENT, une approche d'Ingénierie dirigée par les modèles

En particulier, s'il est possible d'exprimer des liens de traçabilité, ceux-ci s'avèrent difficilement calculables sur les modèles. De même le regroupement d'exigences dans des ensembles sémantiques selon leur nature peut-être réglé de manière algorithmique (une exigence extraite d'une norme étant, par construction, une exigence normative, le regroupement d'exigences au sein de thèmes ne peut pas être acquis via la seule lecture et interprétation du modèle et de ces attributs.

Cette thématique est abordée dans le chapitre suivant et constitue la seconde partie de la contribution INCREMENT, à savoir l'apport de techniques de recherche d'information pour la traçabilité des exigences réglementaires de sûreté

Chapitre 4

INCREMENT-IR : une approche Recherche d'information

4.1 Recherche d'information pour la traçabilité et l'analyse du corpus

Dans le chapitre 3, nous avons présenté une première contribution autour de la formalisation des exigences du contrôle-commande nucléaire. A travers le métamodèle, nous avons posé les bases d'un langage de modélisation du domaine. Avec les briques IncrementParser, IncrementGui, et IncrementTools, nous avons fourni un ensemble de facilités pour l'acquisition automatique de documents et la modélisation associée (IncrementParser), une interface pour la navigation et la manipulation de modèles (IncrementGui) et des facilités d'analyse sur les modèles (IncrementTools).

Suite à ce travail de modélisation, nous observons une première limitation des analyses possibles sur les modèles de par leur contenu textuel. Une seconde observation vient de l'organisation du domaine lui-même. En effet, les documents réglementaires sont complexes dans leur organisation, en interne et entre eux. Ainsi, regrouper les exigences ou avoir une vision globale des exigences relatives à une préoccupation particulière se révèle difficile, même en ayant une connaissance fine du domaine.

Dans ce chapitre, nous abordons un problème différent de ce que nous avons traité dans le chapitre 3 où il était question de formaliser le domaine et ses interactions. Il est désormais question de l'organiser dans les faits et d'évaluer et fournir des moyens pour assister un ingénieur dans son activité d'instanciation du modèle et la capitalisation de la connaissance.

Nous illustrons tout d'abord la problématique par un exemple sur lequel nous mettons en évidence la complexité de la question de la traçabilité dans les référentiels d'exigences(section 4.2). Nous présentons par la suite les principes de l'approche *Theme* autour de l'analyse de thèmes dans le corpus (section 4.3). Par la suite, nous présentons les évaluations de différentes approches autour de deux activités particulières : la traçabilité des exigences, et la détection et la représentation de thèmes. Nous considérerons donc :

1. une approche statistique et la recherche d'occurrences de termes pour la découverte de thèmes.
2. une approche utilisant un algorithme de clustering pour la définition et la constitution de thèmes et de leurs traces.
3. une approche basée sur l'algorithme d'apprentissage LDA pour la définition du triptyque <Nom, signature, traces> formant un thème.
4. une approche se basant sur le score TF-IDF des documents pour faire émerger leur similarité et pour la traçabilité.

4.2 Tracer manuellement une préoccupation au sein de deux référentiels, un exemple

Dans cette section, nous proposons d'illustrer la complexité du corpus d'exigences réglementaires à travers la recherche manuelle (et par conséquent partielle) des exigences relatives à un thème particulier du logiciel pour les systèmes de contrôle-commande classés de sûreté : la validation et la vérification (V&V) du logiciel. Pour rappel, nous représentons l'organisation du domaine dans la figure 4.1. Cette analyse s'effectuera sur deux référentiels différents, France et Etats-Unis et sur trois niveaux distincts : réglementaire, guide réglementaire et normatif. Nous illustrons cette analyse à partir de courts extraits issus de la réglementation et des normes.

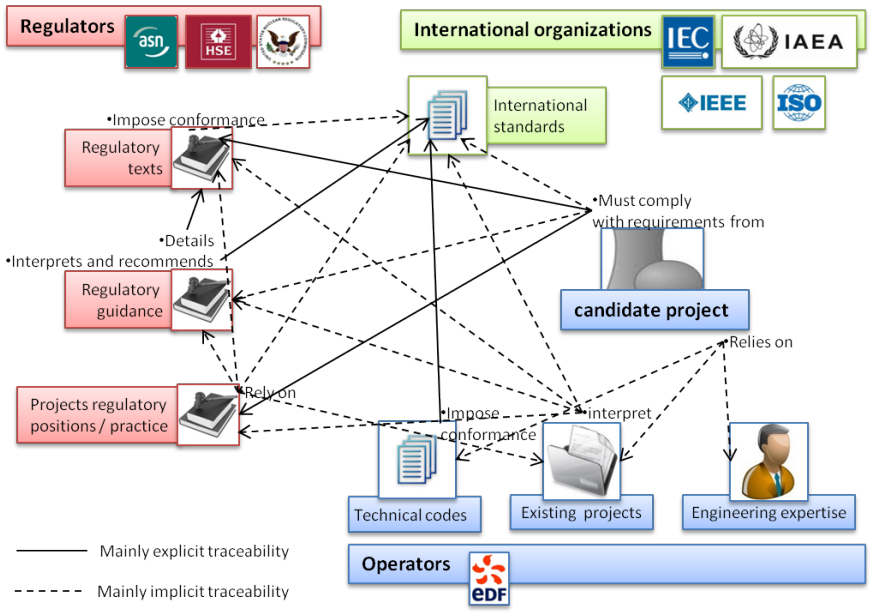


FIGURE 4.1 – Vue globale du domaine des exigences dans l'industrie nucléaire

4.2.1 Au niveau réglementaire

En France, les exigences réglementaires s'expriment dans les règles fondamentales de sûreté et en particulier pour le contrôle-commande, dans la RFS II.4.1.a (dite *RFS Logiciel*) publiée en mai 2000. Les principes et les exigences qu'elle exprime sont référencés au fil du texte. En ce qui concerne la vérification et la validation, il faut considérer l'exigence "Ea 2.1" ainsi que les recommandations qui la suivent et qui sont présentées dans la figure 4.2.

4.2.1.2 Fiabilité
La fiabilité est ici considérée sous un aspect qualitatif.
Ea 2.1. La conception des modules de logiciel et leur documentation doivent permettre l'utilisation de méthodes de vérification et de validation ayant pour objectif de montrer que chaque module fait ce pour quoi il a été spécifié et uniquement cela. Le principe est d'éviter la présence de parties de programme inutilisées, sauf exception justifiée. Dans ce cas, celles-ci doivent être identifiées. Elles doivent être spécifiées, programmées, vérifiées et validées avec le reste du programme.
...
Une pratique acceptable, pour ce qui concerne les méthodes et techniques de vérification, est présentée dans les chapitres 6 (vérification) et 7 (intégration matériel/logiciel) de la publication 60880 (1986) de la Commission Electrotechnique Internationale (CEI).

De même, la réalisation d'essais est une technique acceptable de validation du programme exécutable, notamment du point de vue de ses performances temporelles. En particulier, l'utilisation de cette technique peut s'appuyer sur les dispositions prévues au chapitre 8 de la publication CEI 60880 (1986).
Les deux pratiques de vérification et de validation citées ci-dessus sont complémentaires.

FIGURE 4.2 – Extrait de la RFS II.4.1.a

Aux Etats-Unis, les exigences réglementaires pour le nucléaire sont consignées dans la 10CFR50 (10 Code of Federal Regulation 50) avec en particulier, un extrait que nous présentons dans la figure 4.3.

Au niveau des exigences écrites par les régulateurs, on peut observer un certain nombre de points communs vis-à-vis de la V&V, même si elle n'est pas mentionnée explicitement dans la réglementation américaine (en dehors du terme "tested"). Pour la France, le fait que la V&V doive être effectuée de manière indépendante est explicite. De même, l'adéquation entre le système et sa spécification (pour la validation) est explicitement exprimée.

Dans les deux cas, il est mentionné les plans d'assurance qualité. La notion de conformité aux normes est exprimée à des degrés divers. Les normes sont nommées explicitement dans le cas de la CEI 60880 et des normes IEEE std. 279 et 603. Cependant, si la conformité à la norme 60880 est considérée comme une pratique acceptable, la conformité aux normes IEEE 279 et 603 est exigée par la NRC (Nuclear Regulatory Commission) tandis que les exploitants sont incités à se conformer aux normes et aux meilleures pratiques possibles pour atteindre un niveau de sûreté adéquat.

Par55a(a)(1) : Codes and Standards
(a) Quality standards, ASME Codes and IEEE standards, and alternatives.
(1) Structures, systems, and components must be designed, fabricated, erected, constructed, tested, and inspected to quality standards commensurate with the importance of the safety function to be performed. ...
(h) Protection and safety systems.
(2) Protection systems. For nuclear power plants ... must meet the requirements stated in either IEEE Std. 279 ... or in IEEE Std. 603-1991 ...
(3) Safety systems. Applications ... must meet the requirements for safety systems in IEEE Std. 603–1991 and the correction sheet dated January 30, 1995.
Appendix A to Part 50–General Design Criteria for Nuclear Power Plants
I. Overall Requirements
Criterion 1— Quality standards and records.
Structures, systems, and components important to safety shall be designed, fabricated, erected, and tested to quality standards commensurate with the importance of the safety functions to be performed. ...

FIGURE 4.3 – Extrait de la 10CFR50 américaine, paragraphe 55a et appendice A

On peut néanmoins noter un premier point de divergence dans les termes employés et, surtout, dans la nature des exigences entre l'énumération d'activités d'un côté et les exigences de méthodes et d'objectifs d'un autre.

4.2.2 Au niveau des guides réglementaires

Il n'existe pas de guide réglementaires édictés par l'ASN française. Néanmoins, la RFS mentionne explicitement l'application des chapitres 6, 7, et 8 de la norme CEI 60880 (1986) comme des pratiques acceptables. Il existe également les Règles de Conception et de Construction (RCC et en particulier le RCC-E, pour les matériels électriques et donc pour le contrôle-commande), éditées par EDF et approuvées par l'ASN. Ces dernières ne peuvent cependant pas être interprétées comme des guides réglementaires. En particulier, le RCC-E demande l'application et la conformité vis-à-vis de plusieurs normes internationales telles que la CEI 60880, la CEI 62138, etc.

Aux Etats-Unis, la V&V est décrite partiellement dans le guide réglementaire RG 1.168 dont nous présentons un extrait dans la figure 4.4. Ce guide est relativement petit, onze pages, mais fournit un lien de traçabilité vers la 10CFR50 et vers la norme IEEE 1012. Cette dernière est une norme généraliste sur la V&V des logiciels.

A travers le guide réglementaire, on peut voir les liens de traçabilité (explicite) avec les normes IEEE 1012 et 1028, mais aussi que ces normes sont interprétées d'une certaine façon par le régulateur (correspondant à la pratique réglementaire), en décidant de refuser, d'accepter ou d'interpréter leur contenu. De cette manière, les textes et guides réglementaires et les normes telles qu'elles sont interprétées constituent l'ensemble du référentiel d'exigences.

This regulatory guide endorses IEEE Std 1012-1998, "IEEE Standard for Software Verification and Validation," and IEEE Std 1028-1997, "IEEE Standard for Software Reviews and Audits." IEEE Std 1012-1998, with the exceptions stated in the Regulatory Position, describes a method acceptable to the NRC staff for complying with parts of the NRC's ...
C. REGULATORY POSITION
IEEE Std 1012-1998, "IEEE Standard for Software Verification and Validation", provides methods that are acceptable to the NRC staff for meeting the requirements of 10 CFR Part 50 as they apply to the verification and validation of safety system software, subject to the exceptions listed in these Regulatory Positions. ...
The annexes to IEEE Std 1012-1998 and IEEE Std 1028-1997 contain information that may be useful, but the information in these annexes should not be viewed as the only possible solution or method. ...

FIGURE 4.4 – Extrait du guide réglementaire RG1.168 publié par la NRC et l'approbation de la norme IEEE 1012 pour la V&V du logiciel

4.2.3 Au niveau des normes internationales

Alors que les documents réglementaires sont publiquement accessibles, nous passons une première frontière avec les normes et autres documents techniques puisque ceux-ci deviennent propriétaires et deviennent de facto moins accessibles. Au delà des traces extraites de l'analyse des deux référentiels différents, qui montrent une certaine cohérence d'objectifs dans le sujet traité, nous retrouvons, au niveau normatif, deux normes internationales : une partie de la norme CEI 60880 et la norme IEEE 1012.

Si les deux documents abordent la V&V, les perspectives choisies sont cependant différentes.

Les chapitres 8, 9, et 10 de la norme CEI 60880 traitent des aspects suivants (extrait de la table des matières de l'édition 2006 de la norme) :

- 8 Vérification du logiciel
- 8.1 Processus de vérification du logiciel
- 8.2 Activités de vérification du logiciel
- 9 Aspects logiciels de l'intégration du système
- 9.1 Aspects logiciels du plan d'intégration du système
- 9.2 Intégration du système
- 9.3 Vérification du système intégré
- 9.4 Procédures de résolution de défaut
- 9.5 Aspects logiciels du compte rendu de vérification du système intégré
- 10 Aspects logiciels du plan de validation
- 10.1 Aspects logiciels du plan de validation système
- 10.2 Validation du système
- 10.3 Aspects logiciels du compte rendu de validation du système
- 10.4 Procédures de résolution de défaut

De son côté, la norme IEEE 1012 traite des éléments suivants :

- 4. Software integrity levels

- 5. Software V&V processes
- 5.1 Process : Management
- 5.2 Process : Acquisition
- 5.3 Process : Supply
- 5.4 Process : Development
- 5.5 Process : Operation
- 5.6 Process : Maintenance
- 6. Software V&V reporting, administrative, and documentation requirements
- 6.1 V&V reporting requirements
- 6.2 V&V administrative requirements
- 6.3 V&V documentation requirements
- 7. Software V&V plan outline
- 7.1 SVVP section 1 : Purpose
- 7.2 SVVP section 2 : Referenced documents
- 7.3 SVVP section 3 : Definitions
- 7.4 SVVP section 4 : V&V overview
- 7.5 SVVP section 5 : V&V processes
- 7.6 SVVP section 6 : V&V reporting requirements
- 7.7 SVVP section 7 : V&V administrative requirements
- 7.8 SVVP section 8 : V&V test documentation requirements

On observe que la norme CEI 60880 exprime ses préoccupations (et donc ses exigences) sous la forme d'objectifs et de moyens. La philosophie de la norme IEEE 1012 est différente, puisqu'elle se détaille à travers un certain nombre d'activités et de tâches particulières avec des documents en entrée et en sortie de ces tâches. En particulier, les attendus des sorties de ces activités concernent certaines problématiques comme la traçabilité, les interfaces, la gestion des risques, la sécurité, etc. Si la norme CEI 60880 exprime des exigences quant au contenu de la documentation, celle-ci ne dicte pas, dans ses exigences tout du moins (elle propose un exemple plan dans ses annexes), la façon dont cette documentation doit être rédigée.

4.2.4 Synthèse de l'analyse

Après cet exemple de recherche manuelle de traces à propos du thème "V&V du logiciel" dans les deux référentiels français et américain, nous pouvons faire un certain nombre d'observations.

La première est l'éclatement des exigences à travers toute la hiérarchie du corpus.

La seconde est l'orientation des exigences qui sont exprimées dans les différents documents, et en particulier dans les normes. Cette divergence dans la "nature" des exigences, qu'elles expriment des objectifs, des moyens, des activités, propose un challenge important pour la certification. En effet, la comparaison d'exigences autour d'une même problématique peut mener à la mise en évidence d'exigences qui peuvent être équivalentes, en conflit mais aussi des exigences qui soient simplement différentes, alors qu'elles traitent du même sujet. Cette différence peut être expliquée pour le cas présent par le fait que la norme CEI 60880 soit une norme spécifique au nucléaire et, de

plus, spécifique au logiciel remplissant des fonctions de catégorie A. La norme IEEE 1012 est une norme généraliste, non dédiée à un domaine particulier, et exprime des exigences sans distinction de classification. C'est à travers la définition de niveaux de sûreté (chapitre 4 de la norme), que le périmètre d'application des exigences se dessine.

Comme nous le mentionnions dans le chapitre 1, il existe un écart significatif entre la classification de sûreté de ces deux pays. Les niveaux de sûreté décrits dans la norme IEEE 1012 (qui sont similaires aux niveaux de sûreté décrits dans la DO-178B pour l'aéronautique, l'ISO 26262 pour l'automobile, ou pour la CEI 61508) viennent proposer un autre niveau de classification et tend à complexifier l'analyse des exigences à appliquer pour un système particulier.

La réglementation et les normes évoluent au cours du temps, comme nous l'avons montré dans le chapitre 1. Par exemple, la RFS Logiciel, publiée en 2000, n'a pas été mise à jour depuis et ne peut donc pas tenir compte des nouvelles normes publiées alors que les pratiques, elles, en tiennent compte, ce qui a un impact sur la traçabilité.

Pour appréhender la complexité de ces corpora dans leur globalité, il est important de pouvoir les réduire en sous-ensembles thématiques plus petits et donnant l'accès aux différents documents, malgré l'absence de liens explicites de traçabilité entre eux. La contribution INCREMENT-IR s'articule autour de la définition et de la structuration de ces petits sous-ensembles de thèmes, leur définition et leur constitution. Nous la présentons sous la forme de l'approche *Theme* et de l'évaluation de différentes techniques de recherche d'information pour l'organisation de référentiels en thèmes (ou topics) et pour la traçabilité.

4.3 Principes de l'approche Theme

4.3.1 Définition

A propos de la traçabilité, Gotel et Morris proposent de définir la traçabilité en termes d'individus à tracer, de signes distinctifs, et de traces [GM11]. De la même manière, nous définissons l'approche *Theme* autour des définitions de thèmes, de signatures et de traces.

Définition 4.1 *Définition d'un thème (ou topic)*

1. *Un thème est une considération, une préoccupation, un sujet traité, au sein d'un corpus. Par exemple, la défaillance de cause commune, la maintenance, la gestion de configuration, la classification de sûreté, etc. Un thème est constitué d'un triptyque <nom, signature, traces>.*
2. *La signature d'un thème est l'ensemble des marques permettant d'identifier le thème. Cette définition est reprise de la proposition de Gotel et Morris [GM11] ("an identifying mark made by, or associated with a particular purpose, an animate or inanimate object"). Dans notre cas, il s'agit des termes et mots-clés autour du thème.*

3. *Les traces d'un thème sont la collection de l'ensemble des extraits textuels du corpus qui sont relatifs au thème.*

La figure 4.5 présente le processus de définition et d'acquisition des thèmes dans le corpus d'exigences. Elle se décrit autour des activités suivantes :

1. acquisition des différents documents sous une forme interprétable, dans notre cas un fichier texte ;
2. définition des thèmes observés dans le corpus ;
3. obtention des éléments de signature de ces thèmes
4. recherche des traces thèmes en utilisant leur nom et signature.

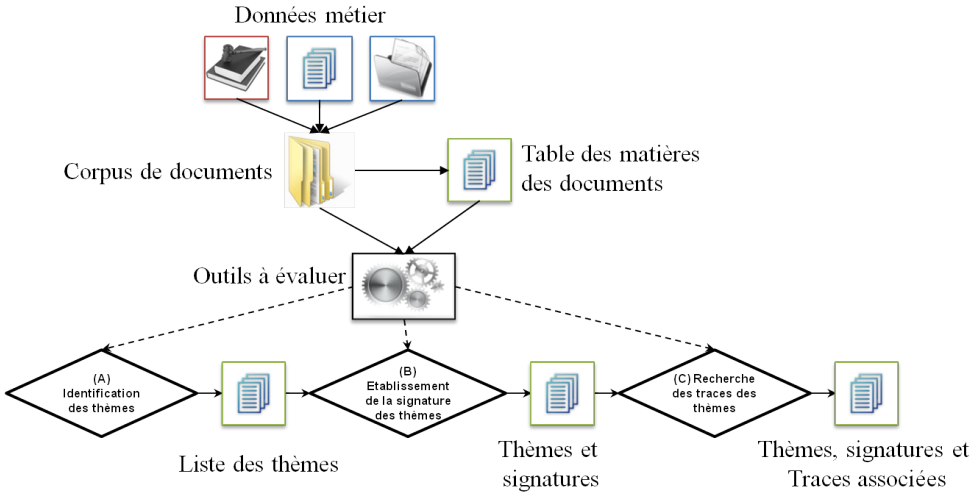


FIGURE 4.5 – L'approache Theme pour la définition et la constitution de thèmes

4.3.2 Mise en place de l'étude

Pour l'étude, nous avons choisi de nous intéresser à un corpus de huit normes internationales, publiées par le sous comité SC45A de la CEI, dédiées au contrôle-commande nucléaire, et appliquées dans la pratique industrielle française. Nous nous concentrons ici sur 8 normes et non sur un panel complet de documents réglementaires pour manipuler une structure homogène. D'une part, ces documents constituent la base d'exigences de travail, et non les textes réglementaires. De plus, nous n'avons pas considéré de textes, de guides ou de normes en rapport avec les pratiques d'autres pays qui sont moins connues. Il s'agit des normes dont nous avons présenté l'évolution chronologique dans le chapitre 1 :

- IEC60880-2006 Software Aspects for Computer-Based Systems Performing Category A Functions
- IEC60987-2007 Hardware Design Requirements for Computer-Based Systems

- IEC61226-2009 Classification of Instrumentation and Control Functions
- IEC61500-2009 Data Communication in Systems Performing Category A Functions
- IEC61513-2011 General Requirements for Systems
- IEC62138-2004 Software Aspects for Computer-based Systems Performing Category B or C Functions
- IEC62340-2007 Requirements for Coping with Common Cause Failure (CCF)
- IEC62566-2011 Development of HDL-programmed Integrated Circuits for Systems Performing Category A Functions

Cet ensemble offre une couverture globale du domaine depuis la perspective d'un éditeur. Le tableau 4.1 présente des informations de haut niveau les concernant.

TABLE 4.1 – Constitution d'un corpus de 8 normes internationales pour l'analyse de thèmes

standard	1st year of publication	#pages (English part)	structure	Documents indexés	Références normatives	Définitions	clauses* (exigences ("shall"), recommandations ("should"),
IEC60880-2006	1986	110	15 sections et 10 annexes normatives ou informatives	125	8	43	355
IEC60987-2007	1989	30	13 sections et 3 annexes informatives	47	10	18	111
IEC61226-2009	2009	32	7 sections et 1 annexe informative	49	19	22	87
IEC61500-2009	1996	14	10 sections	24	10	8	54
IEC61513-2011	2001	98	8 sections et 5 annexes informatives	153	27	62	291
IEC62138-2004	2004	47	6 sections	91	2	36	225
IEC62340-2007	2007	22	9 sections et 1 annexe informative	27	11	26	39
IEC62566-2011	2011	52	17 sections et 2 annexes informatives	106	7	14	263
totals		405	107 éléments de structure	622	94	229	1425

Pour cette analyse nous avons découpé manuellement les 8 normes à la granularité du plus petit paragraphe. Ce découpage manuel a mené à la constitution d'un corpus de 622 extraits de tailles variant du verbatim complet de chaque norme, à chaque section et sous-section. Nous avons aussi extrait séparément les tables des matières des normes pour une analyse manuelle des différents thèmes à retrouver. Nous avons donc volontairement introduit une forme de redondance dans le contenu des documents de notre corpus d'analyse puisque chaque verbatim de chapitre ou de section contient aussi ses sous-sections. Cette redondance peut avoir un effet sur l'étude, et nous analyserons cet effet par la suite. Cette redondance présente également l'avantage de manipuler le corpus indifféremment selon sa granularité. Nous avons également l'intuition que cette redondance, en augmentant les occurrences des termes, peut favoriser l'émergence de termes clés.

Dans la suite du chapitre, nous abordons la question de la définition et la constitution de thèmes à travers l'exploration de plusieurs techniques pour la réalisation des trois tâches pour la constitution des thèmes.

4.4 Analyse du corpus pour la détection et la constitution de thèmes

Dans cette section, nous abordons les différentes approches et techniques que nous avons utilisées pour la constitution de thèmes :

1. une approche naïve à partir de statistiques ;
2. une approche basée sur des algorithmes de clustering ;
3. une approche basée sur les techniques de modélisation de topics et d'apprentissage ;
4. la recherche d'information pour la traçabilité et la constitution de thèmes.

