UNIVERSITÉ DE NICE - SOPHIA ANTIPOLIS

ÉCOLE DOCTORALE STIC
SCIENCES ET TECHNOLOGIES DE L'INFORMATION
ET DE LA COMMUNICATION

T H È S E

pour obtenir le titre de

Docteur en Sciences
de l'Université de Nice - Sophia Antipolis

Mention : **Informatique**
préparée au laboratoire **I3S**

Présentée et soutenue publiquement par

Khalil Riad BOUZIDI

le 11 septembre 2013

Titre:

Aide à la création et à l'exploitation de réglementations basée sur les modèles et techniques du web sémantique

Directeur de thèse: Catherine FARON-ZUCKER
Co-directeur de thèse: Nhan LE THANH

Jury

M. Gilles Kassel, Professeur, Université de Picardie, Rapporteur
M. Habib Abdulrab, Professeur, INSA de Rouen, Rapporteur
M. Bruno Fies, Ingénieur R&D, CSTB, Examinateur
Mme. Catherine Faron-Zucker, Maître de conférences, UNSA, Co-directeur
M. Nhan Le Thanh, Professeur, UNSA, Co-directeur

À mes parents, mon frére et ma soeur

REMERCIEMENTS

Il m’est particulièrement agréable de pouvoir remercier ici Mme. Catherine FARON-ZUCKER directrice de cette thèse, pour avoir bien voulu m’accueillir dans son équipe de recherche et m’avoir encadré durant ces trois années. Je lui exprime ma profonde gratitude. Sa gentillesse et ses encouragements me furent d’un constant appui sans lequel la persévérance m’aurait manqué. Je tiens aussi à remercier M. Nhan LE-THANH, Professeur à l’université de Nice et mon co-directeur de thèse, pour sa disponibilité et son soutien et d’avoir permis que ma thèse se déroule dans les meilleures conditions.

Je remercie mon responsable CSTB M. Bruno FIES de m'avoir fait confiance et pour les conseils qu'il m'a donnés tout au long de ce doctorat. Sa très bonne gestion des moments difficiles de la thèse m’a beaucoup aidé à avancer. Je dois dire que son expérience, sa sagesse et ses qualités humaines très appréciables ont joué un grand rôle dans l'aboutissement de la thèse.

Je remercie M. Gilles KASSEL, Professeur à l’université de Picardie Jules Verne ainsi que M. Habib ABDULRAB, Professeur à l’INSA de Rouen pour avoir accepté de rapporter ce manuscrit, ainsi que pour l’intérêt qu’ils ont manifestés à l’égard de ce travail de thèse.

Je remercie chaleureusement mes parents Fatima et Rachid, mon frère Ali et ma sœur Nesrine pour leurs soutiens tout au long de ces années d’études.

Je remercie les membres du CSTB, Centre Scientifique et Technique du Bâtiment, pour cette opportunité unique de travailler dans un contexte entièrement industriel, en particulier M. Patrick MORAND et M. Alain ZARLI qui m’ont accueilli au sein du CSTB Sophia Antipolis.

Un grand merci à Fabien GANDON et Olivier CORBY pour avoir été disponible pendant les moments difficiles de cette thèse. Leurs expériences scientifiques exceptionnelles ont joué un grand rôle dans l’avancement de mes travaux de recherche.

Je remercie Marc BOURDEAU pour ses conseils judicieux et pour les différentes corrections et relecture d’articles scientifiques.

Je remercie mon ami, collègue et voisin de bureau Franck BERTHELON avec lequel j’ai partagé des moments inoubliables.

Je remercie Damien LEGRAND, Nicolas MARIE et Pavel ARAPOV pour les agréables discussions.

Mes sincères remerciements aux membres de l'équipe WIMMICS: Michel BUFFA, Alain GIBOIN, Peter SANDER, Alain GIBOIN, Serena VILLATA, Andrea TETTAMANZI, Isabelle MIRBEL et bien sûr à mes collègues doctorants : Luca COSTABELLO, Maxime LEFRANCOIS, Oumy SEYE, Zide MENG, Corentin FOLLENFANT, Nguyen THI HOA HUE.

Mes sincères remerciements aux membres du département TIDS : Éric GADUEL, Joëlle CARDOSO LUCAS, Barbara CERRET, Émilie THIBAULT, Christiane PUJOL, Ahmed LAAROUSSI, Khaled ATHAMENA, Alexis BOISSONNAT, Régis DECORME, Bruno Hilaire,

Je remercie Amine MANAA, Pierre BENEDICTO, Vincent CARACCIO, Samuel Fayolle, Cherif KHIRISSI, Myriam MEBARKI, Kahina BOUMAZA, Hocine BOULEDROUA, Seif NEHOUCHI, Bruno CAVALCANTE-SILVA et Théo SHERRER de m'avoir soutenue, chacun à sa manière, pendant cette thèse.