4.4.1 Recherche de thèmes avec une approche statistique

4.4.1.1 Intuition

Pour identifier les thèmes, nous analysons les tables des matières des documents. Celles-ci présentent un condensé des préoccupations abordées dans le corpus. En enlevant les mots vides, une analyse des mots apparaissant le plus devrait permettre de faire ressortir les thèmes principaux qui sont traités dans les documents. L'utilisation de table des matières a déjà été proposée par Cleland et al. [CHCGE10] pour évaluer le domaine auquel appartient le document.

4.4.1.2 Matériel de l'évaluation et opérations

Nous nous basons pour cela sur le regroupement des tables des matières des 8 documents en un seul fichier que nous traitons. Sur ce document, nous effectuons les pré-traitements suivants : suppression des mots vides, remplacement des synonymes et racinisation.

Pour rappel la suppression des mots vides revient à retirer tous les mots trop communs pour porter une information pertinente (comme les articles, les déterminants, pronoms, adjectifs communs, etc.) dans un texte. La racinisation [VRRP80] est l'action de réduire un terme en sa racine (lemme) pour rassembler les termes ayant la même racine. Par exemple, le titre "specific requirements related to blank integrated circuits" deviendra "specif requir relat blank integr circuit".

Cette analyse a été effectuée à l'aide de l'outil TextStat [1]. Ce programme permet de faire l'analyse statistique d'un texte en comptant un certain nombre de paramètres comme le nombre de lettres, de mots, de phrases. Il fournit également un tableau de tous les mots du texte avec le nombre d'occurrences que nous présentons en figure 4.2.

1. http://neon.niederlandistik.fu-berlin.de/en/textstat/

4.4.1.3 Variations de l'étude

Nous avons fait varier les paramètres de l'étude en faisant varier la granularité des tables des matières (en ne retenant que les chapitres de plus haut niveau ou l'ensemble de leur contenu), ou en agissant sur la complexité des termes à analyser (en dissociant ou associant les concepts multi-mots, comme par exemple pour la défaillance de cause commune. La défaillance de cause commune est un concept très particulier, au coeur des exigences sur le contrôle-commande. Pris séparément, le terme "défaillance" représente un thème particulier, "cause" et "commune" sont considérés comme deux mots vides par défaut. De la même façon, les expressions composées comme "spécification d'exigences" sont analysées de manière groupée ou séparée.

Pour la gestion des synonymes, la meilleure approche serait d'utiliser une ontologie du domaine. Cependant, celle-ci n'existe pas et sa définition est en dehors du périmètre de nos travaux. Pour palier à cette absence, nous avons utilisé le dictionnaire WordNet [Mil95] développé par George Miller et maintenu à l'université de Princeton. Le regroupement des synonymes permet d'augmenter la fréquence des termes et ainsi les faire remonter comme des mots clés plus facilement. WordNet est un dictionnaire généraliste sur la langue anglaise et a déjà été utilisé dans d'autres travaux académiques autour des exigences comme dans le projet REVERE [SRG02].

L'utilisation de Wordnet pour la gestion des synonymes dans ce contexte s'est avérée peu pertinente. En effet, le terme "design" à été remplacé via notre algorithme de substitution par le terme "project". Dans WordNet, le mot "design" a pour synonymes : *[aim, blueprint, conception, contrive, designing, excogitation, figure, innovation, intent, intention, invention, pattern, plan, project, purpose]*. Appliqué au domaine nucléaire, tout ces termes ne sont pas des synonymes valides. Pire, "project" et "plan" sont des synonymes faux positifs puisqu'ils représentent des aspects particuliers à traiter, et qui sont différents de la conception. Nous avons finalement renoncé à la substitution de synonymes.

4.4.1.4 Résultat de l'analyse statistique

Le tableau 4.2 présente un échantillon de la remontée statistique sur l'analyse des tables des matières après les différents traitements. Afin de limiter la taille des résultats, nous avons établi empiriquement une valeur de coupure sous laquelle les termes remontés ne sont pas pris en compte. Dans le cas de cette étude, nous comptons le nombre d'occurrences des termes remontés. La valeur de coupure a été fixée à 5 occurrences au minimum. On observe que parmi les termes remontés, certains sont encore très génériques (par exemple, *"requir"*, *"gener"*) ou trop spécifiques comme les acronymes (par exemple, *"hpd"*), et qui viennent ajouter du bruit dans les résultats.

Ces différences dans les termes apparaissent à cause de l'hétérogénéité des normes : niveau d'abstraction, auteurs, organisation du document, perspective technique ou généraliste de la norme, etc. Certains termes trop spécifiques ne conviennent pas pour la capture d'un nombre restreint de thèmes qui couvriraient la globalité du corpus.

Dans le tableau 4.2, nous retrouvons des éléments racinisés mais aussi regroupés pour prendre en compte les concepts multi-mots. Par exemple, *"common_ caus_ failur"* a été

TABLE 4.2 – Approche Statistique pour l'identification de thèmes. Les nombres associés aux termes représentent le nombre d'occurrences de ces termes dans les tables des matières des documents

requir	56	specif	17	aspect	10	procedur	8
softwar	44	implement	15	failur	10	review	8
gener	42	common_caus_failur	13	integr	10	safeti	8
design	35	system_valid	13	overal	10	gener_requir	7
system	33	tool	13	instal	9	i&c_system	7
document	27	verif	13	oper	9	test	7
plan	22	categori	12	qualif	9	develop	6
modif	18	mainten	11	relat	9	life_cycl	6
function	17	softwar_aspect	11	fault	8	perform	6
hpd	17	system_integr	11	i&c	8	pre-develop	6

dissocié de *"failur"*", puisqu'ils représentent deux notions similaires mais différentes. Certains termes particulièrement pertinents ont un score relativement bas, sous la valeur coupure initiale que nous avions proposé (moins de 5 occurrences).

Les résultats de cette analyse sont mitigés. Il n'y a pas de mots-clés qui émergent nettement au milieu du bruit. Cependant, une interprétation de cette table permet de proposer un certains nombre de thèmes. Les éléments grisés du tableau 4.2 sont considérés comme des candidats possibles pour la définition de thèmes. A partir de ces entrées, nous avons pu proposer une liste de 19 thèmes généraux, couvrant les 8 normes.

Le défaut de cette approche est que la proposition des thèmes repose très fortement sur l'interprétation des données et qu'elle n'offre aucun support à la traçabilité. L'analyse statistique ne permet pas d'aller au delà de la première étape d'identification des thèmes. Pour la constitution des éléments de signature et la découverte de traces, il est nécessaire d'envisager d'autres approches. Dans la section suivante, nous abordons une approche basée sur un algorithme de *clustering*.

4.4.2 Identification et regroupement avec des algorithmes de clustering

4.4.2.1 Intuition

La seconde approche que nous présentons utilise des algorithmes de *clustering* pour la construction de regroupements de documents similaires, en plus d'offrir les traces des clusters établis.

Du fait du recouvrement et de la hiérarchie des thèmes dans le corpus, tous les algorithmes de clustering ne conviennent pas. K-mean, bisecting sont des algorithmes

qui proposent des répartitions dans des thèmes exclusifs, donc qui ne se recouvrent pas. Les algorithmes de clustering classiquement utilisés, par exemple par Niu et Mahmoud [NM12], ne sont donc pas de bons candidats pour l'analyse, tandis que des algorithmes autorisant des recouvrements comme Lingo ou STC (Suffix Tree Clustering) peuvent être de bon candidats.

4.4.2.2 Matériel de l'évaluation et opérations

L'algorithme *Lingo* du projet *Carrot²* permet un tel recouvrement de topics. Le projet *Carrot²*[2] est un projet open source autour des moteurs de clustering [OW04]. Il s'intègre à Lucene[3] (que nous avons utilisé pour la dernière analyse) et propose trois algorithmes de clustering, K-moyennes (K-mean), Suffix Tree Clustering (STC), et Lingo. Lingo et STC sont deux algorithmes qui proposent des regroupements se recouvrant ainsi que la possibilité d'associations multi-mots, ce qui était une limite de l'approche statistique évaluée précédemment. K-mean reste mono-mot et est non recouvrant. Il a été écarté de l'analyse.

D'après la documentation de Carrot² [OW04], les algorithmes de clustering fonctionnement mieux lorsqu'ils analysent de courts extraits, idéalement des tables des matières ou des résumés. Nous avons donc utilisé les tables des matières de notre analyse précédent dans le cadre de notre recherche de thèmes.

Pour cette analyse, nous avons considéré chaque entrée des tables des matières comme des documents (au sens indexation) séparés. Nous avons retiré les titres qui faisaient parti des patrons de constructions des documents et qui revenaient de manière récurrentes et n'auraient offert aucune information (par exemple, pour la norme CEI 62138, les sections *"6.4.2 Inputs"*, *"6.4.3 Contents"*, et *"6.4.4 Properties")* et/ou celles qui n'avaient pas de sens particulier (*"general"*). Sur les 622 documents originellement indexés, il n'en reste plus que 463 pour l'analyse.

4.4.2.3 Variations de l'analyse

Comme tout algorithme de clustering, Lingo doit être paramétré avec le nombre désiré de regroupements. Nous avons donc fait varier de manière itérative le nombre de regroupements à générer entre 30 et 160 clusters. Avec peu de clusters, l'algorithme a tendance à proposer des groupes avec une population importante et finalement assez peu corrélés. Avec un nombre plus important de clusters, outre les groupes à forte population, l'algorithme va également proposer des clusters plus petits mais aussi plus précis donc plus pertinents Les meilleurs résultats en terme de distribution apparaissant pour cette dernière valeur. Avec le nombre de regroupements grandissant, nous avons établi une valeur de coupure pour la taille des regroupements générés et que nous prenons en compte pour la détermination des thèmes (quatre documents dans un cluster).

2. http://project.carrot2.org/
3. http://lucene.apache.org/

4.4.2.4 Résultats de l'approche par clustering

Les résultats du clustering par Lingo proposent un ensemble de 158 regroupements, une grande partie d'entre eux contenant seulement deux documents (46), trois documents (16) ou quatre documents (14). Ces documents ne dépassent pas la taille définie par la valeur de coupure et ne sont pas conservés. Nous présentons un extrait des regroupements générés dans la figure 4.6.

Specification (18 docs, score : 27,7)
60880-C6.1-specification of software requirements
60880-ZH-tools for production and checking of specification design and implementation
61513-C6.2.2-system requirements specification
61513-C6.2.3-system specification
61513-C6.2.3.4-software specification
61513-C6.2.4.2.1-functional validation of the application functions requirements specification
61513-C6.4.2-system requirements specification documentation
61513-C6.4.3-system specification documentation
61513-C6.5.2-generic and application specific qualification
62138-C5.3-software requirements specification
62138-C6.3-software requirements specification
62340-C6-requirements to overcome faults in the requirements specification
62340-C6.1-deriving the requirements specification for the I&C from the plant safety design base
62566-C10.3-specific aspects of system integration
62566-C6-requirements specification
62566-C6.2-functional aspects of the requirements specification
62566-C7.2-component requirements specification
62566-C7.4.4-specific requirements related to the blank integrated circuits

Software Requirements Specification (3 docs, score : 25,96)
60880-C6.1-specification of software requirements
62138-C5.3-software requirements specification
62138-C6.3-software requirements specification

FIGURE 4.6 – Extrait de la constitution de regroupements par Lingo avec une construction paramétrée de 160 regroupements et une valeur de coupure de 4

La figure 4.6 présente le cluster *"Specification"*, son nom ainsi que les portions des normes (les sections) remontées par Lingo. Quand Lingo crée un regroupement, il crée également des regroupements spécialisés contenus dans le premier, par exemple le regroupement *"software requirements specification"* de la figure. STC offre moins de regroupements et moins de diversification que Lingo. Par conséquent les regroupements remontés par STC sont sensiblement plus gros et moins pertinents dans notre contexte.

Les regroupements retrouvés par Lingo sont différents de ceux que nous aurions identifiés manuellement en analysant les tables des matières. Cela s'explique par le fait que les sous-regroupements remontés dépendent du niveau d'abstraction (le nombre de re-

groupements paramétré) choisi et qu'ils sont contenus dans des regroupements de niveau plus important. Concrètement, cela amène la création de hiérarchies de regroupements qui peuvent être intéressants pour des recherches thématiques plus fines. Si l'on retire les sous-regroupements complètement contenus dans des parents, il reste 61 regroupements à granularités différentes. Ces 61 regroupements sont difficilement comparables aux 19 thèmes remontés par l'analyse des données remontées par TextStat.

Néanmoins, certains des regroupements remontés par Lingo sont directement des thèmes identifiés précédemment tandis que la plupart des autres peuvent être fondus dans les 19 identifiés. L'usage d'un tel algorithme est donc utile pour construire à la fois les thèmes et leurs traces. Ces traces sont formées par l'agrégation des sections appartenant aux clusters remontés. Autre observation intéressante, les noms des regroupements sont construits et ne nécessitent aucune analyse supplémentaire.

A cause de la racinisation, les documents contenant *"specification"* et *"specific"* ou *"integrated"* et *"integration"* ont été regroupés alors que *specific* et *integration* sont des faux positifs du regroupement, même si le nom du thème reste correct.

Pour l'évaluation de l'approche, nous avons effectué une mesure de couverture du corpus par les traces des thèmes remontés après fusion. A la fin du traitement, l'algorithme laisse 36 documents (sur les 463) en dehors de tout regroupement. Ces sections non affectées représentent des préoccupations très particulières et très précises dans le corpus et, par conséquent, sont présentes de manière unique dans le corpus. Les documents faisant parti de faux positifs dans les regroupements et ceux qui ne passent pas la valeur de coupure sans être présent dans un regroupement conservé sont ajoutés à la liste des regroupements non affectés.

Au final, l'algorithme atteint un taux de couverture du corpus de 86,82%, ce qui est un résultat acceptable pour une telle démarche [CHCGE10] et pour les thèmes que nous avons établis.

La principale limitation de l'approche vient du nombre de regroupements désirés qui est à déterminer de manière empirique. Un faible nombre de regroupements mène à des regroupements de grande tailles, peu pertinents. Le gain en précision des regroupements demande la création d'un nombre important de sous-regroupements (158 regroupements pour obtenir au final 60 regroupements pour 19 thèmes) et nécessite un fort travail d'analyse par la suite.

Cependant, les résultats obtenus sont acceptables, d'autant plus que l'algorithme fournit à la fois les thèmes et les traces, même s'il ne donne pas les signatures. Celles-ci sont importantes pour envisager les synonymies et faciliter les correspondances de thèmes quand le vocabulaire (et donc les signatures) varie en fonction du contexte. C'est le cas ,par exemple, pour le passage des publications du monde CEI à celles du monde IEEE. Elles sont également importantes pour offrir un mécanisme d'apprentissage pour l'analyse de corpora plus étendus et se basant sur les thèmes établis.

L'approche suivante que nous évaluons est justement basée sur les méthodes d'apprentissage pour la définition et la modélisation de thèmes.

4.4.3 Identification et modélisation de thèmes par apprentissage

4.4.3.1 Intuition

Dans les deux approches précédentes, nous avions défini un certain nombre de thèmes ainsi qu'une liste de leurs traces après interprétation d'une liste de regroupements. Nous n'avons cependant pas défini complètement l'ensemble des éléments des thèmes, à savoir un ensemble <nom, signatures, traces>. Ces éléments de signature sont d'autant plus importants que certains thèmes partagent certains éléments de signature. Il convient dès lors, de pondérer la contribution de ces mots clés à ces thèmes.

Les algorithmes de modélisation de thèmes, comme LDA permettent (a) de regrouper les documents partageant des caractéristiques communes, et (b) d'obtenir des informations sur les termes que partagent les documents des regroupements.

4.4.3.2 Matériel de l'évaluation et opérations

Pour cette analyse, nous nous sommes basés sur la brique MALLET (MAchine Learning for LanguagE Toolkit), développé à l'université du Massachusets [4] [McC02]. MALLET est un projet open source pour l'analyse statistique du langage naturel, la classification de documents, le clustering, la modélisation de topics, l'extraction des informations. Pour la modélisation de topics, MALLET propose plusieurs implémentations des algorithmes LDA (Latent Dirichlet Allocation), PA (Pachinko Allocation), H-LDA (Hirearchical LDA). L'approche est complètement automatisée à partir de l'obtention des différents documents du corpus et a été utilisée par Henßet al. [HMM12] pour la construction de FAQs ou par Asuncion et al [AAT10] pour la modélisation de thèmes d'exigences.

Au niveau des données, nous avons considéré le corpus complet des 622 documents que nous avions générés auparavant. Nous analysons ainsi non seulement les titres mais aussi les contenus des documents.

4.4.3.3 Latent Dirichlet Allocation

LDA caractérise chaque thème i comme une distribution de probabilité sur l'ensemble des mots du corpus (notée φ_i). Le tableau 4.3 présente un extrait de la distribution des termes clés d'un thème identifié sur le corpus de 622 documents que nous avons constitué.

Comme chaque document peut appartenir à plusieurs thèmes, LDA définit également une distribution de probabilité des thèmes pour chaque document j (notée θ j). Comme pour les algorithmes de clustering, le nombre de thèmes à construire est un paramètre de l'analyse et celui-ci doit être déterminé de manière empirique.

L'objectif de LDA est de trouver la meilleure approximation de ces deux distributions de probabilité en maximisant la fonction suivante :

$P(\theta, \varphi) = \prod_{i=1}^{K} P(\varphi_i) \prod_{j=1}^{M} P(\theta_j) \prod_{t=1}^{N_j} P(Z_{j,t}|\theta_j) P(W_{j,t}|\varphi_{Z_{j,t}})$

avec : K, le nombre de thèmes à déterminer,

4. http://mallet.cs.umass.edu/

TABLE 4.3 – Distribution des termes pour un thème remonté par MALLET

weight	count	word / phrase	weight	count	word / phrase
0,10182	1062	verification	0,01381	144	hardware
0,04765	497	design	0,01304	136	phases
0,04334	452	development	0,01198	125	team
0,03337	348	clause	0,01112	116	integration
0,03193	333	activities	0,04501	55	life cycle
0,03030	316	process	0,02046	25	verificaton team
0,02982	311	requirements	0,01882	23	verification actitivites
0,02876	300	specification	0,01718	21	development process
0,02445	255	phase	0,01309	16	pre developed
0,01879	196	cycle	0,01146	14	verification verification
0,01755	183	developed	0,01146	14	requirements specification
0,01553	162	implementation	0,01146	14	design implementation
0,01467	153	life	0,00818	10	clause verification
0,01400	146	aspects	0,00818	10	development team
0,01390	145	project	0,00655	8	requirement clause

M, le nombre de documents du corpus,
φ_i, la distribution de probabilités des termes du topic i,
θ_j, la distribution de probabilités des thèmes pour le document j,
N_j, le nombre de termes (ou tokens) du document j,
$W_{j,t}$, est le terme à la position t du document j,
$Z_{j,t}$ est le thème pour le terme à la position t du document j,
$\prod_{t=1}^{N_j} P(Z_{j,t}|\theta_j)P(W_{j,t}|\varphi_{Z_{j,t}})$ la distribution de probabilité terme / document / topic.

LDA utilise également des valeurs a priori α (resp. β) qui représente la distribution des thèmes par document (resp. la distribution des termes par thème) mais dont la valeur est ajustée automatiquement par l'implémentation de LDA de MALLET pour améliorer la précision des thèmes.

4.4.3.4 Variations de l'analyse

A l'instar de Lindo, LDA doit être paramétré avec le nombre de thèmes désiré. Nous avons donc fait varier de manière itérative le nombre de regroupements à générer : 20, 25, 30, 50 et 100.

4.4.3.5 Résultats de l'approche par apprentissage

Les résultats varient en fonction des regroupements et ne sont pas reproductibles, d'une analyse à une autre en fonction des itérations (1000 cycles pour chaque analyse). La figure 4.7 présente un extrait d'un des regroupements produits et que l'on pourrait assimiler au thème de la V&V.

topic id="12" ← Identifier le thème
Keywords: verification, design, development, clause, activities, process, requirements, specification, phase, cycle, developed, implementation, life, aspects, project, hardware, phases, team, integration, life cycle, verification team, verification actitviies ...

Distribution des thèmes dans les documents

60880-C8_software_verification.txt	12→0.3880	11→0.1574	
60880-C8.1-software_verification_process.txt	12→0.4310	11→0.1092	
60880-C8.2-software_verification_activities.txt	12→0.3485	11→0.1615	13→0.1
60880-C8.2.1-verification_plan.txt	12→0.3393		
60880-C8.2.2-design_verification.txt	16→0.14602	12→0.14602	
60880-C8.2.3-implementation_verification.txt	12→0.2881	11→0.1944	0→0.13095
60880-C8.2.3.1-verification_of_implementation_with_general_purpose_languages.txt	12→0.2938	11→0.1498	0→0.1070
60880-C8.2.3.2-verification_of_implementation_with_application-oriented_languages.txt	11→0.2533		
60880-C8.2.3.3-verification_of_configuration_of_pre-developed_software.txt	12→0.3166	11→0.1380	13→0.1218

FIGURE 4.7 – Extrait d'un regroupement constitué par apprentissage avec MALLET et l'algorithme LDA

LDA propose à la fois un ensemble de signatures pondérées, et les traces associées. C'est un algorithme qui autorise le recouvrement des thèmes sur plusieurs documents ainsi qu'une distribution des thèmes par document. La limite principale de cette approche est que chaque modèle produit est difficilement reproductible, variant à chaque nouvelle itération. De même, les résultats varient en fonction de la granularité choisie. L'identification des thèmes est à la charge de l'analyste. Celui-ci doit ainsi interpréter la liste des mots-clés remontés ainsi que la distribution des documents pour déterminer l'identité du thème. Si, dans l'exemple que nous fournissons en figure 4.7, le thème est relativement aisé à définir, cette tâche s'avère beaucoup moins intuitive sur la majeure partie des autres thèmes proposés.

L'approche se révèle donc assez limitée pour obtenir une représentation globale du domaine, bien que l'algorithme se soit révélé pertinent et efficace dans d'autres activités autour de corpus d'exigences [AAT10, HMM12]. Henss et al. signalent également la nécessité de découvrir empiriquement les bons paramètres pour l'apprentissage du modèle, mais cette dimension est moins importante puisque leur démarche n'est pas conservative et vise à la génération de FAQs.

Cette divergence de résultat peut s'expliquer par l'hétérogénéité des documents analysés. En effet, ceux-ci couvrent l'ensemble de la hiérarchie des documents, de la plus petite section, à la norme complète. Cela a pu introduire du bruit dans l'analyse. Le fait que les documents du corpus se recouvrent partiellement et présentent beaucoup de redondances de contenu a aussi pu avoir un effet sur les regroupements. Le fait que les modèles produits ne soient pas identiques d'une itération à l'autre empêche une analyse plus poussée de ces résultats.

4.4.4 Score TF-IDF pour la constitution de thèmes et la traçabilité

Nous présentons à présent l'analyse de notre dernière approche pour la question de la définition de la constitution de thèmes et la traçabilité. Cette approche est celle qui constituera la brique IncrementIndex de la contribution INCREMENT-IR. Les autres approches n'étant pas suffisamment pertinentes à nos yeux.

4.4.4.1 Intuition

Le score TF-IDF (Term Frequency – Inverse Document Frequency) est une mesure qui permet de mesurer l'adéquation d'un document indexé vis-à-vis d'une requête. Il s'agit du produit, parfois pondéré, entre le score TF qui est la fréquence du(es) terme(s) recherché(s) dans le document, et de l'importance du terme (IDF), c'est-à-dire le nombre de documents où ce terme apparait dans tout l'index. Ainsi, plus un terme est rare dans le corpus, plus les documents où il apparait sont importants pour ce terme. Ce score permet de classer les documents par ordre de pertinence d'un point de vue statistique.

Cette méthode est largement rencontrée dans la communauté académique pour les problèmes de traçabilité exigence vers code, un peu moins pour les questions de traçabilité entre exigences. La limite première de cette approche est quelle propose un grand nombre de résultats potentiels, appelés candidats, dont seule une partie permet d'établir des liens de traçabilité avérés. Elle ne permet donc pas d'établir automatiquement des liens de traçabilité.