Je remercie enfin toutes les personnes que j'ai oublié de remercier ici.

RÉSUMÉ

Les réglementations concernant l'industrie de la construction deviennent de plus en plus complexes et touchent plus d'un domaine à la fois. Elles portent sur les produits, les composants et l'exécution des projets. Elles jouent aussi un rôle important pour garantir la qualité d'un bâtiment, ses caractéristiques et minimiser son impact environnemental. Depuis 30 ans, le CSTB prouve son savoir-faire en la matière au travers du développement du REEF, l'encyclopédie complète des textes techniques et réglementaires de la construction. Dans le cadre d'une collaboration entre le CSTB et le laboratoire I3S, nous avons travaillé à la formalisation et au traitement automatisé des informations technico-réglementaires contenues dans le REEF.

Nous avons mis en œuvre notre approche pour aider à la création de nouveaux Avis Techniques. Il s'agit de préciser comment ils sont rédigés et comment standardiser leur structure grâce à la mise en œuvre de services sémantiques adaptés. Nous avons réussi à identifier et à comprendre les problèmes liés à la rédaction d'avis techniques et nous nous sommes focalisés sur le renseignement des dossiers techniques par les industriels. Nos contributions sont les suivantes :

Premièrement, nous avons construit manuellement une ontologie du domaine, qui définit les principaux concepts impliqués dans l'élaboration des Avis Technique. Cette ontologie appelée "OntoDT" est couplée avec le thésaurus du projet REEF. Nous l'avons définie à partir de l'étude des dossiers techniques existants, du thesaurus REEF et en interviewant les instructeurs du CSTB.

Deuxièmement, notre contribution consiste en l'utilisation des standard SBVR (Semantics of Business Vocabulary and Business Rules) et SPARQL pour reformuler, d'une part dans un langage contrôlé et d'autre part dans un langage formel, les contraintes réglementaires présentes dans les GP. SBVR représente une assurance de la qualité du texte des contraintes réglementaires présentées à l'utilisateur et SPARQL permet l'automatisation de la vérification de ces contraintes. Ces deux représentations reposent sur l'ontologie de domaine que nous avons développée.

Troisièmement, notre modèle intègre des connaissances expertes sur le processus même de vérification des dossiers techniques. Nous avons organisé en différents processus les requêtes SPARQL représentant des contraintes réglementaires. A chaque composant intervenant dans

un dossier technique correspond un processus de vérification de sa conformité à la réglementation en vigueur. Un processus élémentaire contient la suite des requêtes SPARQL à exécuter pour vérifier la conformité d'un composant élémentaire. Le processus complexe de vérification d'un dossier technique est défini récursivement et construit automatiquement. Les processus sont représentés de manière déclarative en RDF et un moteur de processus interprète ces descriptions RDF pour ordonner et déclencher l'exécution des requêtes nécessaires à la vérification d'un dossier technique particulier.

Enfin, nous représentons en RDF l'association d'une règle SBVR et d'une requête SPARQL représentant une même contrainte et nous utilisons ces annotations pour produire à l'utilisateur un rapport de conformité en langue naturelle pour l'assister dans la rédaction d'un avis technique.

Un outil nous a permis de valider notre approche. Nous l'avons soumis à une évaluation par les instructeurs du CSTB qui ont validé à la fois le scénario suivi ainsi que la granularité des informations demandées.

ABSTRACT

Regulations in the Building industry are becoming increasingly complex and involve more than one technical area. They cover products, components and project implementation. They also play an important role to ensure the quality of a building, and to minimize its environmental impact. For more than 30 years, CSTB has proved its expertise in this field through the development of the complete encyclopaedia of French technical and regulatory texts in the building domain: the REEF. In the framework of a collaboration between CSTB and the I3S laboratory, we are carrying on research on the acquisition of knowledge from the technical and regulatory information contained in the REEF and the automated processing of this knowledge with the final goal of assisting professionals in the use of these texts and the creation of new texts. We are implementing this work in CSTB to help industrials in the writing of Technical Assessments. The problem is how to specify these assessments and standardize their structure using models and adaptive semantic services.

The research communities of Knowledge Engineering and Semantic Web play a key role in providing the models and techniques relevant for our research, whose main objective is to simplify access to technical regulatory information, to support professionals in its implementation, and to facilitate the writing of new regulations while taking into account constraints expressed in the existing regulatory corpus. We focus on Technical Assessments based on technical guides capturing both regulations and knowledge of CSTB experts when producing these documents.

A Technical Assessment (in French: Avis Technique or ATec) is a document containing technical information on the usability of a product, material, component or element of construction, which has an innovative character. We chose this Technical Assessment as a case study because CSTB has the mastership and a wide experience in these kinds of technical documents.

We are particularly interested in the modeling of the regulatory constraints derived from the Technical Guides used to validate the Assessment. These Guides are regulatory complements offered by CSTB to the various industrials to enable easier reading of technical regulations. They collect execution details with a wide range of possible situations of implementations.

Our work aims to formalize the Technical Guides in a machine-processable model to assist the creation of Technical Assessments by automating their validation.

For this purpose, we first constructed a domain-ontology, which defines the main concepts involved in the Technical Guides. This ontology called “OntoDT” is coupled with domain thesauri. Several are being developed at CSTB among which one seems the most relevant by its volume and its semantic approach: the thesaurus from the REEF project.

Our second contribution is the use of standard SBVR (Semantics of Business Vocabulary and Business Rules) and SPARQL to reformulate the regulatory requirements of guides both in a controlled and formal language

Third, our model incorporates expert knowledge on the verification process of Technical Documents. We have organized the SPARQL queries representing regulatory constraints into several processes. Each component involved in the Technical Document corresponds to a elementary process of compliance checking. An elementary process contains a set of SPARQL queries to check the compliance of an elementary component. A full complex process for checking a Technical Document is defined recursively and automatically built as a set of elementary processes relative to the components which have their semantic definition in OntoDT.

Finally, we represent in RDF the association between the SBVR rules and SPARQL queries representing the same regulatory constraints. We use annotations to produce a compliance report in natural language to assist users in the writing of Technical Assessments.

TABLE DES MATIERES

INTRODUCTION 1
CHAPITRE 1 ETAT DE L'ART 11
1 Introduction 12
2 Modélisation des textes réglementaires 12
2.1 Analyse automatique des réglementations 13
2.2 Analyse manuelle des réglementations 14
2.3 Positionnement 17
3 Contrôle de conformité dans le domaine du bâtiment 18
3.1 Les travaux sur le contrôle de conformité à l'université de Standford 18
3.2 Projet CORENET 20
3.3 Projet AC3 « SMART code » 20
3.4 Projet HITOS 21
3.5 Modèle C3R 22
3.6 Positionnement 23
4 Ontologies et langages de représentation formelle 24
4.1 Classifications des ontologies 26
4.2 Les bonnes pratiques pour la construction d'ontologies 28
4.3 Langages de représentation des connaissances 30
5 Modélisation de règles 35
5.1 SBVR 35
5.2 SPARQL et SPIN 37
5.3 Positionnement 39
6 Modélisation de processus à base d'ontologies 39
6.1 BWW (Bunge-Wand-Weber) 40
6.2 UEML (Unified Enterprise Modeling Language) 40
6.3 Le projet MIT process handbook 41
6.4 TOVE (TOronto Virtual Enterprise) 41
6.5 PSL (Process Specification Language) 42
6.6 PIF (Process Interchange Format) 43
6.7 POP (Process, Organization, Product & others) 44

6.8 Ontologie des processus de propagation des maladies infectieuses 45
6.9 SAWSDL, OWL-S et WSMO 46
6.10 Positionnement 46
7 Conclusion 47
CHAPITRE 2 MODÉLISATION DU PROCESSUS DE RÉDACTION DES DOSSIERS TECHNIQUES 49
1 Processus courant de rédaction d'un dossier technique 50
2 Modélisation du dossier technique 52
2.1 Analyse des besoins 53
2.2 Collecte des données 54
2.3 Création d'une ontologie légère 56
2.4 Définition de concepts 58
2.5 Résultat 60
3 Modélisation du processus de rédaction d'un dossier technique 61
3.1 Identification des étapes du processus de rédaction d'un dossier technique 61
3.2 Modélisation de la dynamique du processus de rédaction d'un dossier technique 62
4 Conclusion 66
CHAPITRE 3 MODELISATION DES GUIDES PRATIQUES 67
1 Introduction 68
2 Modélisation des contraintes réglementaires 71
2.1 Ontologie OntoDT 71
2.2 Modélisation des contraintes réglementaires sous la forme de règles SBVR 74
2.3 Modélisation des contraintes réglementaires en RDF et SPARQL 79
3 Modélisation du processus de vérification des contraintes réglementaires 83
3.1 Modèle de processus 84
3.2 Processus élémentaire de vérification de contraintes 85
3.3 Processus complexe de vérification de contraintes 86
4 Résultats et limitations 87
5 Conclusion 89
CHAPITRE 4 MODELE D'AIDE A LA REDACTION D'AVIS TECHNIQUES 91
1 Introduction 92
2 Automatisation de la vérification de la conformité des dossiers techniques 93
2.1 Orchestration des vérifications de contraintes par un moteur de processus 93
2.2 Vérification des contraintes par interrogation de la représentation du Dossier Technique 95
3 Aide à la prise de décision 95
3.1 Explication du résultat des vérifications automatiques 96