Du point de vue de l'approche *Theme*, elle ne permet d'identifier ni les thèmes, ni leur signature. Cependant, à partir de ces derniers, elle permet de retrouver leurs traces.

4.4.4.2 Matériel de l'évaluation et opérations

Pour la mise en oeuvre de la démarche nous avons utilisé Lucene [5] comme moteur d'indexation et de recherche. Notre corpus à indexer étant constitué des 622 documents manuellement constitués à partir des 8 normes de notre étude.

Cette approche s'évalue traditionnellement à partir des valeurs de rappel et de précision relativement à un étalon. Nous rappelons ces formules :

$$\text{rappel} = \frac{\text{nombre de liens corrects retrouvés}}{\text{nombre de liens corrects}}$$

$$\text{précision} = \frac{\text{nombre de liens corrects retrouvés}}{\text{nombre de liens retrouvés}}$$

Nous n'avons pas d'ensemble étalon pour l'ensemble des recherches que nous avons menées. Sans étalon, il n'est pas de mesure de rappel et de précision possibles. Pour les besoins de l'expérimentation, nous avons construit un étalon constitué autour des thèmes remontés par l'algorithme de clustering Lingo. Cet étalon est très imparfait. En effet, nous ne pouvons affirmer l'exhaustivité de ce dernier. Néanmoins, il est constitué de manière systématique, ce qui limite le biais d'un étalon complètement manuel, et les

5. http://lucene.apache.org

résultats remontés, à défaut d'être exacts, permettent d'avoir une vue homogène (avec le même biais) sur l'ensemble des thèmes.

En l'absence de signatures complètes pour les thèmes nous avons considéré que les noms des thèmes faisant partie de la signature, joueraient également le rôle de signature.

La recherche d'information générant un grand nombre de candidats, nous avons établi de manière empirique une valeur de coupure sous laquelle les documents remontés sont considérés comme non pertinents et sont écartés de l'ensemble des candidats. Cette valeur de coupure a été établie à 0.25, ce qui est une valeur proche de celle commune dans la littérature (0.3) [CG11].

4.4.4.3 Variations de l'analyse

L'introduction dans l'index des sections et des sous-sections mène à proposer du contenu redondant. Contenu d'autant plus redondant que le document propose une structure importante en profondeur. Nous avons analysé les différentes normes à différents niveaux de granularité pour évaluer l'impact d'une telle redondance dans l'index. Pour ce faire, nous avons construit plusieurs index en complément du premier contenant toutes les documents.

Nous avons donc eu quatre index contenant respectivement :

- la totalité des documents du corpus d'analyse (622 documents),
- les normes entières (8 documents)
- les normes entières ainsi que leur découpage au premier niveau (107 documents)
- les normes entières, les découpages de premier et second niveau (275 documents)

.

Nous avons également mené des analyses sur l'impact de la racinisation par rapport à la correspondance parfaite dans nos index, ce qui nous a conduit à doubler les index puisque les index et les moteurs de recherche sont associés et exclusifs.

4.4.4.4 Résultats de l'approche par indexation et recherche d'information

Le tableau 4.4 présente un échantillon des sections de l'index remontés pour la recherche du thème *specification*. Les données du tableau représentent les noms des documents et correspondent aux identifiants des sections. Les documents sont triés par rapport à leur score TF-IDF et les sections colorées font parties des sections faisant parti de l'étalon créé pour l'occasion.

Nous pouvons observer que les sections sont plus ou moins pertinentes, indépendamment de leur position dans la structure de la norme. Par exemple, les sections 6.4.3.2 and 6.4.3.3 de la norme CEI 61513 (IEC61513) ont un score plus faible que leur parent (6.4.3) alors que celui-ci a un verbatim plus important même si les scores sont proches. Ce score est intéressant pour évaluer la contribution des sections au thème.

Nous observons également que les sections dont le titre ne contenait pas le nom du thème remontent parmi les candidats et se montrent également pertinents (par exemple, 62138-C5.5.3 qui contient, en outre, une exigence importante relative à la spécification, mais de manière indirecte puisqu'il ne s'agit pas de la préoccupation première. Ce n'est

pas surprenant puisque nous savons que les thèmes peuvent se superposer et sont disséminés à travers le corpus. En outre, nous observons que pour cette recherche particulière, nous écartons involontairement des documents pertinents du fait de leur faible score TF-IDF, ce qui abaisse le score de rappel.

TABLE 4.4 – Evaluation des scores TF-IDF

score	documents	Score	Documents	Score	Documents	Score	Documents	Score	Documents	Score	documents
0,4495	60987 C6.6	0,3893	60987 ZA	0,3372	61513 C6.4.2.2	0,3141	60880 ZF	0,2810	62566 C7.6	0,2602	60880 C8.2.3.1
0,4495	62138 C5.3.2	0,3769	60880 C6.4	0,3372	62138 C5.3.1	0,3141	62138 C6.3.1	0,2781	62138 C6.4.3	0,2602	62138 C5.3.3
0,4214	61513 C6.4.3	0,3769	62138 C5.5.3	0,3372	62138 C5.4.2	0,3078	62566 C6	0,2781	62138 C6.7	0,2602	62138 C6.3.3
0,4214	61513 C6.4.3.2	0,3554	62138 C5.4	0,3372	62138 C6.4.2	0,3062	62138 C6.4	0,2781	62566 C7.8	0,2529	61513 C6.4
0,4214	61513 C6.4.3.3	0,3554	62566 C6.1	0,3372	62566 C6.2	0,3039	62138 C5.3	0,2753	60880 C6.1	0,2513	60987 C5.1
0,4214	62138 C6.4.1	0,3441	62138 C6.3.2	0,3369	62138 C6.3	0,2973	61513 C6.4.2	0,2753	61513 C6.4.3.1	0,2513	62566 C6.5.2
0,4129	61513 C6.2.3.4	0,3406	61513 C6.2.3.1	0,3261	62566 C7.2	0,2973	62566 C7.2.1	0,2753	62138 C5.5	0,2433	60880 C15.3.1.2
0,4129	62138 C5.4.1	0,3406	62138 C5.4.3	0,3179	61513 C6.4.6.3	0,2920	61513 C6.2.4.2.1	0,2753	62138 C6.5.3	0,2433	60880 C6
0,3933	61513 C6.4.2.1	0,3372	61513 C5.3	0,3179	62566 C6.3	0,2920	62566 C7.2.2	0,2753	62566 C6.4	0,2433	60880 ZH
0,3933	62138 C6.3.4	0,3372	61513 C5.6.1	0,3179	62566 C7.4.1	0,2810	60880 C7.1.3	0,2753	62566 C6.5	0,2433	61513 C6.1

A partir de notre étalon, nous avons mesuré les scores de rappel et de précision pour 10 des 19 thèmes identifiés précédemment.Ces résultats sont présentés dans le tableau 4.5. En traçabilité des exigences, les approches sont évaluées généralement avec de bons scores de rappel et des scores de précision assez faibles [CHCGE10, NM12]. Dans notre cas, nous ne différons pas des scores de la littérature avec des mesures de rappel en moyenne de 86% (en correspondance parfaite) ou à 91% (avec racinisation)contre une précision variant entre 11 et 33% en dehors de deux exceptions notables pour *"safety life cycle"* et *"operation"* qui ont respectivement un rappel et une précision assez basse et un score de rappel bas.

En ce qui concerne l'impact de la redondance des contenus, nous avons lancé les mêmes requêtes sur les différents index et analysé les documents remontés. Nous n'avons pas observé de changement significatif dans l'ordre d'apparition des documents remontés, ce qui permet de supposer que l'impact de la redondance dans l'établissement des scores est limité. Ce résultat est à confirmer sur un index plus large en ajoutant de nouveaux documents.

Nous avons également évalué l'impact de la racinisation sur les index et mesuré les scores de rappel et de précision à partir de notre étalon. Nous présentons cette évaluation dans le tableau 4.5. Nous n'avons pas enregistré de différences majeures entre une correspondance totale et une recherche sur termes racinisés. Le seul cas particulier vient du thème *operation*, où la racinisation à permis de remonter tous les documents de l'étalon. Pour ce thème particulier, il est à noter que ce terme est particulièrement utilisé

sous des formes variées : *"operating", "operational", "operation"*, une caractéristique pour laquelle la racinisation des termes est essentielle.

TABLE 4.5 – Comparaison entre correspondance directe et racinisation

theme	answer set	matched	over threshold	miss or below threshold	recall	precision	matched with stemming	over threshold	miss or below threshold with stemming	recall with steming	precision
specification	15	279	97	3	0,80	0,12	367	108	2	0,87	0,12
common cause failure	15	222	44	4	0,73	0,25	263	55	2	0,87	0,24
system integration	15	421	84	0	1,00	0,18	495	135	0	1,00	0,11
pre-developed software	7	425	63	0	1,00	0,11	435	136	1	0,86	0,04
installation	9	79	27	0	1,00	0,33	90	35	0	1,00	0,26
maintenance	12	116	62	1	0,92	0,18	116	62	1	0,92	0,18
safety life cycle	5	347	58	2	0,60	0,05	347	60	2	0,60	0,05
operation	14	205	40	7	0,50	0,18	310	90	0	1,00	0,16
configuration management	6	206	52	0	1,00	0,12	222	52	0	1,00	0,12
quality assurance	7	191	64	0	1,00	0,11	196	64	0	1,00	0,11
	10,50	249,10	59,10	1,70	0,86	0,16	284,10	79,70	0,80	0,91	0,14

Cette distinction entre racinisation et correspondance parfaite a un intérêt particulier car la racinisation a tendance à remonter plus de documents candidats et donc de favoriser le score de rappel au détriment du score de précision. Or la grande taille des ensembles candidats à valider complique l'analyse et représente un enjeu important de la traçabilité des exigences basée sur la recherche d'information. Dans notre cas, l'avantage d'une racinisation n'est pas significatif et pourrait faire office d'heuristique pour la réduction de la taille des candidats.

4.5 Discussion et synthèse

Dans ce chapitre, nous avons abordé la question de la définition et la constitution de thèmes et de la traçabilité des exigences. Nous avons défini l'approche *Theme* autour de la définition des thèmes sous l'aspect d'un triptyque <nom, signature, traces>. Nous avons évalué quatre démarches, basées sur les statistiques, les algorithmes de clustering, les approches par apprentissage et enfin par score TF-IDF.

Toutes les approches ont montré des limites importantes quant à leur applicabilité dans notre contexte. Cependant, le score TF-IDF, qui est une des bases de la recherche d'information, offre une perspective intéressante pour la constitution d'ensembles de documents à partir de recherche dans l'index des documents. Cette solution est plus flexible que les approches de clustering qui demandent un nombre fixe de thèmes. Dans les deux cas, il faut un effort d'interprétation tout aussi important par la suite.

Dans la contribution globale INCREMENT, la contribution INCREMENT-IR peut être vue comme une seconde perspective en plus de la perspective modélisation. Cette perspective analyse documentaire est représentée dans la figure 4.8. Nous ne travaillons plus à l'échelle du modèle mais à l'échelle du corpus documentaire lui même. Le mécanisme d'indexation et de recherche ont donc été conservé et constituent la brique

IncrementIndex de la contribution INCREMENT-IR.

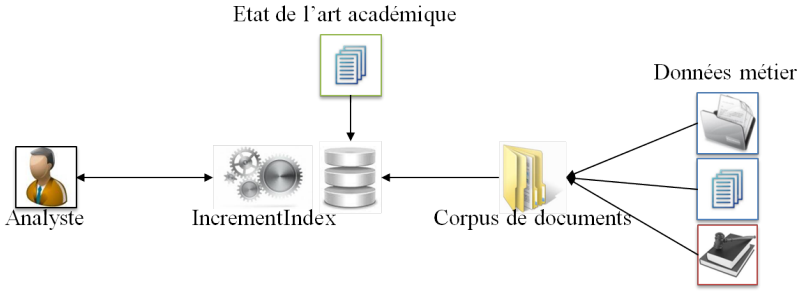


FIGURE 4.8 – INCREMENT-IR, une approche Recherche d'information

Nous discutons à présent de manière plus globale des approches et des limitations de nos analyses.

4.5.1 Discussion autour des approches

Les travaux en recherche d'information sont basés sur de larges ensembles de documents déjà constitués (base d'articles de journaux par exemple). Dans le cas de documents normatifs que l'on voudrait analyser à différents niveaux de granularité (à l'exigence près, a la section près, etc.), ce découpage est à opérer. De la même façon, l'acquisition de métadonnées sur ces documents n'est pas de facto.

Toutes ces approches, si elles montrent de bonnes prédispositions pour la traçabilité des exigences, ont plusieurs défauts par rapport au corpus de documents / d'exigences que nous manipulons. Elles nécessitent un grand nombre de documents pour produire des résultats intéressants. Il faut rappeler que ces techniques sont analysées sur des ensembles constitués de plusieurs années complètes d'archives d'articles de journaux et présentent donc une certaine homogénéité. Ce n'est pas forcément le cas ici.

Pour les algorithmes de clustering, si ceux-ci permettent une définition pertinente des thèmes, tous ne permettent pas des recouvrements alors que c'est une caractéristique importante des thèmes que nous recherchons sur le domaine du contrôle-commande nucléaire. De plus, s'ils permettent d'établir l'identité des thèmes et leur trace, ceux-ci ne permettent pas d'établir les signatures des thèmes, qui représentent un aspect important.

Pour les techniques d'apprentissage, un algorithme comme LDA est contraint par le nombre de regroupements à effectuer. Nous devons donc définir de manière empirique le nombre de regroupements à retrouver. Ces regroupements sont définis de manière itérative en calculant la meilleure distribution des termes du corpus, leur participation plus ou moins importante dans les n thèmes à constituer et la distribution de ces thèmes dans le corpus. Le résultat est donc dépendant du nombre d'itérations, du nombre de thèmes demandé. Les thèmes résultants ne sont pas forcément ceux auxquels on s'attend car la distribution est également plus syntaxique que sémantique et tient peu compte

du vocabulaire du domaine.

Le modèle appris d'un premier corpus est donc difficilement réutilisable dans le cadre de son extension (la proposition d'un échantillon pour l'apprentissage du modèle suivi du passage à l'échelle sur le corpus entier, ou alors sur un corpus progressivement étendu au fur et à mesure de l'ajout de nouveaux documents).

4.5.2 Discussion autour du corpus d'analyse

4.5.2.1 Taille et acquisition manuelle du corpus

Dans les analyses que nous avons présenté, nous nous sommes basés sur un petit corpus de données (622 documents). Les approches de recherche d'information s'évaluent souvent sur de grandes bases de documents, particulièrement préparées à des fins de benchmarking. La taille de notre corpus est donc relativement modeste bien que ce dernier soit déjà de taille conséquente pour une analyse manuelle.

Acquérir manuellement un tel corpus représente une tâche laborieuse. Acquérir l'ensemble des clauses et les définir finement de manière automatique représente un défi important. Cette démarche est également nécessaire pour la traçabilité et nécessite de descendre plus finement dans la structure des documents. Au moment de ces analyses, la brique IncrementParser n'était pas encore développée et elle n'influe pas sur la détermination des thèmes par rapport au corpus constitué.

4.5.2.2 Corpus de document à plat

Les documents réglementaires et les normes sont bien organisés, présentent une hiérarchie externe (entre document) et une structure interne (dans les documents) bien formée. On peut donc découper ces documents à différents degrés ou niveaux de granularité.

Cependant, les approches de recherche d'information ignorent la hiérarchie et la structure des documents. Ou alors, elles procèdent à une extraction préalable des éléments de manière à organiser leur domaine d'entrée.

Dans les deux cas, le corpus considéré est constitué d'un ensemble de documents mis à plat, avec autant d'entités indépendantes les unes des autres.

4.5.3 Discussion autour de la recherche d'information

4.5.3.1 Absence de thésaurus constitué

Ce problème a déjà été soulevé par Baniassad et Clarke dans l'approche *Theme/doc* [BC04]. Malheureusement s'il existe quelques outils déjà disponibles comme l'analyseur de Stanford pour le Part-Of-Speech Tagging (décomposition des phrases en mots, verbes, adjectifs, sujets, etc.), ou pour évaluer la similarité sémantique, ou encore la librairie Wordnet pour la synonymie, ceux-ci ne permettent pas d'adresser convenablement la spécificité du domaine. Si le domaine possède un vocabulaire précis et peu d'homonymes, les synonymes courants peuvent très bien se montrer de faux amis. Catégories de sûreté et classes de sûreté sont effectivement proches mais s'associent à deux entités

très différentes et font l'objet d'une distinction : les systèmes d'un côté et les fonctions de l'autre.

Aborder un tel domaine nécessiterait la mise en place d'une ontologie ad-hoc, pas simplement d'un glossaire, car il faut aussi pouvoir mesurer les relations entre les termes (liens avec les signatures de thèmes). Les compétences et le temps nécessaire à cette activité vont au delà du périmètre de la thèse.

4.5.3.2 Absence d'étalon et multiplicité des thèmes

L'absence d'un étalon exhaustif ne nous permet pas d'avoir une validation totale de notre analyse. La construction d'un étalon ad hoc est une opération courante [CHCGE10, SAY+12, dAFdS12, AAT10, LO10].

Dans notre contexte, un étalon est cependant quasiment impossible à construire. En effet, cela nécessite un effort manuel important de la part des experts, sur l'ensemble du corpus et sur l'ensemble des documents et l'ensemble des requêtes. A titre d'information, une telle opération a été menée dans le cadre du projet CONNEXION. Les experts ont rassemblé les exigences, issues de quatre normes, ayant un impact sur l'architecture du contrôle-commande. Entre mai et juillet 2012, en comptant 3 journées de réunions comptant entre 6 et 10 participants, on parvient à un effort d'un homme-mois sans compter les efforts préliminaires aux réunions. L'effort s'est poursuivi jusqu'en décembre.

Outre le fait que le périmètre était très largement réduit du fait que les exigences recherchées devaient avoir un impact sur l'architecture, les exigences remontées variaient énormément dans leur précision et leur granularité. De plus, les thèmes remontés par les experts divergeaient également de ceux que nous avions définis et validés auparavant. En effet ceux-ci concernaient des thèmes similaires aux nôtres mais également certains plus petits et plus précis, plus spécifiques que ceux que nous cherchions à constituer.

4.5.3.3 Taille des ensembles de candidats

Dans le monde académique, la réduction de cet espace de candidats est un sujet important qui n'a trouvé que des réponses partielles. La méthode la plus courante est d'utiliser une valeur palier en dessous de laquelle le score TF-IDF est considéré comme trop faible pour que le document soit pertinent comme candidat.

Cette méthode ne tient pas compte de la sémantique des termes. Dans nos expérimentations nous avons aussi mis en évidence qu'une telle approche pouvait supprimer des documents potentiellement intéressant et qui devraient être présentés à l'analyse. D'autres approches essayent de regrouper les documents en focntion de leur bonne ou mauvaise qualité pour proposer un premier filtre. Ceux-ci se basent cependant toujours sur une valeur de coupure [NM12].

Dans la partie suivante, nous adressons spécifiquement la question de la taille et de la composition du corpus pour l'indexation et présentons une approche qui hybride la modélisation et la recherche d'information dans un but d'améliorer la précision de cette dernière.

Chapitre 5

INCREMENT-Hybrid : une hybridation modélisation - Recherche d'information

5.1 Introduction : Mixer IDM et RI pour une approche globale sur le domaine

Dans le chapitre 3, nous avons abordé la question de la formalisation du domaine des exigences réglementaires de sûreté pour le contrôle-commande nucléaire. Nous avons établi un métamodèle qui permet l'acquisition et la capitalisation de la connaissance de ce domaine. ainsi que les briques outillées pour l'acquisition et la manipulation de modèles. Nous avons mis en évidence un certain nombre de limitations vis-à-vis des capacités d'analyse des modèles manipulés à cause de la nature textuelle de ses éléments.

Dans le chapitre 4, nous avons présenté nos travaux autour de l'analyse de normes internationales pour la traçabilité des exigences et l'organisation du corpus d'exigences autour de la notion de *topics*. Si les techniques de recherche d'information s'avèrent performantes à large échelle, et ont montré dans la littérature académique de bonnes dispositions pour mettre en œuvre / assister la traçabilité des exigences, elles souffrent de certains défauts qui limitent leur applicabilité, notamment liés au manque de sémantique des documents analysés, une fois découpés et indexés "à plat".

Dans le chapitre 5, nous présentons la troisième partie de la contribution INCREMENT, à savoir une approche d'hybridation entre ces deux mondes très différents. Cette contribution met l'accent sur les heuristiques issues de la modélisation afin d'améliorer la recherche d'information via l'exploitation des types et des attributs comme mécanisme de filtre. Ce mécanisme doit nous permettre de réduire des espaces de recherche. De ce point de vue la recherche d'information est vue comme une fonctionnalité d'analyse supplémentaire sur le modèle et vient agrémenter la base outillée que nous avons commencée à définir dans le chapitre 3.

Dans ce chapitre, nous discutons tout d'abord les bénéfices de l'hybridation (section 5.2. Nous présentons ensuite les challenges et les règles de mise en oeuvre de l'hybri-

dation (section 5.3). Enfin nous présentons une évaluation de cette approche vis-à-vis des questions liées à la recherche d'information par rapport à une approche standard comme celle que nous avons pu mettre en oeuvre dans le chapitre 4.

5.2 Bénéfices de l'hybridation

Dans le Chapitre 4, nous avons constitué manuellement un corpus d'expérimentation en découpant les documents à analyser jusqu'à la granularité de la plus petite section possible. Cette activité avait mené à la création manuelle de 622 documents pour l'indexation. Outre l'effort nécessaire, nous ne sommes pas allés jusqu'au grain le plus fin, c'est à dire l'exigence, mais sommes restés au niveau de la plus petite section.

Ce choix est autant pragmatique que raisonné. En effet, vu la nature particulièrement structuré des documents analysés, il n'était pas nécessaire d'aller à un grain plus fin que la section, puisque l'objectif était de retrouver à la fois des thèmes et des régions des documents les caractérisant. Cependant, il n'est ni raisonnable ni réaliste dès que l'on adresse un niveau plus fin d'abstraction et que l'on souhaite analyser au niveau exigence.

De la même façon, les entités manipulées sont très liées à la structure documentaire et ne portent pas de sémantique particulière. Avec une granularité plus fine, la sémantique des documents prend de l'importance, puisque se mélangent aussi bien du texte informatif, des définitions, des exigences, des recommandations, en plus des sections. Cependant dans un index contenant simplement les noms des documents et leur verbatim, la notion de type n'apparaît pas.

Or, ce découpage des documents est possible avec la brique IncrementParser que nous avons défini dans le chapitre 3. De même, ces informations de type (ainsi que l'ensemble des attributs liés aux objets) sont disponibles dans le modèle d'exigences que nous avons construit. Même si le découpage proposé reste au niveau du paragraphe, il permet d'acquérir la grande majorité des fragments des 8 normes CEI analysées dans le chapitre 4. Cette acquisition a conduit à la construction automatique de plus de 4000 documents (au sens indexation) lorsque basée sur le métamodèle knowledge et plus de 8000 documents pour l'indexation avec le métamodèle connexion, ce qui aurait été difficilement applicable manuellement.

Pour comparaison, le tableau 5.1 propose les informations liées à la capture des 8 normes qui servent pour le scénario d'évaluation de l'approche.

Notre corpus d'exigences ne contient pas que des exigences. Il s'agit d'un ensemble complet de documents contenant différents éléments et dont la validité n'est pas universelle. Par exemple, nous avons introduit les problématiques de classification dans le chapitre 1 comme un facteur de différences entre les pays cibles. A l'instar des normes IEEE 1012 (sur la V&V du logiciel) ou DO 178-B ou C, qui définissent des niveaux de sûreté, un ensemble d'exigences et une validité (pour la conformité à la norme) liée à ce niveau de sûreté, les normes du nucléaires définissent spécifiquement des exigences en fonction de la catégorie des fonctions à réaliser ou de la classification des systèmes. Cette classification est donc un facteur important pour la validité d'une exigence dans un contexte donné.