3.2 Aide à la vérification manuelle de contraintes 98
4 Conclusion 100
CHAPITRE 5 EVALUATION DU SYSTEME D'AIDE A LA REDACTION D'AVIS TECHNIQUE 103
1 Introduction 104
2 Spécifications 104
2.1 Spécifications fonctionnelles 104
2.2 Spécifications techniques 112
3 Conception 113
3.1 Modèle MVC global 113
3.2 Modèle MVC de nos deux outils 115
4 Validation et évaluation 118
4.1 Validation des représentations des contraintes réglementaires 118
4.2 Validation de l'ontologie du Document Technique (OntoDT) 120
4.3 Evaluation et validation du processus de vérification de conformité 122
4.4 Evaluation et validation des systèmes d'aide à la rédaction de dossiers techniques et d'avis techniques 123
5 Conclusion 124
CONCLUSION GENERALE ET PERSPECTIVES 127
BIBLIOGRAPHIE 131

TABLE DES FIGURES

Figure 1 Architecture des langages de représentation des connaissances du W3C 31
Figure 2 Exemple d'une règle représentée en SPARQL 38
Figure 3 Règle SPARQL (Figure 2) représentée en SPIN 38
Figure 4 Modules PSL pour les classes génériques d'activités et leurs relations 42
Figure 5 Les classes et relations PIF 43
Figure 6 Méta-modèle POP 44
Figure 7 Processus de création d'un avis technique 51
Figure 8 Etapes de construction d'une ontologie du dossier technique 53
Figure 9 Extrait du thesaurus REEF 56
Figure 10 Définition de la classe VerrePolymere dans l'éditeur Protege 59
Figure 11 Définition OWL de la classe VerrePolymere 60
Figure 12 Outil d'aide à la rédaction de Dossier Technique 63
Figure 13 Extrait du patron de requête SPARQL pour interroger l'ontologie OntoDT sur les classes définies 63
Figure 14 Instanciation du patron de requête SPARQL pour interroger la définition de VerrePolymere 64
Figure 15 Extrait du résultat de requête SPARQL 64
Figure 16 Extrait de la représentation RDF/N3 d'un dossier technique produite par l'outil d'aide à la rédaction de dossier technique 65
Figure 17 Organisation des guides pratiques 70
Figure 18 Caractéristiques d'une tuile 71
Figure 19 Types de tuiles 71
Figure 20 De la classe des tuiles plates en terre cuite dans l'éditeur Protegé 73
Figure 21 Définition OWL de la classe des TuilePlateTerreCuite 74
Figure 22 Extrait du guide pratique des couvertures en tuiles 78
Figure 23 Représentation RDF d'une règle opérative 80
Figure 24 Transformation des règles SBVR en requêtes SPARQL 81
Figure 25 Exemple d'association d'une règle SBVR et des requêtes SPARQL qui l'opérationnalisent 85
Figure 26 Processus complexe de vérification de contraintes réglementaires 86
Figure 27 Processus complexe de vérification de la conformité d'un verre polymère 87
Figure 28 Patron de requête SPARQL permettant de retrouver les composants d'un processus et leurs types 94
Figure 29 Patron de requête SPARQL pour retrouver les conséquents d'une règle 95
Figure 30 Exemple d'un dossier technique conforme à une contrainte élémentaire 97
Figure 31 Exemple d'un rapport de conformité 98
Figure 32 Extrait de la représentation RDF d'une contrainte de mise en œuvre 98
Figure 33 Patron de requête SPARQL d'extraction des règles de conformité 99
Figure 34 Patron de requête SPARQL d'extraction des représentations SBVR des règles 99
Figure 35 Exemple de rapport de conformité (suite) 100
Figure 36 Aide à la rédaction d'un avis technique 101
Figure 37 Principe de la demande d'avis technique sur un produit ou procédé 105
Figure 38 Interface d'accueil industriel de l'outil d'aide à la rédaction d'Avis Techniques 106
Figure 39 Exemple de formulaire 107
Figure 40 Interface de fin de création d'un Dossier Technique 108
Figure 41 Rapport automatique de conformité 111

Figure 42 Rapport de conformité 112
Figure 43 Modèle global MVC 114
Figure 44 Algorithme du moteur de processus 116
Figure 45 Modèle MVC appliqué pour nos outils..... 117

INTRODUCTION

Le Grenelle de l'environnement génère un surcroît d'informations technico-réglementaires dont l'analyse est perçue par les professionnels comme une charge de plus en plus importante. De nombreux et rapides changements des réglementations et documents techniques vont intervenir afin de tenir l'engagement pris principalement en ce qui concerne l'optimisation des aspects énergétiques du Bâtiment. À ce titre, le Centre Scientifique et Technique du Bâtiment (CSTB)[1], avec d'autres acteurs, ont lancé le programme « Grenelle Bâtiment 2012» dont l'objectif est de faire évoluer les règles de l'art. Cette évolution réglementaire doit se faire dans le cadre d'une approche globale qui prenne en compte d'une part la complexité du secteur du bâtiment, la pluralité des usages et des besoins, et d'autre part les réglementations existantes dans un souci de cohérence et de non-contradiction.