TABLE 5.1 – Acquisition des éléments pour la modélisation de 8 normes internationales

Norme	1ère publication	# pages	structure	# exigences	# Recommand.	Références	définitions	Documents indéxés
IEC60880-2006	1986	110	15 sections and 10 normative or informative annexes	308	92	8	43	939
IEC60987-2007	1989	30	13 sections and 3 informative annexes	53	17	10	18	219
IEC61226-2009	2009	32	7 sections and 1 informative annex	67	12	19	22	261
IEC61500-2009	1996	14	10 sections	43	10	10	8	136
IEC61513-2011	2001	98	8 sections and 5 informative annexes	238	48	27	62	1098
IEC62138-2004	2004	47	6 sections	180	48	2	36	555
IEC62340-2007	2007	22	9 sections + 1 informative annex	46	4	11	26	226
IEC62566-2011	2011	52	17 sections + 2 informative annexes	243	33	7	14	646
totals		405	107 1st level structures	1178	264	94	229	4080

Comme nous manipulons les normes ou les documents dans leur globalité avec un haut niveau d'abstraction, nous ne maîtrisons pas la volumétrie des éléments pour les analyses. Avoir des informations supplémentaires comme la classification, ou alors les thématiques rapportées aux exigences, permet d'obtenir un mécanisme de filtre intéressant pour réduire les espaces de recherche et obtenir une recherche d'information plus "intelligente".

En résumé, les bénéfices d'une hybridation sont liées à l'utilisation et aux fonctionnalités propres de chaque domaine à savoir :

- le modèle pour capitaliser et manipuler les données,
- la recherche d'information comme outil pour travailler du texte à large échelle, tout en ayant de multiples index pour des recherches ciblées,
- le typage et l'utilisation des attributs des éléments du modèle comme un filtre à la recherche.

5.3 Mise en œuvre de l'hybridation

5.3.1 Challenges liés à l'hybridation

Un des challenges pour une hybridation est lié à la différence de représentation concrète des concepts d'un modèle et d'un index. D'un point de vue théorique, un modèle est un ensemble de classes et d'attributs, définis par son métamodèle. Un index a une structure beaucoup plus simple, représentée dans la figure 5.1. Un index contient des documents qui contiennent des champs. Il y a donc une première question à laquelle nous répondrons dans la section suivante, liée à la transposition des concepts.

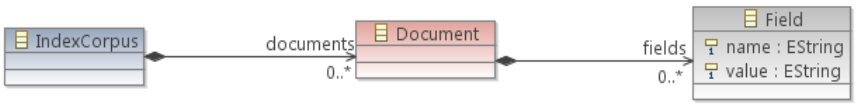


FIGURE 5.1 – Métamodèle d'un index

Un modèle d'un point de vue général, est décrit dans un fichier xml ou est géré directement en mémoire sous la forme d'objets. Contrairement au métamodèle présenté, un index est constitué concrètement d'un certain nombre de tables de hachage avec recherche inversée reprenant chaque terme de l'index et capitalisant leurs fréquences, positions, liens vers les documents, nombre d'occurrences, fréquence d'apparitions, éventuellement mots voisins, etc. Cette différence entre les concepts d'un index et sa représentation concrète est liée aux questions d'efficacité pour la recherche.

Il n'est donc pas possible d'effectuer de recherche d'information à même le modèle. Le passage à une indexation systématique des éléments du modèle rend cependant ces analyses possibles.

Le challenge principal est donc de maintenir en cohérence les deux représentations concrètes (a contrario de leur représentation abstraite : élément de modèle, document,

champ, etc.) pour pouvoir les utiliser. Ainsi, un ajout/suppression/édition dans le modèle doit se traduire dans l'index par un ajout ou une suppression dans l'index afin de conserver la synchronisation entre les deux représentations.

5.3.2 Correspondance entre le modèle et l'index

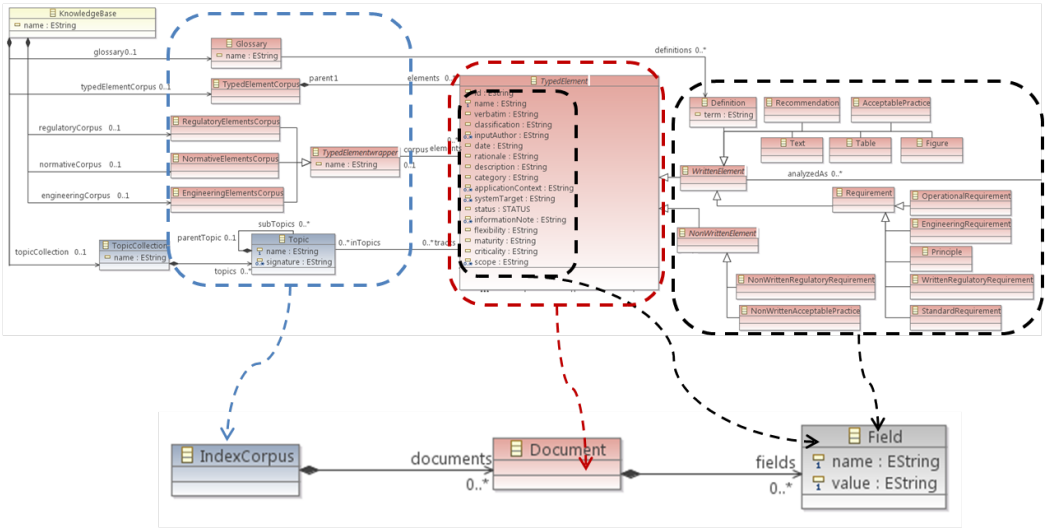


FIGURE 5.2 – Correspondance entre éléments de modèle et éléments d'indexation

La figure 5.2 présente la vue logique entre les deux représentations ainsi que la synchronisation des concepts entre le métamodèle d'exigence et un index.

Les concepts d'index et de corpus se correspondent naturellement. Tout comme nous mettons à disposition un certain nombre de collections additionnelles pour regrouper les exigences, il est possible de créer autant d'index représentatifs. Toutes les collections ne sont cependant pas pertinentes ou intéressante pour l'indexation. En effet, il est plus intéressant d'indexer chaque *Topic* séparément comme autant d'index contenant des documents (les *tracks* des *Topics*) que la collection de topics elle-même.

La notion de *TypedElement* ou de *DocumentFragment* se retrouvent dans la notion de document. Les deux notions sont au cœur du principe d'index et de celui de notre métamodèle. Cependant si le typage est une notion importante du métamodèle, celui-ci n'est plus une information de première classe accessible directement dans l'index.

Dans un index, toute la richesse des concepts du métamodèle se retrouve dans les champs du document. Ainsi un document représentant une norme avec un titre, un éditeur, une date de publication, et un certain contenu doit se retrouver indexé sous la forme d'un document avec son contenu dans un champ particulier, toutes

les propriétés dans autant de champs (titre, éditeur, date de publication, etc.) mais également un champ représentant le typage du document. Tout comme le verbatim d'une exigence ou d'un fragment de texte se retrouve sous la forme d'un simple attribut très peu différent des autres d'un point de vue objet, celui-ci se retrouve dans un champ du document, au même niveau que les autres champs.

5.3.3 Principes de la synchronisation entre un modèle et un index

5.3.3.1 Volatilité des champs des documents

Dans un index, un document est libre quant à la définition et le contenu de ses champs. Ainsi, les documents peuvent ne pas partager les mêmes champs ni le même type d'information pour des champs portant le même nom. Ceci entre en contradiction avec les principes de typage présents dans les modèles et pourrait s'avérer problématique lors d'un passage de l'index au modèle.

Dans notre approche, nous nous basons sur les éléments du métamodèle pour déterminer les champs du document et jamais l'inverse. La définition d'un élément du modèle conduit à la définition d'un document qu'on va ajouter à un ou plusieurs index. Ainsi, nous maintenons la cohérence de l'index vis-à-vis du modèle en forçant la définition et les valeurs des champs du document.

5.3.3.2 Implémentation de la synchronisation

A cause de la volatilité des champs des documents et de la structure à plat et moins riche des index, il n'est pas raisonnable de proposer un mécanisme pour la création d'un document à indexer et créer son sosie dans le modèle. En effet, il faudrait contraindre fortement la constitution des champs d'un document pour permettre la définition, dans le modèle, des éléments dont le type soit cohérent, ayant le bon parent et qui ne viole aucune contrainte statique de bonne formation.

Les mécanismes de synchronisation ont donc été ajoutés aux méthodes opérant sur les modèles et se limitent à ce sens modèle -> index avec le modèle comme élément représentatif de la connaissance à analyser, l'index étant un média tiers venant s'adosser au modèle.

Si on considère le patron CRUD (Create, Read, Update, Delete) pour l'ajout, la lecture, l'édition et la suppression de données, la synchronisation est mise en œuvre de la manière suivante :

- Ajout : l'ajout d'un élément de modèle implique l'ajout d'un document dans l'index. Pour les fragments de documents, la structure composite telle qu'elle existe dans le document (au sens des sections, sous-sections) est perdue dans l'index qui met l'ensemble des documents à plat. Cependant, à la différence d'un document textuel, le verbatim d'une section dans un modèle contiendra la section entière en plus de sa structure composite avec chaque élément de modèle individuel contenant un fragment de ce verbatim. De la même façon, un document représentant une section contiendra le verbatim de la section entière. Chaque document contenant les clauses individuelles est géré de manière indépendante par rapport à sa

section ou son document parent.
- Lecture : la lecture d'un document n'a aucun impact sur le modèle ou l'index
- L'édition d'un élément de modèle peut impacter l'organisation du modèle ; les écarts avec l'index sont variables. Dans Lucene, il n'y a pas de mécanisme d'édition, il faut supprimer le document et ajouter un nouveau document contenant les mises à jour. Si le document était également un élément de structure, il y a aussi un impact avec les documents liés à cet élément. Cependant, la moindre modification d'un élément du modèle entraîne une réindexation du document (suppression et ajout), ce qui peut être pénalisant surtout si on considère que nombre d'entrée sur le modèle sont faites manuellement.
- La suppression d'un élément de modèle impacte l'ensemble des éléments liés. Dans le cas d'une section, il faut supprimer tous les éléments / documents qui appartiennent ainsi que les références vers la section et ses enfants.

Du fait de la limitation de Lucene vis-à-vis du mécanisme d'édition, deux alternatives d'indexation sont proposées. Une indexation, dite passive, qui crée un nouvel index à partir d'un modèle en mémoire en parcourant ce dernier. Une autre alternative, dite active, crée les documents à la volée durant l'acquisition ou effectue la séquence suppression-ajout d'un document dans le cas de l'édition.

Nous présentons le mécanisme d'indexation passif, c'est-à-dire, à partir d'un modèle complet en mémoire. La figure 5.3 présente la transformation générique d'un élément du modèle vers un document Lucene. En particulier, nous encodons le type de l'objet dans un champ "type" et encodons systématiquement tous les attributs de l'objet dans des champs distincts. Certains de ces attributs deviennent des champs à analyser avec les traitements adéquats (suppression des mots vides, éventuellement racinisation), comme les verbatims, tandis que d'autres sont considérer comme des champs dont la valeur ne doit pas être analysée, comme par exemple les champs ayant valeur d'Ids. Il est à noter la gestion particulière des collections qui sont transformées par une redondance de champs portant le même nom mais avec autant de valeurs différentes que dans la collection.

Les figures 5.4 et 5.5 présentent le mécanisme de parcours du corpus d'éléments typés et du corpus de documents avec une particularité pour les documents puisqu'ils possèdent une structure composite (5.5). Nous avons retiré du listing les différentes initialisations et configurations pour les index. Dans le programme, nous effectuons une double indexation. Nous utilisons, à des fins de comparaison de requête, un index dit standard, qui analyse la langue anglaise mais n'effectue pas de racinisation. En parallèle, nous utilisons un second index (englishAnalyzer) qui analyse les documents avec un traitement de racinisation. Il faut rappeler que les phases d'indexation et d'analyse doivent partager le même moteur d'indexation. Il faut donc autant d'index que de moteurs d'analyse souhaités.

En ce qui concerne l'indexation dynamique, il s'agit d'un appel à la fonction buildDocument, une fois terminée la création de l'objet ou de sa mise à jour (avec une suppression du document en plus).

```
/**
  * Generic transformation from a Connexion ModelElement to Lucene Document
  * @param o the object to map into a lucene document
  * @return the Lucene Document with filled fields related to o attributes
  */
@SuppressWarnings("rawtypes")
public Document buildDocument(EObject o) {
  Document doc = new Document();
  List<String> types = getSupertypes(o);
  for (String t : types) {
  doc.add(new Field("type", t,
    Field.Store.YES, Field.Index.NOT_ANALYZED));
  }
  for (EStructuralFeature attr : o.eClass().getEAllAttributes()) {
    String name = attr.getName();
    String value = "";
    if (o.eGet(attr) != null) {
      if (name.equals("verbatim") || name.equals("description")) {
        value = o.eGet(attr).toString();
        doc.add(new Field(name, value,
          Field.Store.YES, Field.Index.ANALYZED));
        } else {
          if (attr.getUpperBound() == -1) {
            Object valList = o.eGet(attr);
            for (Object val : (List) valList) {
              if (val != null) {
                value = val.toString();
                doc.add(new Field(name, value,
                  Field.Store.YES, Field.Index.NOT_ANALYZED));
                        }
                      }
                } else {
                  value = o.eGet(attr).toString();
                  doc.add(new Field(name, value,
                    Field.Store.YES, Field.Index.NOT_ANALYZED));
                  }
                }
          }
  }
  return doc;
}
```

FIGURE 5.3 – Transformation des éléments de modèle en document d'index

```
/**
 * load a requirement model conforming to the Connexion Metamodel
 * and index each TypedElement and DocumentFragment into an index
 * @param modelLocation location of the model to index
 * @param indexLocation location of the directory to put the index
 * @throws IOException
 */
public void indexFromModel(String modelLocation, String indexLocation)
  throws IOException{
  //loading the model
  KnowledgeBase knowledge = load(modelLocation);
  // initializing the indexing process
        ...
        ...
  // getting all models TypedElements and Documents
  EList<connexion.TypedElement> tes = knowledge.getTypedElementCorpus()

  EList<connexion.Document> docs = knowledge.getCorpus()

  // indexing TypedElements
  for (TypedElement te : tes){
    Document doc = buildDocument(te);
    standardWriter.addDocument(doc);
    englishWriter.addDocument(doc);
  }
  // indexing Corpus Documents
  for (connexion.Document d : docs){
    Document doc = buildDocument(d);
    standardWriter.addDocument(doc);
    englishWriter.addDocument(doc);
    for(DocumentModelElement frags : d.getFragments()){
      buildFragments(d, standardWriter, englishWriter);
    }
  }
  standardWriter.close();
  englishWriter.close();
}
```

FIGURE 5.4 – Mécanisme d'indexation passif à partir d'un modèle à charger

```
/**
 * Recursive indexing of the Composite document structure
 * @param d DocumentModelElement to index
 * @param standardWriter index with StandardAnalyzer
 * @param englishWriter index with EnglishAnalyzer
 * @throws IOException
 */
public void buildFragments(DocumentModelElement d,
  IndexWriter standardWriter,
  IndexWriter englishWriter) throws IOException {
  if (d instanceof Section){
    Document doc = buildDocument(d);
    standardWriter.addDocument(doc);
    englishWriter.addDocument(doc);
    for (DocumentFragment frags : ((Section) d).getFragments()){
      buildFragments(frags, standardWriter, englishWriter);
    }
  } else {
    Document doc = buildDocument(d);
    standardWriter.addDocument(doc);
    englishWriter.addDocument(doc);
  }
}
```

FIGURE 5.5 – Indexation des différents *DocumentFragment*

5.4 Evaluation d'une mise en oeuvre hybride

5.4.1 Objectifs de l'évaluation

En traçabilité des exigences, les approches favorisent le score de rappel au détriment de la précision. Concrètement, cela signifie que parmi la liste des documents remontés figure un grand nombre de faux positifs. Cet effet est d'autant plus important sur des corpus de grande taille.

Pour palier à la question de la taille de l'ensemble des candidats, les approches de recherche d'information reposent souvent sur une valeur de coupure, calculée empiriquement, en dessous de laquelle on considère que les documents remontés ont un score de similarité trop bas. Cette démarche pose deux limites. Premièrement, celle-ci impose de calculer empiriquement cette valeur. Deuxièmement, une fois passé l'établissement de cette valeur de coupure, il est possible que des documents pertinents soient retirés de manière arbitraire de la liste des candidats pour analyse, ce qui est inacceptable du point de vue de la traçabilité.

Notre démarche vise à réduire la taille des ensembles de candidats sans avoir recours à une valeur de coupure. L'objectif de cette analyse est de mesurer la réduction de cet espace à travers notre approche hybride et de la comparer à une approche standard. Notre évaluation porte donc sur la base du même corpus de documents indexés, mais dont l'un des index possède les informations issues du modèle et en particulier les informations de typage liées au document. L'objectif ici n'est pas de discuter la validité des documents comme étant pertinents ou non par rapport à la requête initiale.

Comme nous l'indiquions précédemment, nos index ne sont pas constitués que d'exigences, ils contiennent beaucoup de données supplémentaires qui, pour l'occasion, génèrent un certain "bruit". Ce bruit est relatif car les documents indexés sont importants vis-à-vis de la capitalisation de la connaissance. Cependant, les exigences ne sont pas à appliquer à chaque instant mais dépendent de plusieurs paramètres, comme la classification ou l'activité d'ingénierie en cours. De fait, les données indexées sont moins maitrisées que pour les approches classiques de traçabilité à partir de recherche d'information [CHCGE10, LO10, CG11].

5.4.2 Méthodologie pour la comparaison approche hybride - approche standard

Dans cette section, nous abordons les expérimentations menées pour évaluer la pertinence d'une telle hybridation. Nous présentons l'approche que nous avons suivie sous la forme du patron MISC (pour Modeling-Indexing-Searching-Comparing). Nous présentons la méthode MISC dans la figure 5.6

Dans un premier temps, le corpus est acquis de manière systématique via la brique IncrementParser et nous obtenons un modèle contenant 8 normes et leurs éléments.

Dans un premier index, nous indexons les documents issus du modèle avec la seule information issue des verbatims textuels et du nom des éléments pour leur identification. Dans un second index, nous ajoutons aux documents la notion de type issue du

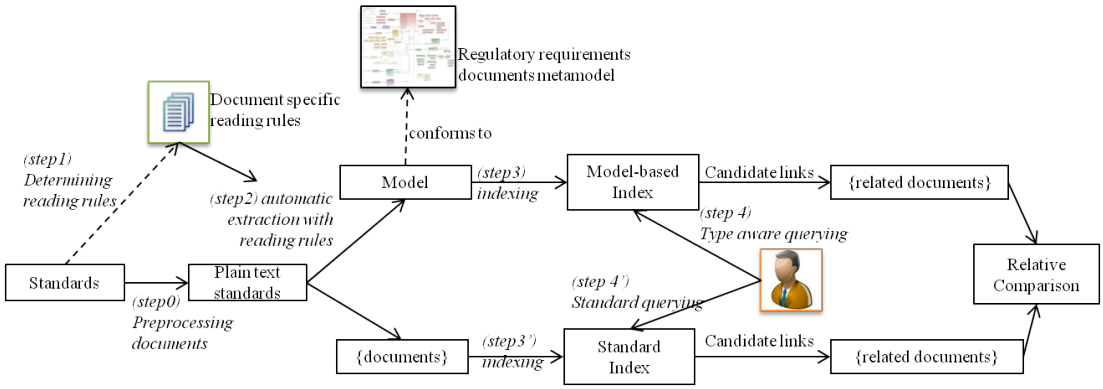


FIGURE 5.6 – Evaluation de l'approche Hybride selon le pattern MISC

modèle. Nous avons alors deux index, un index dit "classique" et un index "basé sur le modèle". Les index sont construits selon les mêmes étapes de pré-traitements et les mêmes moteurs, comme décrit dans le chapitre 4.

Nous effectuons un certain nombre de requêtes sur les deux index que nous présentons dans les tableaux 5.3 et 5.4. Dans l'index "classique", nous remontons tous les documents au dessus d'une valeur de coupure que nous avons déterminée de manière empirique à 0.2 (au lieu des 0.3 classiquement utilisés dans la littérature [CG11]). Dans l'index "modèle", nous ajoutons une heuristique de recherche qui est de se limiter aux exigences et aux recommandations contenues dans l'index. Concrètement, cela signifie que nous allons également rechercher dans les documents dont le champs type a pour valeur *Requirement* ou *Recommendation*. Pour cette recherche, nous n'utilisons pas de valeur de coupure et considérons la totalité des documents remontés.

Enfin nous effectuons une comparaison entre les deux ensembles de documents remontés et mesurons l'importance relative du bruit dans l'index et la réduction de l'espace de candidats si on enlève ce bruit.

5.4.3 Evaluation de la réduction de l'espace des documents candidats à travers l'indexation des éléments du modèle

5.4.3.1 Evaluation de la part de bruit dans les documents

Le tableau 5.2 propose un premier résultat dans l'analyse des éléments de modèles créés et, parmi ceux-là, le nombre d'exigences et de recommandations dans le modèle. Les normes acquises varient en ce qui concerne la proportion d'exigences qu'elles contiennent. Cette proportion est même faible. Si on ne tient pas compte des fragments de texte (et donc de la partie corpus documentaire au sens modèle) la proportion d'exigences et de recommandations dans les documents correspond à 35% du corpus puisque seuls 1142 fragments sur les 4080 générés sont analysés comme des exigences.

Cette observation confirme la possibilité, dans notre cas, d'améliorer de manière significative la recherche d'information dans le corpus en utilisant les informations de

TABLE 5.2 – Acquisition des éléments de 8 normes internationales pour l'indexation

standard	1st year of publication	number of pages (English part)	structure	corresponding indexed documents	external normative references	definitions	statements requirements ("shall"), recommendations ("should")
IEC60880-2006	1986	110	15 sections and 10 normative or informative annexes	966	8	43	389 (305 / 84)
IEC60987-2007	1989	30	13 sections and 3 informative annexes	219	10	18	70 (53 / 17)
IEC61226-2009	2009	32	7 sections and 1 informative annex	261	19	22	79 (67 / 12)
IEC61500-2009	1996	14	10 sections	136	10	8	53 (43 / 10)
IEC61513-2011	2001	98	8 sections and 5 informative annexes	1102	27	62	286 (238 / 48)
IEC62138-2004	2004	47	6 sections	555	2	36	220 (180 / 40)
IEC62340-2007	2007	22	9 sections + 1 informative annex	226	11	26	50 (46 / 4)
IEC62566-2011	2011	52	17 sections + 2 informative annexes	646	7	14	276 (243 / 33)
totals		405	107 1st level structures	4111	94	229	1587 (1339 / 248)

typage. En supposant une extraction correcte, on peut déjà extraire de ce corpus de huit normes un sous-ensemble plus petit qu'un analyste humain peut traiter plus facilement que s'ils les manipulait comme un tout.

Cette mesure a été effectuée sur huit normes internationales, du même éditeur. Cependant, ces huit normes ne couvrent que les seuls aspects logiciels pour le contrôle-commande nucléaire dans la pratique française. Une analyse supplémentaire d'autres documents issus de réglementations différentes ou de type de réglementation différentes pourrait permettre d'affiner cette mesure.

Dit autrement, cette approche est pertinente si les données écartées pour l'étude, et que nous avons définies comme du bruit, peuvent être remontées en résultat de la requête et forme effectivement du bruit.

5.4.3.2 Comparaison et analyse qualitative des espaces de candidats proposés

La première évaluation concerne l'impact de la valeur de coupure sur la localisation du bruit à base d'un ensemble de requêtes simples correspondant à des préoccupations courantes du domaine. Nous la présentons dans le tableau 5.3. Si la valeur de coupure réalise la réduction de l'espace, celle-ci est relativement inefficace car l'ensemble des liens au dessus du seuil contient entre 62 et 90% de liens qui ne sont pas typés par la suite comme des exigences ou des recommandations. Non seulement, le bruit est remonté dans les résultats de la requête, mais en plus celui-ci représente une valeur significative des liens candidats, avant même d'évaluer la pertinence des liens restants.

La deuxième évaluation concerne l'impact de l'apport des informations de typage sur les mêmes requêtes et sur l'index "modèle". Nous présentons les résultats dans le tableau 5.4. Comme attendu, puisqu'il s'agissait d'une propriété de la requête, l'ensemble remonté ne contient que les éléments identifiés comme exigence ou comme recommandation.