D'autre part, les réglementations concernant l'industrie de la construction deviennent de plus en plus complexes et touchent plus d'un domaine à la fois. Elles portent sur les produits, les composants et l'exécution des projets. Elles jouent aussi un rôle important pour garantir la qualité d'un bâtiment, ses caractéristiques et minimiser son impact environnemental.

Depuis 30 ans, le CSTB prouve son savoir-faire en la matière au travers du développement du REEF[2], l'encyclopédie complète des textes techniques et réglementaires de la construction. C'est dans ce contexte général que s'inscrivent nos travaux de recherche où la communauté d'ingénierie des connaissances a un rôle important à jouer, pour proposer des modèles et techniques permettant de simplifier l'accès à l'information technico-réglementaire, faciliter son appropriation, accompagner les professionnels dans sa mise en œuvre, faciliter l'écriture de règlementations en tenant compte des contraintes exprimées dans les corpus règlementaires existants et identifier les façons selon lesquelles on peut apporter une aide dans ce domaine grâce à la mise en œuvre de services sémantiques adaptés.

[1] http://www.cstb.fr/
[2] http://reef.cstb.fr/

Parmi les règles de construction, on distingue les textes officiels et les textes techniques. Les textes officiels sont soit à caractère obligatoire soit à caractère informatif.

- Les textes officiels à caractère obligatoire regroupent les lois, décrets, arrêtés et codes. Les lois sont publiées au Journal Officiel, sont émises par le gouvernement, adaptées par les ministres et votées par les deux assemblées. Elles sont rarement techniques et décrivent plutôt l'approche générale pour le bon fonctionnement d'un bâtiment. Les décrets, également publiés au Journal Officiel, sont définis par un ministère et détaillent l'application et l'exécution de certaines lois. Les arrêtés visent à l'exécution d'un décret ou d'une loi, ils sont définis par un ou plusieurs ministères et délivrés par différentes autorités administratives : arrêtés préfectoraux, arrêtés municipaux. Enfin, les codes regroupent lois, décrets et arrêtés relatifs à un domaine. Il existe plusieurs codes relatifs au domaine de la construction, par exemple, le Code de la Construction et de l'habitation, le Code de l'Urbanisme, le Code du Travail, le Code de la Santé publique, etc.
- Les textes officiels à caractère informatif regroupent les circulaires et les questions écrites. Les circulaires regroupent des directives, des instructions et des notes délivrées par un ou plusieurs ministères et publiées dans le Bulletin Officiel. Les questions écrites sont posées par des députés ou des sénateurs et adressées aux ministères. Les réponses à ces questions sont publiées au Journal Officiel.
- Les textes techniques regroupent les textes concernant des produits, des procédures, des équipements et des règles de mise en œuvre. Il s'agit principalement des normes (NF, EN, ISO), Avis Techniques (ATec), Documents Techniques d'Application (DTA), Appréciations Techniques d'Expérimentation (ATEx) et Documents Techniques Unifiés (DTU).
    - Les normes définissent des solutions aux problèmes techniques concernant des produits, des services et des procédures. Elles représentent des garanties sur leur qualité et leur utilisation efficace.
    - Les ATec sont des avis autorisés sur des méthodes de construction, des matériaux ou des produits, nouveaux ou innovants et non encore normalisés.
    - Les DTA décrivent des recommandations sur l'application technique d'un produit / matériel / procédure, selon une ATec.
    - Les ATEx sont des procédures rapides visant à évaluer si des DTA peuvent être développées pour un produit innovant / matériel / procédure.

- Les DTU sont des cahiers des charges « *types* » pour les travaux. Ils sont utilisés comme des références pour l'établissement des clauses contractuelles pour la réalisation d'un ouvrage donné. Ils ont le statut de norme (NF DTU) et sont élaborés par des commissions de normalisation sous le contrôle général de l'AFNOR et publiés par le CSTB.

Comme cela apparaît dans la nomenclature simplifiée décrite ici, plusieurs acteurs interagissent à l'élaboration des règles de construction. Celles-ci sont émises par des organismes de normalisation tels que l'AFNOR ou par les ministères. Le CSTB joue un rôle majeur dans l'élaboration des textes techniques : il réunit les commissions d'étude qui travaillent à l'élaboration d'Avis Techniques en appliquant les normes sur les dossiers de produits ou procédés fournis par les industriels.

Le CSTB joue également un rôle majeur dans la diffusion de l'ensemble des règlementations. Il édite différents guides à l'usage des professionnels de la construction, dont notamment *Les règles de construction* (Pan et al., 2008) qui a pour vocation d'aider les professionnels à mieux comprendre l'ensemble des règles de construction afin de mieux les appliquer ainsi que les Guides Pratiques qui constituent un complément réglementaire offert par le CSTB aux différents industriels pour leur permettre une lecture plus facile des règlementations techniques de construction. Ils recueillent des détails d'exécution présentant un large éventail de situations possibles de mise en œuvre.