TABLE 5.3 – Analyse des documents remontés pour un index avec une approche standard et valeur de coupure

query	retrieved links in the standard index	candidate links (above the threshold)	retrieved requirements and recommendations above the threshold	unconsistent fragments remaining	% noise remaining
configuration management	438	221	72	149	67,42
common cause failure CCF	602	154	17	137	88,96
specification	668	576	216	360	62,50
independence	102	64	14	50	78,13
validation	404	347	96	251	72,33
verification	555	445	169	276	62,02
quality assurance	421	259	84	175	67,57
defence in depth	141	81	8	73	90,12
integration	280	237	65	172	72,57
self supervision	125	92	25	67	72,83
modification	271	214	70	144	67,29
diversity	114	103	16	87	84,47
isolation	53	38	8	30	78,95

TABLE 5.4 – Analyse des documents remontés pour un index avec prise en compte des informations de typage

query	retrieved links in the model-based index	retrieved requirements and recommendation in the model-based index	Requirements and Recommendations above the threshold in the model-based index	Requirements and Recommendations below the threshold
configuration management	106	106	72	34
common cause failure CCF	115	115	17	98
specification	216	216	216	0
independence	14	14	14	0
validation	96	96	96	0
verification	171	171	169	2
quality assurance	106	106	84	22
defence in depth	14	14	8	6
integration	65	65	65	0
self supervision	25	25	25	0
modification	70	70	70	0
diversity	16	16	16	0
isolation	8	8	8	0

La taille des ensembles de documents remontés est globalement plus petite que pour une approche standard à base de valeur de coupure. En dehors d'un cas où la réduction de l'espace de recherche est faible, une requête sur les défaillances de cause commune (*Common cause failure CCF*) et une réduction de 25%, l'approche sur la base d'un index augmenté des informations de typage montre une réduction de l'espace des candidats. Cette réduction est en moyenne de 67% avec des écarts entre 52 et 84%. Cette réduction est significative pour des ensembles contenant plusieurs dizaines, voire centaines, de documents. Ce résultat est à mettre en rapport avec la proportion d'éléments assimilés à du bruit dans la première évaluation.

Outre la réduction du volume de documents remontés il est intéressant d'analyser la typologie des éléments conservés ou écartés dans les deux cas. Nous présentons une comparaison des deux approches dans le tableau 5.5.

TABLE 5.5 – Comparaison des deux approches

query	candidate links (above the cut-off value)	retrieved requirements and recommendations above the threshold	retrieved links in the model-based index	straightforward reduction of the research space (%)	Loss of requirements and recommendations
configuration management	221	72	106	52,04	34
common cause failure CCF	154	17	115	25,32	98
specification	576	216	216	62,50	0
independence	64	14	14	78,13	0
validation	347	96	96	72,33	0
verification	445	169	171	61,57	2
quality assurance	259	84	106	59,07	22
defence in depth	81	8	14	82,72	6
integration	237	65	65	72,57	0
self supervision	92	25	25	72,83	0
modification	214	70	70	67,29	0
diversity	103	16	16	84,47	0
isolation	38	8	8	78,95	0

Dans l'approche standard, le sous-ensemble des documents écartés, car ayant un score inférieur à la valeur seuil, contient des éléments typés comme des exigences et comme des recommandations. Même si nous n'évaluons pas la pertinence effective du document vis-à-vis de la requête, l'usage de la valeur de coupure aura amputé l'ensemble des liens candidats d'un certain nombre de documents potentiellement plus intéressants à analyser que nombre de documents ayant été typés comme du bruit et qui ont été conservés. Cette observation illustre la limitation principale d'une approche standard et démontre l'intérêt d'utiliser les mécanismes de filtres proposés par l'utilisation des informations fournies par le modèle d'exigences.

5.5 Discussion

A travers le mécanisme d'hybridation que nous avons présenté, nous avons ajouté une nouvelle fonctionnalité autour du métamodèle *Connexion*. Ce mécanisme permet de synchroniser un modèle Connexion et un index contenant les informations du modèle sous une représentation différente mais plus efficace pour la recherche d'information.

La contribution INCREMENT-Hybrid, illustrée dans la figure 5.7 est donc une hybridation entre les aspects modélisation et recherche d'information. Au niveau des briques, cela se traduit par la prise en compte de l'indexation dans l'acquisition de modèles via la brique IncrementParser ainsi que l'extension avec les facilités de typage de la brique IncrementIndex.

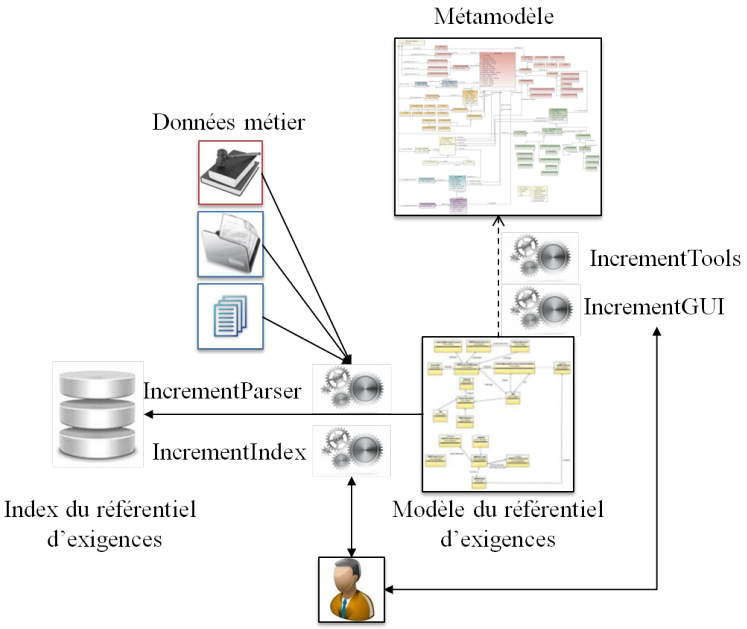


FIGURE 5.7 – Hybridation modélisation et indexation pour l'analyse de modèles d'exigences

5.5.1 Synthèse de l'évaluation

En conclusion de l'évaluation, nous pouvons émettre les observations suivantes :

- La plupart des documents étant des exigences ou des recommandations qui ont été remonté par les deux approches ont un score au dessus de la valeur seuil, sauf dans un cas (Common cause failure) qui en écarte un nombre très important. Notre approche hybride est capable de remonter des documents potentiellement intéressants mais qui seraient écartés car ayant un score trop faible.
- Considérant la réduction de l'espace des candidats, et l'élimination des documents

définis comme du bruit, notre approche est viable dans l'optique d'offrir à une analyse manuelle un ensemble de documents beaucoup plus petit et raisonnablement plus pertinent que pour une approche standard.
– Notre approche est donc conservative, comparée à l'approche standard. Non seulement nous retrouvons l'ensemble des documents typés comme exigences ou recommandations mais en plus, nous retrouvons parfois un nombre d'exigences et de recommandation plus important que pour l'approche standard.
– Nous proposons une heuristique supplémentaire pour la réduction des espaces de candidats, en plus des valeur de coupure [CG11], des clustering de documents [NM12]), etc.

5.5.2 Discussion autour de l'évaluation et de la contribution INCREMENT-Hybrid

5.5.2.1 Absence d'un ensemble d'étalons

La première limite de l'évaluation vient de l'absence d'étalon (*gold standard*) pour mesurer effectivement la pertinence des liens. Cependant, cet étalon n'existe pas et la constitution d'un tel étalon, pour le domaine du contrôle-commande numérique nucléaire, requiert un effort considérable. Les partenaires du projet CONNEXION ont mené une opération allant dans ce sens en identifiant les exigences réglementaires ayant un impact sur l'architecture du contrôle commande dans un corpus légèrement différent du nôtre. Non seulement leur périmètre d'étude était différent du nôtre, mais en plus, cette activité a requis un effort important pour des résultats parcellaires et une acquisition des exigences hétérogène (la granularité d'acquisition variait de l'élément de liste à la section entière). De plus, pour un ensemble de requêtes, cela nécessiterait un ensemble d'étalons.

De fait, nous nous sommes plus intéressés à adresser un des problèmes majeurs de l'analyse par recherche d'information à savoir la taille de l'espace de recherche et nous avons proposé une heuristique avec une approche utilisant le modèle d'exigences comme corpus à indexer plutôt que la seule décomposition des documents.

Pour rappel, les mesures de rappel et de précision se mesurent de la manière suivante :

$$\text{rappel} = \frac{\text{nombre de liens corrects retrouvés}}{\text{nombre de liens corrects}}$$

$$\text{précision} = \frac{\text{nombre de liens corrects retrouvés}}{\text{nombre de liens retrouvés}}$$

Le rappel représente donc la part de liens corrects retrouvés tandis que la précision évalue la taille de l'ensemble qu'il aura fallu obtenir pour obtenir ces liens corrects.

Le fait de retirer ce bruit d'un ensemble n'a pas de conséquence sur la mesure de rappel. Ce qui signifie que nous sommes à rappel constant par rapport à une approche standard. En émettant l'hypothèse que les documents identifiés comme du bruit et retirés des espaces de candidats le sont effectivement, alors notre approche offre un mécanisme d'amélioration de la précision, par construction puisque le nombre de liens corrects retrouvés n'a pas varié mais que le nombre de liens retrouvés a diminué.

5.5.2.2 Evaluation empirique de la valeur de coupure

Nous avons établi de manière empirique une valeur de coupure à 0.2, score TF-IDF en dessous duquel nous considérons les documents remontés non pertinents pour les étapes suivantes d'analyse. Cette valeur est plus basse que la valeur traditionnelle de 0.3 employée (comme le mentionne [NM12]). Cette valeur a pour effet d'augmenter le nombre de documents remontés, ce qui peut-être présenté comme un désavantage pour l'approche. L'évaluation du nombre "d'exigences" remontées et écartées amène à penser que le compromis avec un seuil à 0.2 est raisonnable puisque nous n'avons écarté que peu "d'exigences". Pour toutes les obtenir, nous aurions du réduire cette valeur, ce qui aurait pour effet d'augmenter le nombre de documents remontés et, par effet de bord, le "bruit" remonté.

5.5.2.3 Evaluation par rapport aux différents métamodèles

Pour l'évaluation présentée, l'acquisition s'est opérée pour l'instanciation d'un modèle conforme au métamodèle *Knowledge*, et non *Connexion*. Une des raisons est l'aspect chronologique de cette évaluation qui intervient avant l'apparition et la stabilité de ce dernier métamodèle. Fondamentalement, les différences majeures entre l'évaluation sur ces deux métamodèles proviendraient du nombre de documents à indexer beaucoup plus important avec le métamodèle *Connexion* du fait de la séparation entre les documents (et les fragments textuels) et les éléments typés qui leur sont associés. La structure du métamodèle *Knowledge* est beaucoup plus proche de l'approche standard que celle du métamodèle Connexion.

De plus, il n'est pas pertinent d'indexer deux fois les mêmes documents pour l'approche standard en indexant à la fois les *TextFragment* et les *TypedElement* qui leur son associés alors qu'ils partagent le même verbatim, avec pour exception les sections. De ce point de vue, l'évaluation et la comparaison s'effectuent sur deux index cohérents et similaires en taille et en éléments.

5.5.2.4 Evaluation de la granularité d'acquisition, du typage des éléments et du polymorphisme

Nous avons utilisé la brique IncrementParser pour faire l'acquisition du modèle. Basé sur ce modèle nous avons généré les documents pour l'indexation standard d'un côté et utilisé le modèle pour l'indexation basée modèle. Si la brique présente la limitation d'être au grain du paragraphe, les deux index sont cohérents l'un par rapport à l'autre en terme de nombre de documents indexés et de leur champ verbatim.

La question du polymorhpisme ne se pose pas pour l'évaluation que nous avons menée en nous basant sur le métamodèle *Knowledge*. Elle se pose cependant sur le métamodèle *Connexion* où les exigences écrites et non écrites sont instanciables et réifiables. De même la nature d'une exigence écrite (*Requirement* dans le métamodèle) peut être normative (issue d'une norme imposée par le régulateur), réglementaire (au sens issue d'un texte réglementaire) ou issue de l'ingénierie. Pour l'indexation, il faut

également prendre en charge dans le champ *type* l'arbre d'héritage pour tenir compte du polymorphisme lors de la recherche.

5.5.2.5 Ouverture à d'autres heuristiques que le typage et multi-indexation

Dans cette évaluation, nous n'avons utilisé que le mécanisme de typage comme heuristique pour filtrer les éléments dans les index. Si ce mécanisme est intéressant, il existe d'autres possibilités d'heuristiques de filtrage.

Un autre attribut fort qui joue comme heuristique est la classification de sûreté. Toutes les exigences ne sont pas valides au même moment. En effet, les normes CEI 60880 et 62138 décrivent les exigences autour des aspects logiciels pour les systèmes réalisant respectivement des fonctions de catégorie A et B ou C. Pour un système de contôle-commande important pour la sûreté, seul un sous-ensemble des exigences est applicable.

Une autre heuristique possible est de considérer les différents regroupements possibles définis dans le métamodèle comme autant d'index possibles. Si les *TypedElementWrapper* reposent déjà sur le typage des exigences et présentent peu d'intérêt, d'autres regroupement contenant des informations avec une sémantique particulière sont intéressants. Les *Topics* offrent la possibilité d'avoir des corpus plus petits mais avec une meilleure adéquation pour répondre à un problème donné. C'est aussi le cas pour les règles de conception (*DesignRules*) et pour les projets qui n'utilisent que des sous-ensembles du corpus.

Troisième partie

Conclusion et perspectives

Conclusion

Conclusion

Résumé de la problématique

Depuis l'implantation du premier système numérique de contrôle-commande à la centrale de Paluel, les systèmes à base de logiciel n'ont cessé de prendre de l'importance dans les systèmes critiques. Ce changement majeur s'est accompagné d'un accroissement significatif de l'arsenal réglementaire vis-à-vis de ces systèmes dont la sûreté de fonctionnement est beaucoup plus difficile à démontrer que pour leurs aînés à base de technologie analogique.

Avec la période de renouveau du nucléaire qui s'est amorcée au début des années 2000, l'industrie du nucléaire a vu émerger un peu partout dans le monde un certain nombre de projets de construction de nouvelles centrales. Pour ces projets, les candidatures françaises, portées par EDF ou Areva avec un EPR certifié en France, ont connu des fortunes diverses dans leur qualification à l'étranger, notamment dans celle du contrôle-commande.

Ces observations de l'histoire récente de l'industrie nucléaire ont montré que si la qualification des systèmes importants pour la sûreté et les exigences réglementaires qui y sont liées sont maitrisées par les acteurs du nucléaire dans le périmètre national, ces questions sont beaucoup moins évidentes quand il s'agit de transposer ces expertises au niveau international. Prendre ces exigences en considération pour la production d'une architecture générique qui soit acceptable réclame de formaliser, des deux côtés, cette connaissance sur le domaine pour être à même de les analyser pour en tirer des similitudes et différences.

Résumé des contributions

Cette thèse s'est attachée à adresser cette question de la formalisation des connaissances du contrôle commande nucléaire dans ce milieu mixte recherche académique et recherche de développement industriel. La première contribution que nous avons présenté dans cette thèse consiste en la description inédite de l'évolution récente du domaine du contrôle-commande nucléaire. Les trois autres contributions majeures de la thèse s'organisent autour de l'approche INCREMENT (Instrumentation aNd Control (Regulatory) Requirements Modeling EnvironmeNT) et en représentent trois facettes : une facette d'ingénierie dirigée par les modèles avec INCREMENT-MDE, une facette traçabilité et

recherche d'information avec INCREMENT-IR, et une facette hybride ou se mêlent les aspects modélisation et recherche d'information avec INCREMENT-Hybrid.

A travers la contribution INCREMENT-MDE, présentée dans le chapitre 3, nous avons parcouru la question de la formalisation du domaine sous la forme de la construction itérative d'un métamodèle du domaine. Ce métamodèle est accompagnée d'une base outillée *IncrementParser, IncrementTools et IncrementGUI*. Cet aspect va au delà de l'état de l'art car, pour la première fois, les exigences ont été au cœur des préoccupations de représentation et pas seulement un point de départ vers un autre objectif. Il s'agissait de représenter les exigences réglementaires pour ce qu'elles sont : des éléments ambigus, non vérifiables, d'un niveau d'abstraction tel qu'elles sont particulièrement difficiles à allouer ou à tracer dans le cycle de vie d'un système. C'est aussi la première fois qu'un corpus d'exigences est considéré d'un point de vue global, comme ayant une structure et une organisation qu'il est important d'appréhender et de capitaliser et non pas comme une succession d'exigences, ou comme un ensemble particulier d'exigences avec des objectifs précis de développement ou de vérification qui leur sont associés.

A travers la contribution INCREMENT-IR, nous sommes allés au delà de la seule sémantique des liens de traçabilité que nous avons proposés dans le métamodèle en mettant en œuvre des approches de recherche d'information pour retrouver de manière semi-automatique des liens de traçabilité entre exigences faisant référence à un même thème, à travers plusieurs documents. A partir d'un ensemble de normes internationales du domaine, nous avons mené des expérimentations autour de la détection de thèmes en adaptant des approches statistiques, de clustering, d'apprentissage et de mesures de similarités. Ces travaux ont montré la viabilité de telles approches appliquées au contexte des exigences réglementaires avec cependant un certain nombre de limitations. L'absence de thésaurus (ou de lexique) qui est à construire pour le domaine n'a pas pu être compensé par des offres tierces telles que les dictionnaires de Stanford ou Wordnet, trop généralistes pour être valables dans un contexte industriel. La seconde limitation vient de la granularité de l'index construit pour la recherche d'information. Celui-ci a été acquis manuellement en découpant les normes jusqu'au plus petit niveau de section. Si cette granularité est suffisante pour la détection de thèmes à travers les différentes approches, celle-ci n'est pas suffisante pour les questions de traçabilité des exigences.

A travers la contribution INCREMENT-Hybrid , nous avons proposé de palier à la segmentation manuelle du corpus d'exigences que nous manipulions en nous basant sur les informations issues du modèle que nous avions construit à partir de la brique *IncrementParser*. En plus de la granularité plus fine à laquelle nous avons pu travailler, nous avons pu acquérir de manière systématique des attributs liés aux éléments de modèle pour fournir un jeu de métadonnées supplémentaires à l'indexation.

L'une des limitations des approches à base de recherche d'information est le caractère "à plat" des documents. Il est impossible, sans métadonnées, de savoir qu'un

document représente une exigence, une section, une norme entière ou simplement un morceau de texte descriptif. En conséquence, lors d'une recherche d'information, une quantité non négligeable de candidats faux positifs sont remontés. Cela ne pose pas de soucis avec un corpus peu important comme dans les analyses de traçabilité d'exigences de la littérature ou alors dans des corpora maîtrisés comme pour les études de recherche d'information classiques. Or, dans notre cas, nous avons choisi une approche globale, sans rien retirer du contenu des documents (en dehors des préambules et tables des matières des documents qui n'apportent aucune information valable). Dans ces conditions, être en possession d'informations supplémentaires se révèle être crucial pour filtrer et paramétrer la recherche d'information et remonter des ensembles pertinents de taille plus réduite en ciblant les documents pertinents et en écartant d'office les documents ne pouvant pas convenir.

Les mesures sur la taille des ensembles de liens candidats générés par nos analyses tenant compte des métadonnées, et en particulier de la nature du document, ont montré une réduction de 65% en moyenne de ces espaces par rapport aux analyses menées de manière classique à plat. Bien qu'il ne nous soit pas possible de valider l'ensemble de l'approche sans mesure de rappel ou de précision en l'absence d'un étalon impossible à constituer, nous pouvons émettre l'hypothèse qu'à score de rappel maintenu, la mesure de précision est améliorée, par construction, de par la taille réduite de l'espace de candidats. Cet effet vient s'ajouter aux nombreuses heuristiques proposées par la communauté et trouve un fort intérêt sur les index contenant une large diversité de documents. Un second contexte favorable pour cette heuristique est le cas où le contexte de validité des exigences (ou plus largement des "documents") varie dans le temps, au fil des contextes. C'est le cas, par exemple, des exigences dépendantes de la classification des systèmes. Un troisième cas d'utilisation est lié à la nature des informations dans leur regroupement thématique, auquel cas la diversité des index proposés en synchronisation des différents regroupements proposés dans le métamodèle peut jouer à plein.

Perspectives

INCREMENT dans le projet CONNEXION et en dehors du monde nucléaire

Notre contribution couvre une partie de la problématique du projet CONNEXION, celle de la capitalisation des connaissances liées aux exigences et de leur vérification vis-à-vis de l'architecture via les mécanismes intermédiaires de règles de conception et les éléments de justification. Pour cette partie, le métamodèle *Connexion* s'est révélé mature pour la représentation d'exigences, qu'ils s'agisse de les acquérir dans le large en extrayant les exigences des normes, ou qu'il s'agisse d'acquérir un ensemble fourni par les partenaires industriels.

Si les questions autour de la variabilité (au sens similarités et différences) entre exigences d'un ou plusieurs corpora) ont toujours fait partie des préoccupations et ont même été présentées puis retirées du métamodèle, cet aspect doit être approfondi dans le cadre du projet CONNEXION et viendra enrichir le métamodèle. De la même manière,

les fonctionnalités d'analyse autour du métamodèle doivent être précisées pour envisager un outil prototype à l'horizon 2014-2015 et seront l'objet de développements futurs.

Si nous nous sommes concentrés sur l'analyse de documents issus du monde nucléaire, le métamodèle et les briques *IncrementParser* et *IncrementIndexer* ont été utilisées pour l'acquisition, la modélisation et l'indexation de documents normatifs issus d'un domaine tiers, en l'occurrence pour la performance thermique des bâtiments (GA P50-784, guide d'application de la norme NF EN 13829 :2001). Cette expérimentation nous conforte dans la généricité du métamodèle et des outils développés autour de ce dernier et dans son extensibilité aux exigences d'autres domaines.

Captures et analyses des données documentaires

Si nous avons proposé, avec *IncrementParser*, un outil pour l'acquisition automatique des documents, et la constitution de corpus de documents et d'exigences, cette opération n'est pas valide systématiquement. En effet, nous reposons sur une approche de pattern matching, en utilisant les propriétés de bonne formation des documents et des exigences. Cependant, nous sommes encore limités à la granularité du paragraphe. Or, contrairement aux exigences bien formées, le périmètre d'une clause exprimant une exigence peut, en fait, en contenir plusieurs, comme dans le cas des listes. Si le métamodèle permet de créer des raffinements aux éléments textuels pour prendre en compte la diversité des décompositions d'un paragraphe, cette analyse supplémentaire reste une tâche manuelle dévolue à l'analyste.

Une autre limitation de notre base outillée provient de la gestion des figures et des tableaux, tableaux comparatifs, qui sont porteurs d'informations. Actuellement, dans les documents réglementaires ou normatifs nucléaires, il n'existe que très peu d'exigences représentées autrement que sous la forme d'entrées textuelles. Cependant, et c'est le cas de la norme IEEE 1012 (norme générique autour de la V&V du logiciel des systèmes importants pour la sûreté), il se peut que des notions très importantes soient contenues dans ces tableaux. Pour la norme IEEE 1012, le périmètre d'application des exigences est défini par la notion de Safety Integrity Level (niveaux de sûreté) et l'application des exigences se fait en fonction de ces niveaux et sont représentés en tableaux que nous ne savons pas traiter, aujourd'hui. Cette prise en compte nécessite un traitement ad hoc.

Exigences, réglementation et conformité dans des contextes multiples

La question de la qualification et de la conformité des exigences dans des contextes multiples est une question récente dans l'histoire de l'ingénierie des exigences. Ce travail figure donc parmi les précurseurs sur le domaine avec des initiatives similaires au projet CONNEXION français avec, par exemple, le projet européen OPENCOSS, lancé depuis octobre 2011, autour de la constitution d'un cadre générique pour la certification avec des acteurs industriels de l'aéronautique, du ferroviaire et de l'automobile chez les industriels. Du côté académique, on retrouve les travaux récents de Gordon et Breaux ou Massey et al. autour des exigences réglementaires et des problématiques liées au cloud computing avec des systèmes géographiquement distribués et devant répondre à

des réglementations différentes.

Aujourd'hui, tout ou presque est encore à construire dans ce domaine, car les approches (y compris la nôtre) se concentrent sur un aspect particulier : la représentation du domaine, la conception d'architecture, l'assurance qualité, les processus, la violation de propriétés ou de lois. Or, les exigences réglementaires peuvent s'exprimer autour de tous les aspects cités et demandent des traitements différenciés qu'il est difficile à détecter par avance. Cette activité nécessiterait des travaux supplémentaires autour des patrons d'exigences, non pas dans leur spécification mais dans la détection de la modélisation adaptée.

La "nature" d'une exigence, son "contrat" [JM97, BJPW99] doit pouvoir être détecté pour ensuite adresser la meilleure forme de représentation voire d'assistance à la certification, ou pour tout autre activité liée à l'exigence. Les travaux autour d'INCREMENT ne prennent pas en compte cette dimension qui est une dimension supplémentaire des exigences telles qu'on les décrit aujourd'hui traditionnellement dans la littérature que d'après leur caractère fonctionnel/non fonctionnel, réglementaire/normative/d'ingénierie, exigences/attentes, etc. Cette notion de contrat a déjà été proposée par Nebut et al. [NFTJ03b] mais elle reste dans la droite lignée des contrats de Jézéquel et Meyer [JM97]pour le développement ou le test. Cette notion doit être étendue dans le cadre d'une vision plus globale des exigences.

"Et maintenant, où va aller le nouveau-né ?
Le Net est vaste et infini ..."
"Major" Motoko Kusanagi
Ghost in the Shell (1995) - Mamoru Oshii
d'après l'oeuvre de Masamune Shirow

Annexe et bibliographie

Annexe A

Détails du métamodèle Connexion

Dans cette annexe, nous présentons brièvement les grands concepts contenus dans le métamodèle *Connexion*. Celui-ci s'articule autour :

1. des "exigences" au sens large puisqu'il y a bien d'autres éléments dans la réglementation ;
2. de la documentation ;
3. des projets ;
4. des regroupements au sein de thèmes ;
5. des regroupements en fonction de la nature des éléments ;
6. des interactions pour la comparaison ou la traçabilité ;
7. des règles de conception ;
8. la justification.

A.1 Exigences et éléments typés

Les éléments typés concernent tous les types d'éléments rencontrés dans les documents/projets au niveau exigence. Il s'agit des exigences (clauses obligatoires), recommandations (clauses optionnelles), des définitions, des tableaux et figures. Une vue de ces éléments est proposée en figure A.1.

La métaclasse abstraite *TypedElement* se spécialise en deux axes, les éléments écrits et les éléments non écrits ; Parmi les éléments écrits se retrouvent l'ensemble des concepts mentionnés ainsi qu'une spécialisation particulière pour les exigences. Une exigence peut donc ainsi être une exigence (sans particularité) ou être plus précise et provenir de l'ingénierie (*OperationalRequirement*, *EngineeringRequirement*), du monde normatif (*StandardRequirement*) ou du monde réglementaire (*WrittenRegulatoryRequirement*). De même, il peut s'agir d'exigences tacites (non écrites) émises par les autorités ou d'une pratique acceptée implicite liées à des projets.

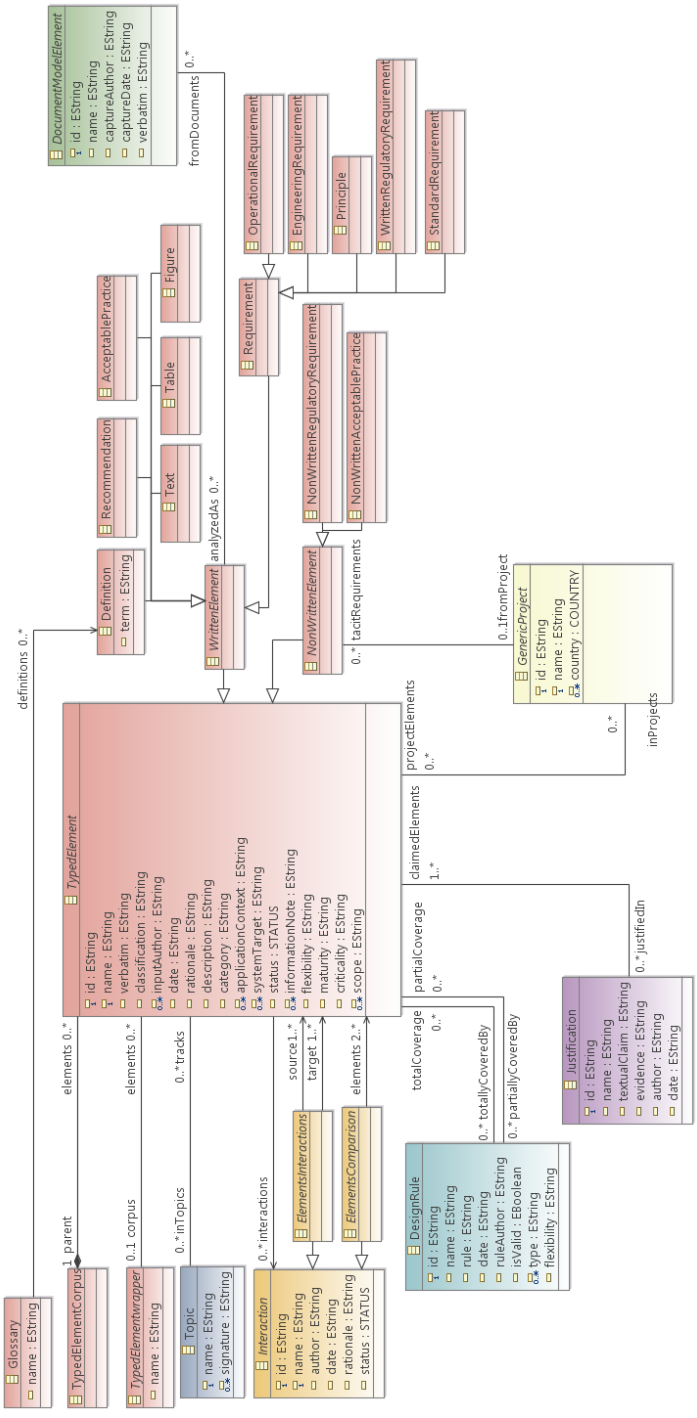


FIGURE A.1 – Vue globale des éléments typés ou "exigences" (au sens large)

A.2 Documents

Les documents sont tous des spécialisations d'un concept abstrait (non instanciable) *DocumentModelElement*. Cela permet de manipuler des documents à tous les niveaux de granularité possibles, qu'il s'agisse d'un document entier, d'un chapitre, d'une section, d'un fragment de texte (paragraphe, phrase). Les documents sont spécialisés autour de trois grands axes : les documents réglementaires, les documents normatifs, les documents de l'ingénierie. Nous présentons ces éléments dans la figure A.2 et définissons les grands concepts.

Pour les documents réglementaires (*GenericRegulatoryDocument*) :

- Document réglementaire (*RegulatoryDocument*) : Il s'agit de documents comme la règle fondamentale de sûreté autour du logiciel (RFS-II.4.1.a) éditée par l'ASN (France), des Safety Assessment Principles (SAP) éditées par le HSE (Royaume-Uni), de la 10CFR50 (Code of Federal Regulation) éditée par la NRC (Etats-Unis)
- Guide réglementaire (*RegulatoryGuidance*) : Il n'existe pas vraiment de documents de ce genre dans le contexte français. Cependant, on en retrouve aux Etats-Unis ou au Royaume Uni. Il s'agit de documents comme les guides réglementaires US (NRC regulatory guide, par exemple le NRC regulatory guide 1-153 « Criteria for Safety System »), ou des Technical Assessment Principles (TAG) publié par le HSE (Health Safety Executive devenu ONR depuis).
- Position réglementaire (*RegulatoryPosition*) : Il s'agit ici de documents relatifs à des décisions/avis des autorités sur des points précis spécifiques à des projets et non pas de documents à caractère perpétuel comme les documents et guides réglementaires.

Pour les documents normatifs (*GenericNormativeDocument*) :

- Les normes (*Standard*) : Il s'agit des normes, recommandations, standards, publiés par des organismes nationaux ou internationaux et qui sont appliqués pour les projets de système de contrôle-commande.

Pour les documents d'ingénierie (*GenericEngineeringDocument*) :

- Codes techniques (*TechnicalCode*)
- Documents d'ingénierie (*EngineeringDocument*)

Les documents d'ingénierie concernent essentiellement les exigences « industrielles », « opérationnelles ».

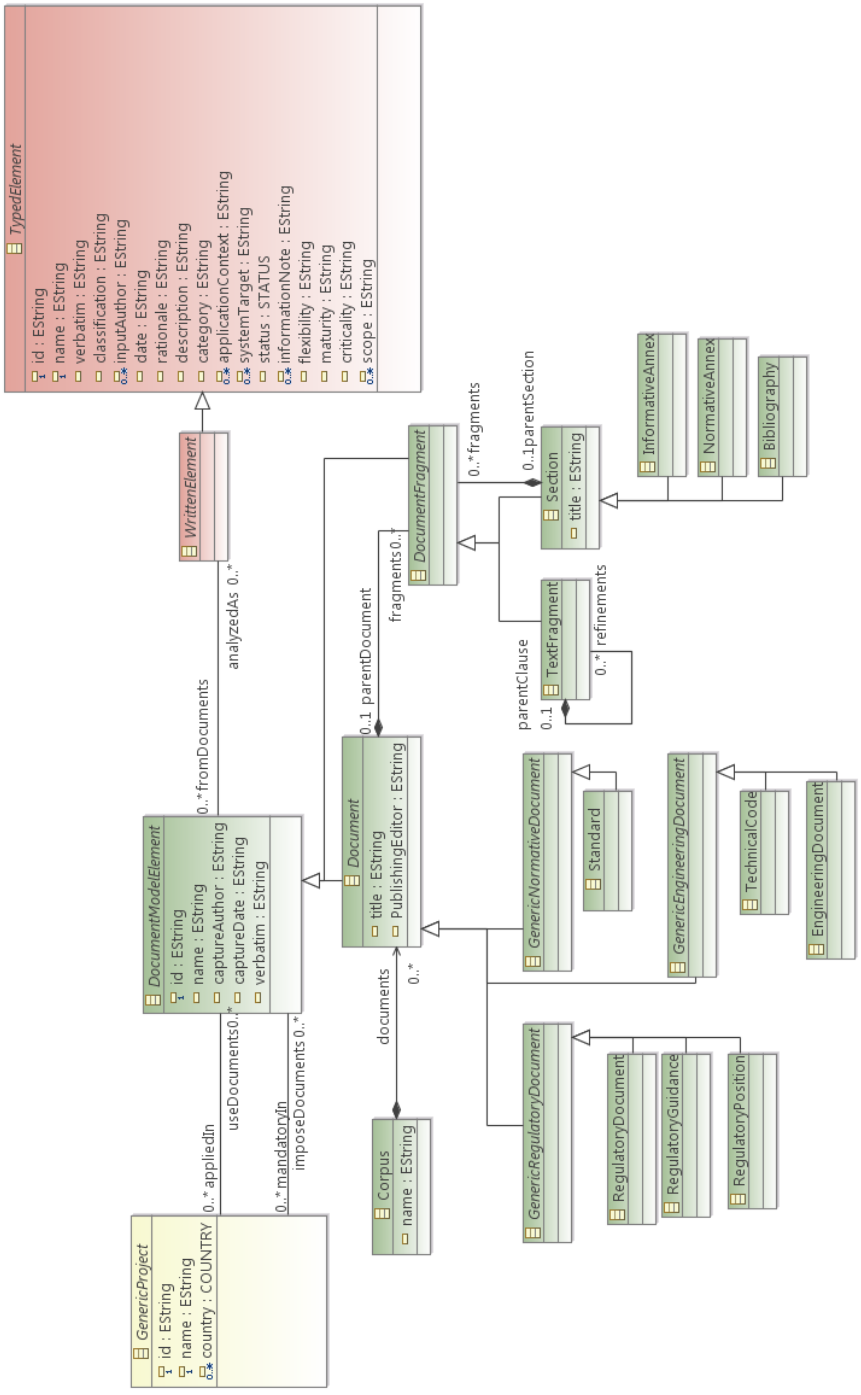


FIGURE A.2 – Vue des documents

A.3 Projets de systèmes de contrôle-commande

La notion de projet définit un périmètre d'application pour certaines exigences et certains des documents du référentiel d'exigences. Pour chaque projet, de nouvelles exigences particulières, non écrites (tacites), peuvent survenir et vont construire la pratique. Ils peuvent être spécifiques à un pays ou en viser plusieurs et référencent les éléments typés et les documents à appliquer pertinents.

A l'instar des thèmes et des regroupements par nature, le projet forme un ensemble cohérent d'exigences et de documents. Tandis que le thème concerne une préoccupation particulière sur le contrôle-commande, tandis que les regroupement offrent une perspective en fonction de la nature des éléments, le projet offre une vue différente liée au périmètre du système de contrôle-commande réaliser.

La métaclasse générique *GenericProject* permette de créer des projets "spécifiques" (*Project*), ou plus "généraux" (*GeneralProject*) tout en référençant les projets dont ils sont une généralisation (association *projects*).

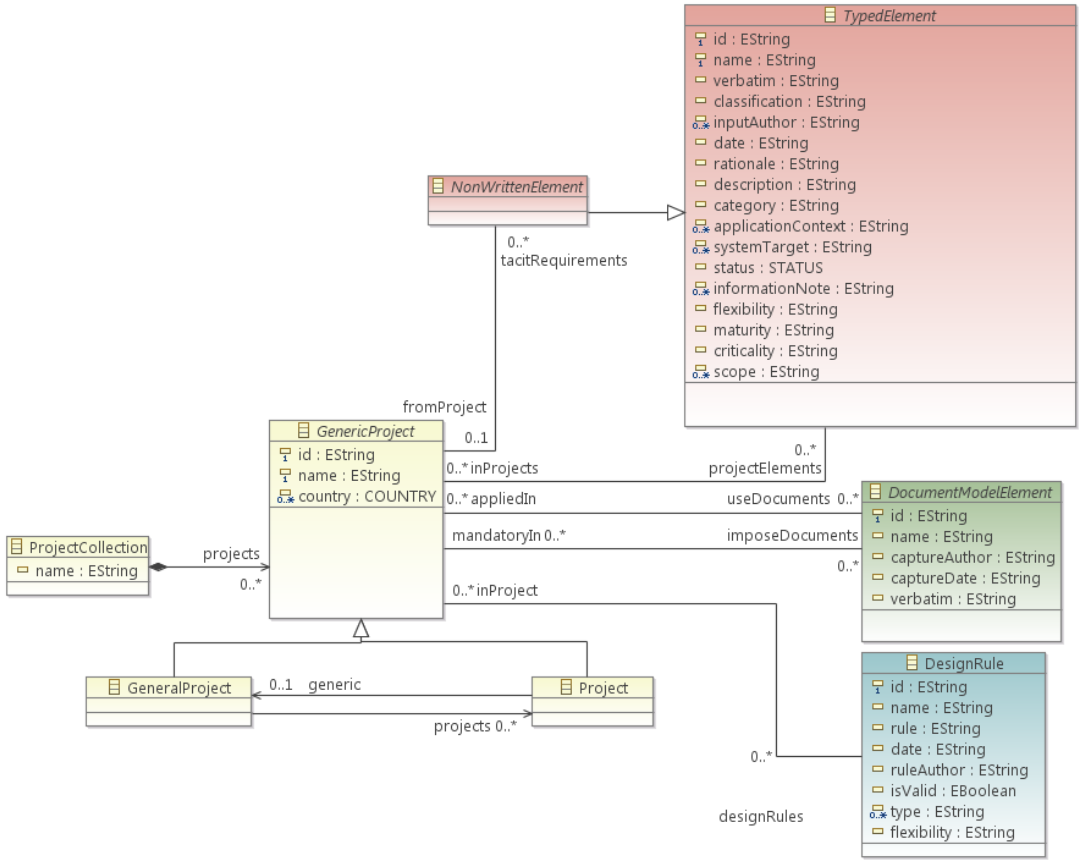


FIGURE A.3 – Vue des projets

A.4 Gestion des thèmes

Les thèmes/topics (*Topic*) sont utilisés pour la classification des exigences au sein de regroupement sémantiques. L'ensemble des éléments capturés peuvent varier en termes de portée (système ou partie de l'architecture impacté) ou de scope (thème ou aspect de l'architecture impacté). Une analyse fine de ces exigences n'est possible que sur un volume raisonnable d'éléments. L'objectif des thèmes de réduire le volume d'éléments à montrer à un ingénieur par la proposition de regroupements d'éléments proches. Les thèmes peuvent être raffinés et hiérarchisés en niveaux. Comme les regroupements génériques, ils permettent une organisation et un point de vue différents sur le référentiel d'exigences.

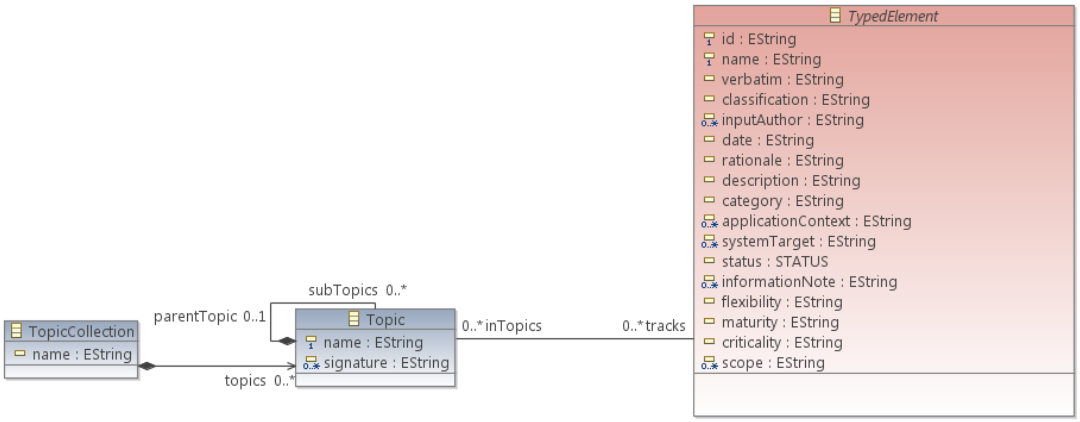


FIGURE A.4 – Vue des thèmes

A.5 Regroupements selon la nature des éléments

A la différence des documents qui ont une structure (sections, sous-sections, annexes, clauses, etc.) les éléments typés ne le sont pas et il est nécessaire de proposer des regroupements afin de rassembler et manipuler plus facilement les éléments typés.

Ces regroupements sont liés à la nature des éléments. Le parent d'un *TypedElement* est un regroupement générique *TypedElementCorpus*. Il existe cependant quatre autres types de regroupement qui sont liés à la provenance des éléments :

- *RegulatoryElementCorpus* : pour les exigences du monde réglementaire ;
- *NormativeElementCorpus* : pour les exigences des normes / standards ;
- *EngineeringElementCorpus* : pour les éléments venant de l'ingénierie ;
- *Glossary* : pour regrouper les définitions.

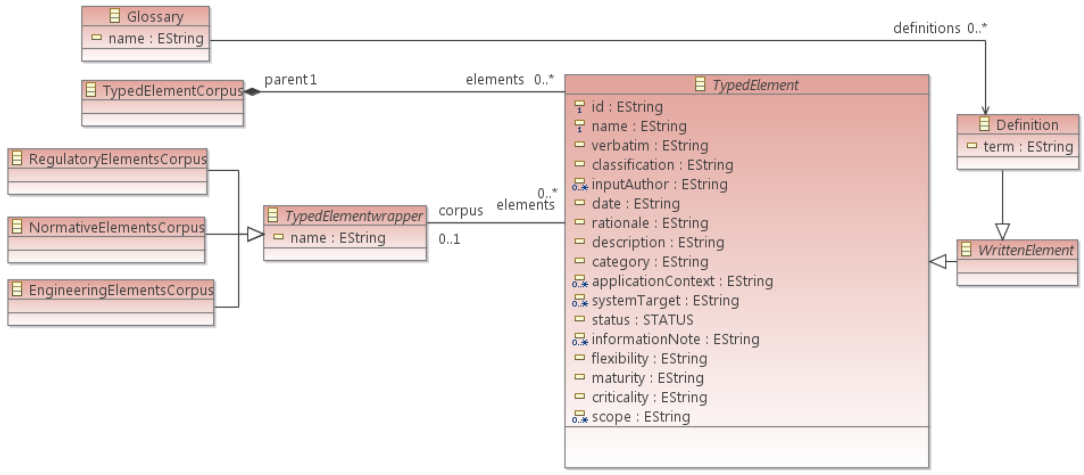


FIGURE A.5 – Vue des regroupements thématiques

A.6 Interactions pour la traçabilité des exigences et leur comparaison

Les interactions sont les types de liens, explicites ou implicites, entre deux ou plusieurs éléments typés du modèle. Nous considérons ici deux types d' interactions : (1) les interactions avec une sémantique de comparaison et (2) les interactions et qui visent le cadre plus large de la traçabilité.

Au niveau de la comparaison :

- Contradiction Forte/Absolue (*StrongConflict*) : il s'agit de représenter un caractère exclusif entre deux éléments qui ne peuvent être justifiées/réalisables en même temps. Incompatibilité absolue.
- Contradiction faible (*WeakConflict*) : il s'agit de représenter une contradiction entre deux éléments qui vont dans des sens opposés, mais lesquels, un point d'équilibre peut être trouvé afin de justifier/réaliser partiellement chacune.
- Equivalence forte (*TotalEquivalence*) : La portée, le contexte, la classification ainsi que le besoin exprimé de l'exigence sont identiques en termes de verbatim.
- Equivalence faible (*PartialEquivalence*) : La portée, le contexte, la classification ainsi que le besoin exprimé de l'exigence sont similaires mais non identiques.

Au niveau des liens de traçabilité :

- Référence (*Reference*) : il s'agit d'un lien de référence, implicite ou explicite, entre deux éléments. Les normes se font références entre elles.
- Requiert (*Require*) : Il s'agit d'un lien d'implication, plus fort que le lien de référence, entre deux éléments. Il peut apparaître, par exemple, dans la 10CFR50 qui impose l'application de la norme IEEE603 pour tous les systèmes.
- Spécialisation (*Specialize*) : Il s'agit de représenter une notion de vue (une exigence peut être implémentée de façon différente suivant la vue, architecture, logiciel ...)

entre deux éléments, en fonction de spécificités. C'est une relation inverse de la généralisation.
- Généralisation (*Generalize*) : il s'agit de la relation inverse de la spécialisation.
- Participe (*Contribute*) : Une exigence peut participer à la réalisation de plusieurs exigences. Une exigence peut-être la somme de plusieurs exigences participatives.
- Pratique acceptée (*AcceptedPractice*) : Une autorité reconnaît que pour répondre à une exigence particulière, l'application d'un autre (souvent à une granularité plus haute) élément est une pratique acceptable. L'exigence e2a.1 autour de la V&V du logiciel reconnaît ainsi que l'application des chapitres 6 (Software Verification), 7 (Software Integration) et 8 (System Validation) de la CEI 60880-1986 sont des pratiques acceptables pour les exigences de fiabilité du logiciel.
- Couvre (*Cover*) : exprime une similitude entre deux éléments mais que l'un des deux va plus loin dans les contraintes ou les propriétés ou les définitions.