Dans le cadre d'une collaboration entre le CSTB et le laboratoire I3S de l'Université Nice Sophia Antipolis, notre travail de thèse vise à la formalisation et au traitement automatisé des connaissances contenues dans les corpus réglementaires, pour assister les professionnels dans la création de nouveaux textes réglementaires.

Plus précisément nous mettons en œuvre notre approche au CSTB pour aider à la rédaction d'Avis Techniques. Il s'agit de préciser comment ils sont rédigés et comment standardiser leur structure grâce à la mise en œuvre de modèles et des services sémantiques adaptés.

Le CSTB délivre sur mandat des industriels, des Avis Techniques sur les produits de construction mettant en œuvre des procédés non traditionnels. Ces avis techniques offrent une garantie de préservation des performances d'un produit une fois mis en œuvre. La bibliothèque d'Avis Techniques constituée par le CSTB permet ainsi d'orienter les professionnels dans le choix de leurs produits.

En enquêtant auprès des groupes spécialisés en charge de délivrer les Avis Techniques, nous avons pu définir quelques scénarios « *types* » définissant les différentes étapes suivies par les professionnels afin de valider un produit et acquérir un ATec.

Nous avons choisi les Avis Techniques comme modèle d'étude pour les raisons suivantes :

- ils font partie des textes techniques et sont un aspect de notre problématique qui reste très vaste (réglementation des ERP, accessibilité, sécurité incendie, etc.) ;
- le CSTB a une maîtrise et une large expérience des Avis Techniques, ce qui nous permet d'étendre notre champ d'expérimentation ;
- nous avons pu conduire des interviews sur le site de Sophia-Antipolis auprès d'experts impliqués directement dans la rédaction des Avis Techniques.

Par ailleurs, l'étude des avis technique a démontré qu'ils sont constitués de trois parties :

- la page de présentation et d'identification de l'Avis Technique,
- l'Avis Technique proprement dit, formulé par le groupe Spécialisé,
- Le Dossier Technique du produit ou procédé établi sous la responsabilité du détenteur de l'Avis.

La page de couverture rassemble divers éléments, à caractère administratif, constituant ce que l'on peut appeler un état signalétique de l'Avis Technique. Ces éléments sont les suivants :

- le numéro de l'Avis Technique (GS / Millésime (année) – N° d'ordre),
- la dénomination de la famille de produits à laquelle est rattaché le produit (ou procédé), en trois langues (Français, Anglais, Allemand),
- la dénomination commerciale du produit,
- l'identification et les coordonnées du titulaire de l'Avis Technique,
- l'identification du Groupe Spécialisé qui a formulé l'Avis,
- la date d'enregistrement de l'Avis Technique,
- divers renseignements relatifs à l'édition de l'Avis et à l'éditeur (CSTB).

L'avis Technique proprement dit, formulé par le groupe Spécialisé, comporte trois parties :

- une définition succincte,
- L'avis, les conclusions du Groupe Spécialisé et le délai de sa validité,
- des remarques complémentaires du Groupe Spécialisé.

Le Dossier Technique du produit ou procédé établi sous la responsabilité du détenteur de l'Avis Technique est donc fourni par les industriels qui travaillent sur un nouveau produit ou

procédé ou matériau, afin de proposer ce produit, innovant, aux experts du CSTB. Pour être valide d'un point de vue formel, le Dossier Technique doit contenir les sections suivantes :

- Une description détaillée du procédé et du domaine d'emploi visé par le demandeur. C'est une partie de présentation où l'industriel explique le domaine d'emploi du produit ainsi que sa dénomination commerciale et décrit les caractéristiques générales du produit ainsi que les composants qui rentrent dans sa fabrication.
- Les prescriptions de mise en œuvre qui peuvent être selon les cas :
    - spécifiques au procédé,
    - identiques à celles déjà utilisées dans le domaine. Dans ce cas, le Dossier Technique fait référence à des dispositions figurant dans les DTU,
    - communes à une famille de produits sous Avis Technique. Dans ce cas, le Dossier Technique renvoie à un compte rendu de mise en œuvre commun à cette famille.
- Des résultats expérimentaux. Dans cette partie, on explique tous les essais qui on été réalisés par l'industriel pour valider son produit ; ces essais, fournis à l'appui de la demande ou obtenus pendant l'instruction, sont conformes aux normes européennes ou françaises de construction et sont obligatoires pour la validation du dossier technique.
- Des références. On cite dans cette partie des cas d'utilisation et de déploiement du produit.

Pour mieux étudier les Dossiers Techniques nous nous sommes rapprochés des experts concernés. Nous avons notamment organisé trois réunions :

- La première, avec les expert du CSTB Sophia Antipolis afin de nous aider à comprendre la démarche liée à l'élaboration d'un Dossier Technique en bonne et due forme.
- La deuxième, à titre préalable, avec des documentalistes du CSTB pour nous présenter les différents types de réglementation technique. À l'issue de cette réunion, nous avons pu comprendre la différence entre ATec, DTU, DTA, etc. ainsi que le rôle et l'utilisation de chacune. Aussi, on a pu déterminer la place que prennent les Avis Techniques dans la grande famille des textes réglementaires et leurs différentes utilisations.

- La troisième, avec les instructeurs de CSTB spécialisés dans les Avis Techniques, du domaine photovoltaïque. Ils nous ont expliqué les étapes de rédaction d'un Avis Technique, depuis la demande de l'industriel à la délivrance de l'Avis Technique. Ils nous ont, aussi, fait part des problèmes rencontrés et notamment de la perte considérable de temps qu'engendrait ce processus perçu comme compliqué par les industriels.

Une fois ces entretiens terminés, nous avons identifié plusieurs profils d'utilisateurs susceptibles d'être concernés par les Avis Techniques :

- Les bureaux d'études techniques consultent les ATec lors de l'élaboration de réponses à des appels d'offre et recherchent parmi les produits répondant à leurs contraintes techniques ceux qui sont validés par un ATec. Dans un premier temps, il s'agit donc pour eux d'effectuer des recherches, au sein d'une famille de produits ayant obtenu un avis favorable du CSTB ou un domaine d'application donné. Dans un second temps, il leur faut comparer ces produits selon leurs caractéristiques techniques et leurs performances.
- Les fabricants de produits travaillant sur des Dossiers Techniques consultent les Avis Techniques du CSTB, en vue de préparer une demande d'ATec, soit à titre d'exemple soit au titre de veille technologique ou concurrentielle. Dans le cas d'une préparation de demande d'ATec, il s'agira pour ces fabricants de rechercher un produit à l'intérieur d'une famille de produits et pour un domaine d'application donné. Dans le cas d'une veille ce sont plutôt des noms commerciaux et des produits de sociétés connues qui sont recherchés. Une fois le produit à analyser repéré, le fabricant s'intéressera au contenu du Dossier Technique. Dans le cadre d'un choix de produit ou de la substitution d'un produit par un autre ayant des caractéristiques proches, artisans et entreprises recherchent les produits correspondant à une sous-famille et à un domaine d'emploi particulier. Une fois la sélection réalisée, les informations utiles sont la référence commerciale du produit sélectionné, ses caractéristiques techniques et ses précautions de mise en œuvre. La référence et les caractéristiques techniques permettront de piloter l'approvisionnement. Les précautions de mise en œuvre seront communiquées aux responsables de chantier.

Dans le prolongement des travaux de thèse d'Anastasyia Yurchyshyna au CSTB et à l'I3S sur la construction d'ontologie dans le domaine du bâtiment et le contrôle de conformité des projets de construction aux normes en vigueur (Yurchyshyna, 2009), nous abordons dans le

cadre de notre thèse la modélisation des textes réglementaires et des processus de contrôle de la conformité d'un texte par rapport à ceux existant. Plus précisément, nous avons travaillé à la modélisation du processus de rédaction d'un Dossier Technique et du processus de rédaction d'un Avis Technique, selon une approche basée sur des ontologies afin de construire un modèle sémantiquement riche du Dossier Technique et des contraintes réglementaires.

Pour parvenir à un Dossier Technique recevable par les instructeurs de CSTB, les industriels ont besoin de bien comprendre et bien renseigner une trame générique de Dossier Technique fournie par le CSTB. Cette trame contient toutes les informations nécessaires pour qu'un procédé, produit ou matériau soit étudié par un Groupe Spécialisé qui aura la charge de délivrer l'Avis Technique sur le dossier.

Cependant, les industriels n'arrivent pas à renseigner correctement la trame du Dossier Technique. Un dialogue s'instaure entre le CSTB et le demandeur et de trop nombreux échanges sont nécessaires avant d'arriver à une version acceptable du dossier. Le processus de renseignement du dossier est mal anticipé par les industriels et jugé trop administratif. Le CSTB de son côté ne souhaite pas mobiliser des ressources hors d'un quelconque devis. Comme conséquence directe, cela conduit à un traitement des dossiers long, de l'ordre de 6 à 8 mois. Le rôle du CSTB devrait se limiter à vérifier le contenu de ce dossier et non à assister l'industriel.

D'un point de vue applicatif, notre travail vise à répondre à deux questions importantes :

Comment peut-on assister les industriels dans leur tâche de rédaction de documents techniques ?

Comment peut-on assister les instructeurs du CSTB dans leur tâche de vérification de la conformité des Documents Techniques ?

Pour répondre à ces questions, nous avons développé une approche et mis en œuvre des outils d'ingénierie des connaissances pour guider les industriels dans la rédaction des dossiers techniques et aider les instructeurs CSTB à rédiger des avis techniques sur ces dossiers en générant automatiquement des rapports de conformité des dossiers qui peuvent être vus comme des éléments d'aide à la prise de décision, c'est-à-dire à l'émission d'un Avis

Technique favorable ou non. Ce faisant, nous avons abordé trois questions d'ingénierie des connaissances suivantes :

Comment représenter des dossiers techniques et le processus de leur acquisition par les industriels ?