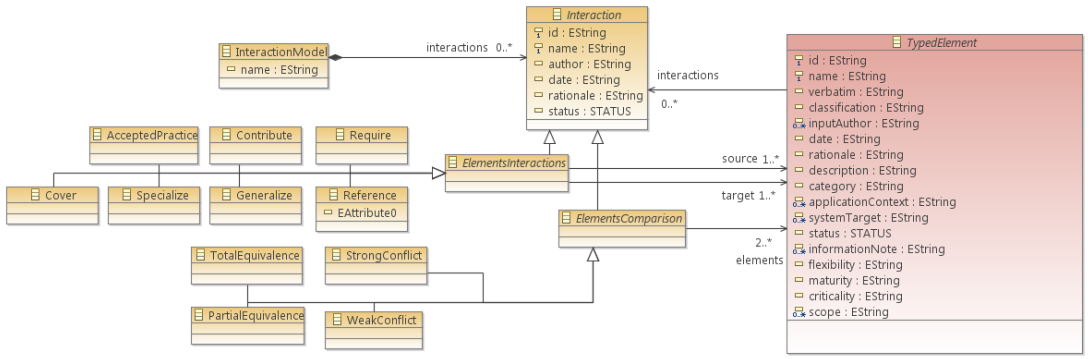


FIGURE A.6 – Vue des interactions pour la traçabilité et la comparaison

A.7 Règles de conception

Les règles de conception sont des éléments intermédiaires permettant de faire le pont entre les exigences réglementaires ou normatives, souvent ambiguës et non directement applicables et l'architecture. Les règles de conception sont ces indirections. Elles s'expriment sous la forme d'une règle énoncée ainsi qu'un lien vers une justification.

Une règle de conception peut satisfaire totalement ou partiellement une ou plusieurs exigences.

A.8 Justification

La justification est un des éléments du triptyque <exigence, architecture, justification> qui a initié le métamodèle. La justification est basée sur une assertion (textualClaim) qui référence les exigences liées à la justification, les règles de conception associées ainsi qu'une argumentation (evidence) qui étaye la justification.

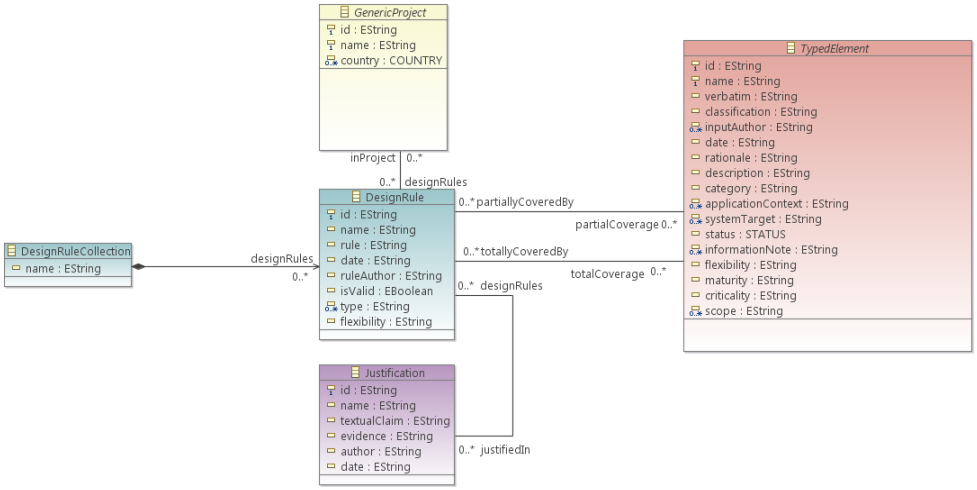


FIGURE A.7 – Vue des règles de conception pour l'architecture

Ce concept très minimaliste n'est pas mature dans le métamodèle et sera discuté par les partneaires du projet CONNEXION.

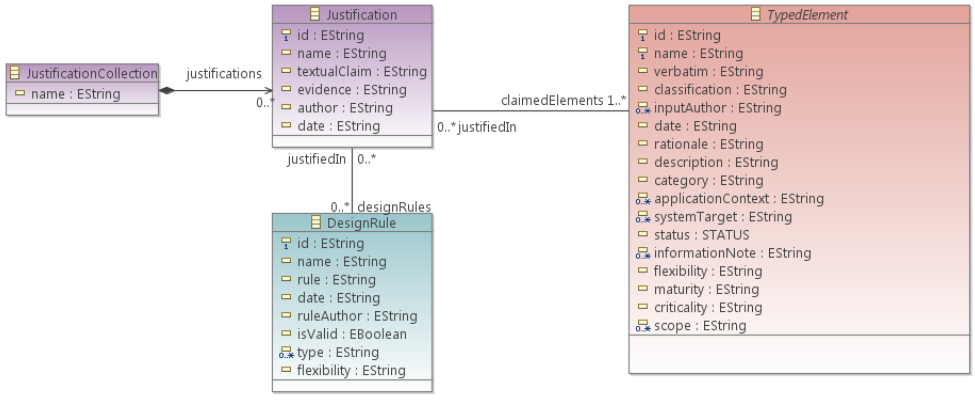


FIGURE A.8 – Vue des justifications

Bibliographie

[AAT10] Hazeline U. Asuncion, Arthur U. Asuncion, and Richard N. Taylor. Software traceability with topic modeling. In *Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering - Volume 1*, ICSE '10, pages 95–104, New York, NY, USA, 2010. ACM.

[ABKO04] A. Alfonso, V. Braberman, N. Kicillof, and A. Olivero. Visual Timed Event Scenarios. In *Proceedings of the 26th International Conference on Software Engineering*, ICSE '04, pages 168–177, Washington, DC, USA, 2004. IEEE Computer Society.

[ABLG12] Manzoor Ahmad, Jean-Michel Bruel, Régine Laleau, and Christophe Gnaho. Using RELAX, SysML and KAOS for Ambient Systems Requirements Modeling. *Procedia Computer Science*, 10(0) :474 – 481, 2012. ANT 2012 and MobiWIS 2012.

[AGI+13] Silvia Abrahao, Carmine Gravino, Emilio Insfran, Giuseppe Scanniello, and Genoveffa Tortora. Assessing the Effectiveness of Sequence Diagrams in the Comprehension of Functional Requirements : Results from a Family of Five Experiments. *IEEE Transactions on Software Engineering*, 39(3) :327–342, 2013.

[AIE07] AIEA. *IAEA Safety Glossary : Terminology Used in Nuclear Safety and Radiation Protection, 2007 Edition*. IAEA publications, 2007.

[AM11] Daniel Amyot and Gunter Mussbacher. User Requirements Notation : The First Ten Years, The Next Ten Years (Invited Paper). *Journal of Software (JSW)*, 6(5) :747–768, 2011.

[AMGK11] Daniel Amyot, Gunter Mussbacher, Sepideh Ghanavati, and Jason Kealey. GRL Modeling and Analysis with jUCMNav. In Jaelson Brelaz de Castro, Xavier Franch, John Mylopoulos, and Eric S. K. Yu, editors, *iStar*, volume 766 of *CEUR Workshop Proceedings*, pages 160–162. CEUR-WS.org, 2011.

[ARSM99] Camille Ben Achour, Colette Rolland, Carine Souveyet, and Neil A. M. Maiden. Guiding Use Case Authoring : Results of an Empirical Study. In *RE*, pages 36–43. IEEE Computer Society, 1999.

[ASN11] Autorité de sûreté nucléaire ASN. Complementary Safety Assessments of the French Nuclear Power Plants (European Stress Tests). Technical report, ASN, 2011.

[BABD09] Travis D. Breaux, Annie I. Antón, Kent Boucher, and Merlin Dorfman. IT Compliance : Aligning Legal and Product Requirements. *IT Professional*, 11(5) :54–58, 2009.

[BAD08] Travis D. Breaux, Annie I. Antón, and Jon Doyle. Semantic parameterization : A process for modeling domain descriptions. *ACM Trans. Softw. Eng. Methodol.*, 18(2), 2008.

[BBT+07] Erwan Brottier, Benoit Baudry, Yves Le Traon, David Touzet, and Bertrand Nicolas. Producing a Global Requirement Model from Multiple Requirement Specifications. In Donald W. Sparrow Jr., Marcus Spies, and M. Brian Blake, editors, *EDOC*, pages 390–404. IEEE Computer Society, 2007.

[BC04] Elisa L. A. Baniassad and Siobhán Clarke. Theme : An Approach for Aspect-Oriented Analysis and Design. In Anthony Finkelstein, Jacky Estublier, and David S. Rosenblum, editors, *ICSE*, pages 158–167. IEEE Computer Society, 2004.

[BCBB11] Arnaud Blouin, Benoît Combemale, Benoit Baudry, and Olivier Beaudoux. Modeling model slicers. *Software and Systems Modeling*, pages 62–76, 2011.

[BCS+12] Edna Braun, Nick Cartwright, Azalia Shamsaei, Saeed Ahmadi Behnam, Gregory Richards, Gunter Mussbacher, Mohammad Alhaj, and Rasha Tawhid. Drafting and modeling of regulations : Is it being done backwards ? In Annie Antón, Travis Breaux, Daniel Amyot, and Wade Chumney, editors, *RELAW*, pages 1–6. IEEE, 2012.

[Ber02] Daniel M. Berry. The importance of ignorance in requirements engineering : An earlier sighting and a revisitation. *Journal of Systems and Software*, 60(1) :83–85, 2002.

[BFS+06] Erwan Brottier, Franck Fleurey, Jim Steel, Benoit Baudry, and Yves Le Traon. Metamodel-based Test Generation for Model Transformations : an Algorithm and a Tool. In *ISSRE*, pages 85–94. IEEE Computer Society, 2006.

[BG13] Travis D. Breaux and David G. Gordon. Regulatory Requirements Traceability and Analysis Using Semi-formal Specifications. In Joerg Doerr and Andreas L. Opdahl, editors, *REFSQ*, volume 7830 of *Lecture Notes in Computer Science*, pages 141–157. Springer, 2013.

[BJPW99] Antoine Beugnard, Jean-Marc Jezequel, Noel Plouzeau, and Damien Watkins. Making components contract aware. *Computer*, 32(7) :38–45, 1999.

[BMC+12] Erwan Bousse, David Mentré, Benoit Combemale, Benoit Baudry, and Katsuragi Takaya. Aligning SysML with the B Method to Provide V&V for Systems Engineering. In *Model-Driven Engineering, Verification, and Validation 2012 (MoDeVVa 2012*, Innsbruck, Austria, September 2012.

[BNJ03] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 3 :993–1022, 2003.

[BNT07] Benoit Baudry, Clémentine Nebut, and Yves Le Traon. Model-Driven Engineering for Requirements Analysis. In Donald W. Sparrow Jr., Marcus Spies, and M. Brian Blake, editors, *EDOC*, pages 459–466. IEEE Computer Society, 2007.

[BVA06] Travis D. Breaux, Matthew W. Vail, and Annie I. Anton. Towards regulatory compliance : Extracting rights and obligations to align requirements with regulations. In *RE'06 : Proceedings of the 14th IEEE International Requirements Engineering Conference (RE'06)*, pages 49–58, Washington, DC, USA, September 2006. IEEE Society Press.

[BYRN+99] Ricardo Baeza-Yates, Berthier Ribeiro-Neto, et al. *Modern Information Retrieval*, volume 463. ACM press New York, 1999.

[CA07] Betty H. C. Cheng and Joanne M. Atlee. Research Directions in Requirements Engineering. In Lionel C. Briand and Alexander L. Wolf, editors, *FOSE*, pages 285–303, 2007.

[CCHcB12] Adam Czauderna, Jane Cleland-Huang, Murat Çinar, and Brian Berenbach. Just-in-time traceability for mechatronics systems. In *RESS*, pages 1–9. IEEE, 2012.

[CCMS02] Laura A. Campbell, Betty H. C. Cheng, William E. McUmber, and Kurt Stirewalt. Automatically Detecting and Visualising Errors in UML Diagrams. *Requirements Engineering*, 7(4) :264–287, 2002.

[CG11] Xiaofan Chen and John Grundy. Improving automated documentation to code traceability by combining retrieval techniques. In *Proceedings of the 2011 26th IEEE/ACM International Conference on Automated Software Engineering*, pages 223–232. IEEE Computer Society, 2011.

[CHBC+07] Jane Cleland-Huang, Brian Berenbach, Stephen Clark, Raffaella Settimi, and Eli Romanova. Best practices for automated traceability. *IEEE Computer*, 40(6) :27–35, 2007.

[CHCGE10] Jane Cleland-Huang, Adam Czauderna, Marek Gibiec, and John Emenecker. A machine learning approach for tracing regulatory codes to product specific requirements. In Jeff Kramer, Judith Bishop, Premkumar T. Devanbu, and Sebastián Uchitel, editors, *ICSE (1)*, pages 155–164. ACM, 2010.

[CHHH+12] Jane Cleland-Huang, Mats Per Erik Heimdahl, Jane Huffman Hayes, Robyn R. Lutz, and Patrick Maeder. Trace Queries for Safety Requirements in High Assurance Systems. In Björn Regnell and Daniela E. Damian, editors, *REFSQ*, volume 7195 of *Lecture Notes in Computer Science*, pages 179–193. Springer, 2012.

[CHSDZ05] J. Cleland-Huang, R. Settimi, Chuan Duan, and Xuchang Zou. Utilizing supporting evidence to improve dynamic requirements traceability. In *Requirements Engineering, 2005. Proceedings. 13th IEEE International Conference on*, pages 135–144, 2005.

[CSBW09] Betty H. C. Cheng, Peter Sawyer, Nelly Bencomo, and Jon Whittle. A Goal-Based Modeling Approach to Develop Requirements of an Adaptive System with Environmental Uncertainty. In Andy Schürr and Bran Selic, editors, *MoDELS*, volume 5795 of *Lecture Notes in Computer Science*, pages 468–483. Springer, 2009.

[DAC99] Matthew B Dwyer, George S Avrunin, and James C Corbett. Patterns in property specifications for finite-state verification. In *Software Engineering, 1999. Proceedings of the 1999 International Conference on*, pages 411–420. IEEE, 1999.

[dAFdS12] David de Almeida Ferreira and Alberto Rodrigues da Silva. RSLingo : An information extraction approach toward formal requirements specifications. In *MoDRE*, pages 39–48, 2012.

[DGH+11] Horatiu Dumitru, Marek Gibiec, Negar Hariri, Jane Cleland-Huang, Bamshad Mobasher, Carlos Castro-Herrera, and Mehdi Mirakhorli. On-demand feature recommendations derived from mining public product descriptions. In *Proceedings of the 33rd International Conference on Software Engineering*, ICSE '11, pages 181–190, New York, NY, USA, 2011. ACM.

[DHH05] Alex Dekhtyar and Jane Hayes HUffman. A framework for comparing requirements tracing experiments. *International Journal of Software Engineering and Knowledge Engineering*, 15(05) :751–781, 2005.

[DLFOT05] Andrea De Lucia, Fausto Fasano, Rocco Oliveto, and Genoveffa Tortora. Adams re-trace : A traceability recovery tool. In *Software Maintenance and Reengineering, 2005. CSMR 2005. Ninth European Conference on*, pages 32–41. IEEE, 2005.

[DLR13] Leticia Duboc, Emmanuel Letier, and David S. Rosenblum. Systematic elaboration of scalability requirements through goal-obstacle analysis. *IEEE Trans. Software Eng.*, 39(1) :119–140, 2013.

[DLVL06] Christophe Damas, Bernard Lambeau, and Axel Van Lamsweerde. Scenarios, goals, and state machines : a win-win partnership for model synthesis. In *Proceedings of the 14th ACM SIGSOFT international symposium on Foundations of software engineering*, pages 197–207. ACM, 2006.

[DMMM+06] Marie-Catherine De Marneffe, Bill MacCartney, Christopher D Manning, et al. Generating typed dependency parses from phrase structure parses. In *Proceedings of LREC*, volume 6, pages 449–454, 2006.

[FKMT05] Kathi Fisler, Shriram Krishnamurthi, Leo A. Meyerovich, and Michael Carl Tschantz. Verification and change-impact analysis of access-control policies. In Gruia-Catalin Roman, William G. Griswold, and Bashar Nuseibeh, editors, *ICSE*, pages 196–205. ACM, 2005.

[GAP07] Sepideh Ghanavati, Daniel Amyot, and Liam Peyton. Towards a Framework for Tracking Legal Compliance in Healthcare. In John Krogstie,

Andreas L. Opdahl, and Guttorm Sindre, editors, *CAiSE*, volume 4495 of *Lecture Notes in Computer Science*, pages 218–232. Springer, 2007.

[GB12] David G. Gordon and Travis D. Breaux. Reconciling multi-jurisdictional legal requirements : A case study in requirements water marking. In Mats Per Erik Heimdahl and Pete Sawyer, editors, *RE*, pages 91–100. IEEE, 2012.

[GB13] David G. Gordon and Travis D. Breaux. A cross-domain empirical study and legal evaluation of the requirements water marking method. *Requirements Engineering*, 18(2) :147–173, 2013.

[GCHH+12] Orlena Gotel, Jane Cleland-Huang, Jane Huffman Hayes, Andrea Zisman, Alexander Egyed, Paul Grünbacher, and Giuliano Antoniol. The quest for Ubiquity : A roadmap for software and systems traceability research. In Mats Per Erik Heimdahl and Pete Sawyer, editors, *RE*, pages 71–80. IEEE, 2012.

[GF94] Orlena CZ Gotel and Anthony CW Finkelstein. An analysis of the requirements traceability problem. In *Proceedings of the 1st International Conference on Requirements Engineering (ICRE'94), Colorado Springs, Colorado, USA, April 18-22 1994*, pages 94–101. IEEE, 1994.

[GHJV93] Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides. *Design patterns : Abstraction and reuse of object-oriented design*. Springer, 1993.

[GKvdB08] Arda Goknil, Ivan Kurtev, and Klaas van den Berg. A Metamodeling Approach for Reasoning about Requirements. In Ina Schieferdecker and Alan Hartman, editors, *ECMDA-FA*, volume 5095 of *Lecture Notes in Computer Science*, pages 310–325. Springer, 2008.

[GM11] Orlena Gotel and Stephen J. Morris. Out of the labyrinth : Leveraging other disciplines for requirements traceability. In Mylopoulos and Heymans [MH11], pages 121–130.

[GPF12] Arda Goknil and Marie-Agnès Peraldi-Frati. A DSL for specifying timing requirements. In Mussbacher et al. [MAS12c], pages 49–57.

[GRS+97] Georges Grosz, Colette Rolland, Sylviane R. Schwer, Carine Souveyet, Véronique Plihon, Samira Si-Said, Camille Ben Achour, and Christophe Gnaho. Modelling and Engineering the Requirements Engineering Process : An Overview of the NATURE Approach. *Requir. Eng.*, 2(3) :115–131, 1997.

[HDS06] Jane Huffman Hayes, Alex Dekhtyar, and Senthil Karthikeyan Sundaram. Advancing candidate link generation for requirements tracing : The study of methods. *Software Engineering, IEEE Transactions on*, 32(1) :4–19, 2006.

[HDS+07] Jane Huffman Hayes, Alex Dekhtyar, Senthil Karthikeyan Sundaram, Ashlee Holbrook, Sravanthi Vadlamudi, and Alain April. REquirements TRacing On target (RETRO) : improving software maintenance through

traceability recovery. *Innovations in Systems and Software Engineering*, 3(3) :193–202, 2007.

[HK10] Jonas Helming and Maximilian Koegel. Managing iterations with UNICASE. In Jeff Kramer, Judith Bishop, Premkumar T. Devanbu, and Sebastián Uchitel, editors, *ICSE (2)*, pages 313–314. ACM, 2010.

[HMM12] Stefan Henß, Martin Monperrus, and Mira Mezini. Semi-automatically extracting FAQs to improve accessibility of software development knowledge. In Martin Glinz, Gail C. Murphy, and Mauro Pezzè, editors, *ICSE*, pages 793–803. IEEE, 2012.

[IEE90] IEEE. IEEE Standard Glossary of Software Engineering Terminology. *IEEE Std 610.12-1990*, pages 1–84, 1990.

[ISM11] Silvia Ingolfo, Alberto Siena, and John Mylopoulos. Establishing Regulatory Compliance for Software Requirements. In Manfred A. Jeusfeld, Lois M. L. Delcambre, and Tok Wang Ling, editors, *ER*, volume 6998 of *Lecture Notes in Computer Science*, pages 47–61. Springer, 2011.

[Jac95] Michael Jackson. The world and the machine. In *Software Engineering, 1995. ICSE 1995. 17th International Conference on*, pages 283–283, 1995.

[JCB+13] Jean-Marc Jézéquel, Benoit Combemale, Olivier Barais, Martin Monperrus, and François Fouquet. Mashup of metalanguages and its implementation in the kermeta language workbench. *Software & Systems Modeling*, pages 1–16, 2013.

[JCV12] Jean-Marc Jézéquel, Benoit Combemale, and Didier Vojtisek. *Ingénierie Dirigée par les Modèles : des concepts à la pratique...* Références sciences. Ellipses, February 2012.

[JM97] Jean-Marc. Jézéquel and Bertrand Meyer. Design by contract : the lessons of ariane. *Computer*, 30(1) :129–130, 1997.

[Joh01] Gary Johnson. Comparison of IEC and IEEE standards for computer-based control systems important to safety. In *Nuclear Science Symposium Conference Record, 2001 IEEE*, volume 4, pages 2474–2481. IEEE, 2001.

[Kam05] Erik Kamsties. Understanding ambiguity in requirements engineering. In *Engineering and Managing Software Requirements*, pages 245–266. Springer, 2005.

[KC02] Sascha Konrad and Betty H. C. Cheng. Requirements Patterns for Embedded Systems. In *RE*, pages 127–136. IEEE Computer Society, 2002.

[KCH+90] Kyo C Kang, Sholom G Cohen, James A Hess, William E Novak, and A Spencer Peterson. Feature-oriented domain analysis (FODA) feasibility study. Technical report, DTIC Document, 1990.

[KSTT84] Noriaki Kano, Nobuhiko Seraku, Fumio Takahashi, and Shinichi Tsuji. Attractive quality and must-be quality. *Journal of the Japanese Society for Quality Control*, 14(2) :147–156, 1984.

[KZB+07] Nadzeya Kiyavitskaya, Nicola Zeni, Travis D. Breaux, Annie I. Antón, James R. Cordy, Luisa Mich, and John Mylopoulos. Extracting rights and obligations from regulations : toward a tool-supported process. In R. E. Kurt Stirewalt, Alexander Egyed, and Bernd Fischer, editors, *ASE*, pages 429–432. ACM, 2007.

[KZB+08] Nadzeya Kiyavitskaya, Nicola Zeni, Travis D. Breaux, Annie I. Antón, James R. Cordy, Luisa Mich, and John Mylopoulos. Automating the Extraction of Rights and Obligations for Regulatory Compliance. In Qing Li, Stefano Spaccapietra, Eric S. K. Yu, and Antoni Olivé, editors, *ER*, volume 5231 of *Lecture Notes in Computer Science*, pages 154–168. Springer, 2008.

[Lap07] Jean-Claude Laprie. Safety Demonstration and Software Development. In Francesca Saglietti and Norbert Oster, editors, *SAFECOMP*, volume 4680 of *Lecture Notes in Computer Science*, pages 289–300. Springer, 2007.

[Lev12] Nancy G. Leveson. *Engineering a safer world : Systems thinking applied to safety*. Mit Press, 2012.

[LFOT07] Andrea De Lucia, Fausto Fasano, Rocco Oliveto, and Genoveffa Tortora. Recovering traceability links in software artifact management systems using information retrieval methods. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, 16(4) :13, 2007.

[LNHK11] Yang Li, Nitesh Narayan, Jonas Helming, and Maximilian Koegel. A domain specific requirements model for scientific computing. In Richard N. Taylor, Harald Gall, and Nenad Medvidovic, editors, *ICSE*, pages 848–851. ACM, 2011.

[LO10] Jörg Leuser and Daniel Ott. Tackling Semi-automatic Trace Recovery for Large Specifications. In Roel Wieringa and Anne Persson, editors, *REFSQ*, volume 6182 of *Lecture Notes in Computer Science*, pages 203–217. Springer, 2010.

[LSM+10] Régine Laleau, Farida Semmak, Abderrahman Matoussi, Dorian Petit, Ahmed Hammad, and Bruno Tatibouët. A first attempt to combine SysML requirements diagrams and B. *ISSE*, 6(1-2) :47–54, 2010.

[LST+11] Vincent Lorenzo, Rémi Schnekenburger, Yann Tanguy, Patrick Tessier, and Sébastien Gerard. L'outil de modélisation graphiquemdt : Papyrus état actuel et perspectives. *Génie logiciel*, (97), 2011.

[LTE+09] Agnes Lanusse, Yann Tanguy, Huascar Espinoza, Chokri Mraidha, Sebastien Gerard, Patrick Tessier, Remi Schnekenburger, Hubert Dubois, and François Terrier. Papyrus UML : an open source toolset for MDA. In *Proc. of the Fifth European Conference on Model-Driven Architecture Foundations and Applications (ECMDA-FA 2009)*, pages 1–4. Citeseer, 2009.

[LVD06] Marco Lormans and Arie Van Deursen. Can LSI help reconstructing requirements traceability in design and test ? In *Software Maintenance and Reengineering, 2006. CSMR 2006. Proceedings of the 10th European Conference on*, pages 10–pp. IEEE, 2006.

[LW00] Dean Leffingwell and Don Widrig. *Managing software requirements : a unified approach*. Addison-Wesley Professional, 2000.

[MA09] Jeremy C. Maxwell and Annie I. Antón. Developing Production Rule Models to Aid in Acquiring Requirements from Legal Texts. In William N. Robinson and Kevin Ryan, editors, *RE*, pages 101–110. IEEE Computer Society, 2009.

[MAS11a] Jeremy C. Maxwell, Annie I. Antón, and Peter Swire. A legal cross-references taxonomy for identifying conflicting software requirements. In Mylopoulos and Heymans [MH11], pages 197–206.

[MAS11b] Jeremy C Maxwell, Annie I Antón, and Peter Swire. A legal cross-references taxonomy for identifying conflicting software requirements. In *Requirements Engineering Conference (RE), 2011 19th IEEE International*, pages 197–206. IEEE, 2011.

[MAS12a] Jeremy C. Maxwell, Annie I. Antón, and Peter Swire. Managing changing compliance requirements by predicting regulatory evolution. In Mats Per Erik Heimdahl and Pete Sawyer, editors, *RE*, pages 101–110. IEEE, 2012.

[MAS+12b] Jeremy C. Maxwell, Annie I. Antón, Peter Swire, Maria Riaz, and Christopher M. McCraw. A legal cross-references taxonomy for reasoning about compliance requirements. *Requir. Eng.*, 17(2) :99–115, 2012.

[MAS12c] Gunter Mussbacher, Joao Araujo, and Pablo Sanchez, editors. *Second IEEE International Workshop on Model-Driven Requirements Engineering, MoDRE 2012, Chicago, IL, USA, September 24, 2012*. IEEE, 2012.

[MBC+13] Martin Monperrus, Benoit Baudry, Joël Champeau, Brigitte Hoeltzener, and Jean-Marc Jézéquel. Automated measurement of models of requirements. *Software Quality Journal*, 21(1) :3–22, 2013.

[McC02] Andrew Kachites McCallum. MALLET : A Machine Learning for Language Toolkit. http ://mallet.cs.umass.edu, 2002.

[MCH10] Patrick Mäder and Jane Cleland-Huang. A Visual Traceability Modeling Language. In Dorina C. Petriu, Nicolas Rouquette, and Øystein Haugen, editors, *MoDELS (1)*, volume 6394 of *Lecture Notes in Computer Science*, pages 226–240. Springer, 2010.

[MCN92] John Mylopoulos, Lawrence Chung, and Brian A. Nixon. Representing and using nonfunctional requirements : A process-oriented approach. *IEEE Trans. Software Eng.*, 18(6) :483–497, 1992.

[MFJ05] Pierre-Alain Muller, Franck Fleurey, and Jean-Marc Jézéquel. Weaving executability into object-oriented meta-languages. *SOSYM*, pages 264–278, 2005.

[MH11] John Mylopoulos and Patrick Heymans, editors. *RE 2011, 19th IEEE International Requirements Engineering Conference, Trento, Italy, August 29 2011 - September 2, 2011*. IEEE, 2011.

[Mil95] George A Miller. Wordnet : a lexical database for english. *Communications of the ACM*, 38(11) :39–41, 1995.

[MLHS+11] Raúl Mazo, Roberto Erick Lopez-Herrejon, Camille Salinesi, Daniel Diaz, and Alexander Egyed. Conformance Checking with Constraint Logic Programming : The Case of Feature Models. In *COMPSAC*, pages 456–465. IEEE Computer Society, 2011.

[MMAS11] Gunter Mussbacher, Ana Moreira, Joao Araujo, and Pablo Sanchez, editors. *First Model-Driven Requirements Engineering Workshop, MoDRE 2011, Trento, Italy, August 29, 2011*. IEEE, 2011.

[MMH12] Martin Mahaux, Alistair Mavin, and Patrick Heymans. Choose Your Creativity : Why and How Creativity in Requirements Engineering Means Different Things to Different People. In Björn Regnell and Daniela E. Damian, editors, *REFSQ*, volume 7195 of *Lecture Notes in Computer Science*, pages 101–116. Springer, 2012.

[MN11] Anas Mahmoud and Nan Niu. TraCter : A tool for candidate traceability link clustering. In Mylopoulos and Heymans [MH11], pages 335–336.

[MOA09] Aaron K. Massey, Paul N. Otto, and Annie I. Antón. Prioritizing Legal Requirements. In *RELAW*, pages 27–32. IEEE Computer Society, 2009.

[MOHA10] Aaron K Massey, Paul N Otto, Lauren J Hayward, and Annie I Antón. Evaluating existing security and privacy requirements for legal compliance. *Requirements Engineering*, 15(1) :119–137, 2010.

[MR05] Neil A. M. Maiden and Suzanne Robertson. Integrating Creativity into Requirements Processes : Experiences with an Air Traffic Management System. In Colette Rolland and Joanne Atlee, editors, *RE*, pages 105–116. IEEE Computer Society, 2005.

[MR07] Kannan Mohan and Balasubramaniam Ramesh. Traceability-based knowledge integration in group decision and negotiation activities. *Decision Support Systems*, 43(3) :968–989, 2007.

[MSCHc12] Mehdi Mirakhorli, Yonghee Shin, Jane Cleland-Huang, and Murat Çinar. A tactic-centric approach for automating traceability of quality concerns. In Martin Glinz, Gail C. Murphy, and Mauro Pezzè, editors, *ICSE*, pages 639–649. IEEE, 2012.

[MSOA11] Aaron K Massey, Ben Smith, Paul N Otto, and Annie I Antón. Assessing the accuracy of legal implementation readiness decisions. In Mylopoulos and Heymans [MH11], pages 207–216.

[MWHN09] Alistair Mavin, Philip Wilkinson, Adrian Harwood, and Mark Novak. Easy Approach to Requirements Syntax (EARS). In William N. Robinson and Kevin Ryan, editors, *RE*, pages 317–322. IEEE Computer Society, 2009.

[NE00] Bashar Nuseibeh and Steve M. Easterbrook. Requirements engineering : a roadmap. In Anthony Finkelstein, editor, *ICSE - Future of SE Track*, pages 35–46. ACM, 2000.

[NFTJ03a] Clémentine Nebut, Franck Fleurey, Yves Le Traon, and Jean-Marc Jézéquel. A Requirement-Based Approach to Test Product Families. In Frank van der Linden, editor, *PFE*, volume 3014 of *Lecture Notes in Computer Science*, pages 198–210. Springer, 2003.

[NFTJ03b] Clémentine Nebut, Franck Fleurey, Yves Le Traon, and Jean-Marc Jézéquel. Requirements by Contracts allow Automated System Testing. In *ISSRE*, pages 85–98. IEEE Computer Society, 2003.

[NM12] Nan Niu and Anas Mahmoud. Enhancing candidate link generation for requirements tracing : the cluster hypothesis revisited. In *Requirements Engineering Conference (RE), 2012 20th IEEE International*, pages 81–90. IEEE, 2012.

[NRC11] US Nuclear Regulatory Commission NRC. Recommendations for enhancing reactor safety in the 21st century, the near-term task force review of insights from the fukushima dai-ichi accident. Technical report, NRC, 2011.

[Nut04] William J Nuttall. *Nuclear renaissance : technologies and policies for the future of nuclear power*. CRC Press, 2004.

[OMG10] OMG. Documents Associated With Software Assurance Evidence Metamodel (SAEM) Version 1.0 - Beta 1. http://www.omg.org/spec/SAEM/1.0/Beta1/, 2010.

[OMG12] OMG. Documents Associated With Systems Modeling Language (SysML), Version 1.3. http://www.omg.org/spec/SysML/1.3/, 2012.

[OMG13] OMG. Structured Assurance Case Metamodel (SACM). http://www.omg.org/spec/SACM/, 2013.

[ONR11] UK Office for Nuclear Regulation ONR. Japanese earthquake and tsunami : Implications for the uk nuclear industry, final report. Technical report, ONR, 2011.

[OW04] Stanislaw Osinski and Dawid Weiss. Conceptual Clustering Using Lingo Algorithm : Evaluation on Open Directory Project Data. In Mieczyslaw A. Klopotek, Slawomir T. Wierzchon, and Krzysztof Trojanowski, editors, *Intelligent Information Systems*, Advances in Soft Computing, pages 369–377. Springer, 2004.

[PBBT09] Gilles Perrouin, Erwan Brottier, Benoit Baudry, and Yves Le Traon. Composing Models for Detecting Inconsistencies : A Requirements Engineering Perspective. In Martin Glinz and Patrick Heymans, editors, *REFSQ*, volume 5512 of *Lecture Notes in Computer Science*, pages 89–103. Springer, 2009.

[PG96] Francisco AC Pinheiro and Joseph A Goguen. An object-oriented tool for tracing requirements. *Software, IEEE*, 13(2) :52–64, 1996.

[PKGJ08] Gilles Perrouin, Jacques Klein, Nicolas Guelfi, and Jean-Marc Jézéquel. Reconciling Automation and Flexibility in Product Derivation. In *SPLC*, pages 339–348. IEEE Computer Society, 2008.

[Poh97] Klaus Pohl. *Process-centered requirements engineering*. John Wiley & Sons, Inc., 1997.

[Poh10] Klaus Pohl. *Requirements Engineering - Fundamentals, Principles, and Techniques*. Springer, 2010.

[Pot01] Colin Potts. Metaphors of Intent. In *RE*, pages 31–39. IEEE Computer Society, 2001.

[PWKSB11] Rajwinder Kaur Panesar-Walawege, Torbjørn Skyberg Knutsen, Mehrdad Sabetzadeh, and Lionel Briand. Cresco : Construction of evidence repositories for managing standards compliance. In *Advances in Conceptual Modeling. Recent Developments and New Directions*, pages 338–342. Springer, 2011.

[PWSB11] Rajwinder Kaur Panesar-Walawege, Mehrdad Sabetzadeh, and Lionel C. Briand. A Model-Driven Engineering Approach to Support the Verification of Compliance to Safety Standards. In Tadashi Dohi and Bojan Cukic, editors, *ISSRE*, pages 30–39. IEEE, 2011.

[RAC11] Rehan Rauf, Michal Antkiewicz, and Krzysztof Czarnecki. Logical structure extraction from software requirements documents. In *Requirements Engineering Conference (RE), 2011 19th IEEE International*, pages 101–110. IEEE, 2011.

[Ram02] B. Ramesh. Process knowledge management with traceability. *Software, IEEE*, 19(3) :50–52, 2002.

[RD11] Amine Raji and Philippe Dhaussy. Use Cases for Context Aware Model-Checking. In Jörg Kienzle, editor, *MoDELS Workshops*, volume 7167 of *Lecture Notes in Computer Science*, pages 202–216. Springer, 2011.

[RHW06] WENRA Reactor Harmonization Working Group RHWG. Harmonisation of Reactor Safety in WENRA Countries. Technical report, WENRA, 2006.

[RHW11] WENRA Reactor Harmonization Working Group RHWG. Progress towards harmonisation of safety for existing reactors in wenra countries. Technical report, WENRA, 2011.

[RKH03] Björn Regnell, Lena Karlsson, and Martin Höst. An Analytical Model for Requirements Selection Quality Evaluation in Product Software Development. In *RE*, pages 254–263. IEEE Computer Society, 2003.

[RMTV11] Mikko Raatikainen, Tomi Männistö, Teemu Tommila, and Janne Valkonen. Challenges of requirements engineering - A case study in nuclear energy domain. In Mylopoulos and Heymans [MH11], pages 253–258.

[RR99] Suzanne Robertson and James Robertson. *Mastering the Requirements Process*. Addison-Wesley, 1999.

[RSA98] Colette Rolland, Carine Souveyet, and Camille Ben Achour. Guiding goal modeling using scenarios. *IEEE Trans. Software Eng.*, 24(12) :1055–1071, 1998.

[SAY+12] Krishna Sapkota, Arantza Aldea, Muhammad Younas, David A Duce, and Rene Banares-Alcantara. Extracting meaningful entities from regulatory text : Towards automating regulatory compliance. In *Requirements Engineering and Law (RELAW), 2012 Fifth International Workshop on*, pages 29–32. IEEE, 2012.

[SAYD12] Patrizia Scandurra, Andrea Arnoldi, Tao Yue, and Marco Dolci. Functional requirements validation by transforming use case models into Abstract State Machines. In Sascha Ossowski and Paola Lecca, editors, *SAC*, pages 1063–1068. ACM, 2012.

[SB88] Gerard Salton and Christopher Buckley. Term-weighting approaches in automatic text retrieval. *Information processing & management*, 24(5) :513–523, 1988.

[SB11] Nicolas Sannier and Benoit Baudry. Défis pour la variabilité et la traçabilité des exigences en ingénierie système. In *INFORSID 2011*, Lille, France, May 2011.

[SB12a] Nicolas Sannier and Benoit Baudry. Defining and retrieving themes in nuclear regulations. In Annie Antón, Travis Breaux, Daniel Amyot, and Wade Chumney, editors, *RELAW*, pages 33–41. IEEE, 2012.

[SB12b] Nicolas Sannier and Benoit Baudry. Toward multilevel textual requirements traceability using model-driven engineering and information retrieval. In Mussbacher et al. [MAS12c], pages 29–38.

[SB14] Nicolas Sannier and Benoit Baudry. Acquiring and Analyzing Regulations with a mix MDE-IR approach. Technical report, Inria, 2014. submitted to REFSQ'14.

[SBN11] Nicolas Sannier, Benoit Baudry, and Thuy Nguyen. Formalizing standards and regulations variability in longlife projects. A challenge for Model-driven engineering. In Mussbacher et al. [MMAS11], pages 64–73.

[SFG99] H. Sharp, A. Finkelstein, and G. Galal. Stakeholder identification in the requirements engineering process. In *Database and Expert Systems Applications, 1999. Proceedings. Tenth International Workshop on*, pages 387–391, 1999.

[SGN11] Pete Sawyer, Vincenzo Gervasi, and Bashar Nuseibeh. Unknown knowns : Tacit knowledge in requirements engineering. In Mylopoulos and Heymans [MH11], page 329.

[SJI+12] Alberto Siena, Ivan Jureta, Silvia Ingolfo, Angelo Susi, Anna Perini, and John Mylopoulos. Capturing variability of law with nomos 2. *ER*, 7532 :383–396, 2012.

[SK98] Ian Sommerville and Gerald Kotonya. *Requirements engineering : processes and techniques*. John Wiley & Sons, Inc., 1998.

[SM86] Gerard Salton and Michael J McGill. *Introduction to modern information retrieval*. McGraw-Hill, Inc., New York, NY, USA, 1986.

[SRG02] Peter Sawyer, Paul Rayson, and Roger Garside. REVERE : Support for Requirements Synthesis from Documents. *Information Systems Frontiers*, 4(3) :343–353, 2002.

[SS05] Andrew Stone and Pete Sawyer. Finding tacit knowledge by solving the pre-requirements tracing problem. In *REFSQ*. Citeseer, 2005.

[SSC06] Vibha Sinha, B. Sengupta, and Satish Chandra. Enabling collaboration in distributed requirements management. *Software, IEEE*, 23(5) :52–61, 2006.

[SWY75] Gerard Salton, Anita Wong, and Chung-Shu Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18(11) :613–620, 1975.

[URMT11] Eero J. Uusitalo, Mikko Raatikainen, Tomi Männistö, and Teemu Tommila. Structured natural language requirements in nuclear energy domain towards improving regulatory guidelines. In *RELAW*, pages 67–73. IEEE, 2011.

[VCd88] Alain Villemeur, Paul Caseau, and Arnould d'Harcourt. *Sûreté de fonctionnement des systèmes industriels : fiabilité, facteurs humains, informatisation*. Eyrolles, 1988.

[VCMÁ07] Cristina Vicente-Chicote, Begoña Moros, and José Ambrosio Toval Álvarez. REMM-Studio : an Integrated Model-Driven Environment for Requirements Specification, Validation and Formatting. *Journal of Object Technology*, 6(9) :437–454, 2007.

[vL08] Axel van Lamsweerde. Requirements engineering : from craft to discipline. In Mary Jean Harrold and Gail C. Murphy, editors, *SIGSOFT FSE*, pages 238–249. ACM, 2008.

[vL09] Axel van Lamsweerde. *Requirements Engineering - From System Goals to UML Models to Software Specifications*. Wiley, 2009.

[VPW13] Jose Luis de la Vara and Rajwinder Kaur Panesar-Walawege. Safetymet : A metamodel for safety standards. Technical report, Simula Research Laboratory, 2013. Paper submitted at MODELS 2013.

[VRRP80] Cornelis J Van Rijsbergen, Stephen Edward Robertson, and Martin F Porter. *New models in probabilistic information retrieval*. Computer Laboratory, University of Cambridge, 1980.

[WCR10] Krzysztof Wnuk, David Callele, and Björn Regnell. Guiding Requirements Scoping Using ROI : Towards Agility, Openness and Waste Reduction. In Didar Zowghi and Jane Cleland-Huang, editors, *RE*, pages 409–410. IEEE Computer Society, 2010.

[WNGF09] Westley Weimer, ThanhVu Nguyen, Claire Le Goues, and Stephanie Forrest. Automatically finding patches using genetic programming. In *ICSE*, pages 364–374. IEEE, 2009.

[WP10] Stefan Winkler and Jens Pilgrim. A survey of traceability in requirements engineering and model-driven development. *Software and Systems Modeling (SoSyM)*, 9(4) :529–565, 2010.

[WRK09] Krzysztof Wnuk, Björn Regnell, and Lena Karlsson. What Happened to Our Features ? Visualization and Understanding of Scope Change Dynamics in a Large-Scale Industrial Setting. In William N. Robinson and Kevin Ryan, editors, *RE*, pages 89–98. IEEE Computer Society, 2009.

[WRS09] Krzysztof Wnuk, Björn Regnell, and Claes Schrewelius. Architecting and Coordinating Thousands of Requirements - An Industrial Case Study. In Martin Glinz and Patrick Heymans, editors, *REFSQ*, volume 5512 of *Lecture Notes in Computer Science*, pages 118–123. Springer, 2009.

[WSB+09] Jon Whittle, Peter Sawyer, Nelly Bencomo, Betty H. C. Cheng, and Jean-Michel Bruel. RELAX : Incorporating Uncertainty into the Specification of Self-Adaptive Systems. In William N. Robinson and Kevin Ryan, editors, *RE*, pages 79–88. IEEE Computer Society, 2009.

[WSB+10] Jon Whittle, Pete Sawyer, Nelly Bencomo, Betty H.C. Cheng, and Jean-Michel Bruel. Relax : a language to address uncertainty in self-adaptive systems requirement. *Requirements Engineering*, 15(2) :177–196, 2010.

[YA10] Jessica D. Young and Annie I. Antón. A Method for Identifying Software Requirements Based on Policy Commitments. In Didar Zowghi and Jane Cleland-Huang, editors, *RE*, pages 47–56. IEEE Computer Society, 2010.

[YBL11] Tao Yue, Lionel C Briand, and Yvan Labiche. A systematic review of transformation approaches between user requirements and analysis models. *Requirements Engineering*, 16(2) :75–99, 2011.

[YBL13] Tao Yue, Lionel C Briand, and Yvan Labiche. Facilitating the transition from use case models to analysis models : Approach and experiments. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, 22(1) :5, 2013.

[YRG+12] Hui Yang, Anne N. De Roeck, Vincenzo Gervasi, Alistair Willis, and Bashar Nuseibeh. Speculative requirements : Automatic detection of uncertainty in natural language requirements. In Mats Per Erik Heimdahl and Pete Sawyer, editors, *RE*, pages 11–20. IEEE, 2012.

[Yu97] Eric SK Yu. Towards modelling and reasoning support for early-phase requirements engineering. In *Requirements Engineering, 1997., Proceedings*

of the Third IEEE International Symposium on, pages 226–235. IEEE, 1997.

[Zav97] Pamela Zave. Classification of research efforts in requirements engineering. *ACM Comput. Surv.*, 29(4) :315–321, December 1997.

[ZBL07] Gregory Zoughbi, Lionel C. Briand, and Yvan Labiche. A UML Profile for Developing Airworthiness-Compliant (RTCA DO-178B), Safety-Critical Software. In Gregor Engels, Bill Opdyke, Douglas C. Schmidt, and Frank Weil, editors, *MoDELS*, volume 4735 of *Lecture Notes in Computer Science*, pages 574–588. Springer, 2007.

[ZBL11] Gregory Zoughbi, Lionel C. Briand, and Yvan Labiche. Modeling safety and airworthiness (RTCA DO-178B) information : conceptual model and UML profile. *Software and System Modeling*, 10(3) :337–367, 2011.

[ZMZ05] Wei Zhang, Hong Mei, and Haiyan Zhao. A Feature-Oriented Approach to Modeling Requirements Dependencies. In Colette Rolland and Joanne Atlee, editors, *RE*, pages 273–284. IEEE Computer Society, 2005.

Table des figures

Liste des tableaux

Résumé

Les systèmes de contrôle-commande importants pour la sûreté de fonctionnement doivent répondre à un certain nombre d'exigences, au premier rang desquelles se trouvent les exigences réglementaires, édictées par les autorités nationales et complétées par un ensemble de recommandations pratiques et de textes normatifs. Les exigences de ce domaine sont peu formalisées, les relations de traçabilité, et par conséquent l'organisation des exigences de ce vaste domaine est souvent implicite. Enfin, les passerelles entre contextes nationaux différents sont très peu développées.

Les travaux de cette thèse se situent dans ce contexte industriel en partenariat avec EDF R&D et au sein du projet CONNEXION regroupant les acteurs majeurs du contrôle-commande nucléaire français. Les contributions de la thèse s'articulent autour de l'approche INCREMENT (Intrumentation aNd Control regulatory REquirement Modeling Environment) qui adresse les deux premiers challenges présentés, et en particulier : (1) **la formalisation du domaine** où nous proposons à la fois une description du domaine et un métamodèle permettant une capitalisation et une vue globale d'un référentiel d'exigences, (2) **une base outillée** pour l'acquisition automatique de documents, un environnement graphique pour la manipulation de modèles et l'apport de techniques de recherche d'information pour la traçabilité des exigences, (3) une approche originale avec une **hybridation entre modélisation et recherche d'information** pour une amélioration de la traçabilité des exigences.

Le métamodèle proposé et ses outils sont utilisés dans l'industrie dans le projet CONNEXION. Notre approche hybride a permis dans nos expérimentations de réduire, en moyenne, la taille de ces espaces de 65% comparé aux approches standard de recherche d'information, sans en dégrader le contenu.

Abstract

Instrumentation and Control (I&C) Systems important to safety must conform to their requirements, where regulatory requirements are first class entities, written by national safety authorities and completed using a set of national recommendation guides or standards. The global domain knowledge is scattered, not formalized and traceability links and the organization within the domain are implicit. Bridges between different national practices are not developed, whereas the understanding of requirements and practices variability concerns becomes a significant industrial issue.

The thesis sets up in an industrial context with EDF R&D and the CONNEXION project that gathered the French nuclear I&C industry. Its contributions are defined around the INCREMENT approach (Instrumentation aNd Control Regulatory Requirement Modeling Environment) that addresses the two first challenges previously introduced. In particular, they consist in : (1) **the domain formalization** itself by the proposal of a metamodel that allows a high level capitalization of a requirements corpus as well as its organization, (2) **a tool-support basis** to gather partial knowledge from the textual documents, manipulate such models that conform to the proposed metamodel, and Information retrieval techniques to support better requirements traceability, (3) the proposal of an original hybrid approach, **mixing both metamodeling and information retrieval**, and combine them in a mutual beneficial joint use.

The metamodel and its tool support are used in the industrial context of the CONNEXION project. Where information retrieval techniques for requirements traceability suffer from large sets of false positives limitations, our hybrid approach allowed us to reduce this noise and the size of the candidate links research space by a mean of 65% without decreasing their global quality.