Comment extraire des contraintes réglementaires à partir des guides pratiques du CSTB et comment les représenter ?

Comment représenter les processus métier de vérification de la conformité d'un dossier technique aux contraintes réglementaires qui s'appliquent et comment automatiser l'exécution de ces processus ?

Cette thèse présente les éléments de réponse que nous avons apportés à ces questions de recherche, en reposant sur les modèles et techniques du Web sémantique. De façon synthétique nous pouvons résumer notre contribution scientifique comme suit :

- Nous avons construit une ontologie de domaine, qui définit les principaux concepts impliqués dans l'élaboration des Avis Technique. Cette ontologie, appelée OntoDT, est représentée en OWL et couplée avec le thésaurus REEF développé au CSTB.
- Nous proposons un modèle et un outil de construction de dossier technique qui repose sur l'interrogation d'OntoDT et produit une représentation RDF d'un dossier technique.
- Nous avons choisi d'utiliser les standards SBVR (Semantics of Business Vocabulary and Business Rules) et SPARQL pour reformuler, à la fois dans un langage contrôlé et dans un langage formel, les contraintes réglementaires extraites des les guides pratiques. SBVR représente une assurance de la qualité du texte des contraintes réglementaires présentées à l'utilisateur et SPARQL permet l'automatisation de la vérification de ces contraintes. Ces deux représentations reposent sur l'ontologie de domaine que nous avons développée.
- Nous proposons un modèle de processus déclaratif qui intègre des connaissances expertes sur le processus même de vérification des dossiers techniques. Nous représentons un processus en RDF, et cette représentation intègre les représentations

SBVR et SPARQL des contraintes réglementaires. Le processus global de vérification d'un dossier technique est défini récursivement et construit automatiquement sur la base de la représentation RDF du dossier technique.

- Nous proposons un modèle et un outil de vérification automatique des contraintes de conformité qui peuvent l'être. Nous reposons sur un moteur de processus qui interprète la représentation RDF d'un processus, teste la conformité du dossier technique aux contraintes représentées en SPARQL que la représentation RDF du processus intègre et produit un rapport de conformité qui utilise les représentations SBVR des contraintes pour expliquer les prises de décision automatiques et lister à l'utilisateur de façon synthétique les contraintes qui lui restent à vérifier manuellement.

Le présent document est organisé en 5 chapitres.

Chapitre 1

Ce chapitre est consacré à la présentation de l'état de l'art actuel sur le problème du contrôle de conformité dans le domaine de la construction. En raison de la complexité de ce problème, nous nous concentrons, dans cet état de l'art, sur différents axes de recherche. Dans un premier temps, différentes définitions scientifiques et institutionnelles sont présentées. À partir de la littérature, nous mettons l'accent sur les différents types d'analyse et de modélisation de textes réglementaires. Ce tour d'horizon nous conduit à présenter les différents outils de contrôle de conformité. Nous discutons aussi les différentes approches de modélisation réglementaire basées sur des ontologies, nous présentons les composants ontologiques principaux et les langages de représentation du web sémantique. Cette analyse nous permet finalement d'aborder les travaux de recherche sur les modèles de processus basés sur des ontologies.

Chapitre 2

Dans ce chapitre, nous présentons notre approche pour modéliser le processus de rédaction d'un Dossier Technique. Nous présentons le processus traditionnel de rédaction ; nous détaillons notre contribution pour faciliter la rédaction de ces documents, en construisant une ontologie du domaine et en modélisant le processus de rédaction. Cette approche a pour but d'un côté de faciliter le travail des instructeurs pour vérifier les dossiers techniques et d'un autre côté d'aider et accompagner les industriels dans la rédaction de leurs dossiers.

Chapitre 3

Nous traitons, dans ce chapitre, la problématique de la modélisation des textes réglementaires et du processus de vérification du respect de ces textes réglementaires. Nous utilisons les Guides Pratiques du CSTB comme source documentaire. Nous présentons notre approche pour modéliser les contraintes réglementaires en langage naturel issues de ces guides pratiques dans un langage semi-formel, SBVR, puis dans un langage formel, SPARQL ; ces deux représentations reposent sur un même vocabulaire du domaine. Nous détaillons ensuite notre approche de modélisation des processus de vérification des contraintes réglementaire ; cette représentation est déclarative, la mise en œuvre de ces processus passe par l'interprétation de leur représentation. Nous détaillerons, les différents résultats et limitations de cette approche.

Chapitre 4

Ce chapitre décrit les différentes étapes de la vérification de la conformité d'un Dossier Technique aux textes qui s'appliquent et l'élaboration automatique d'un rapport de conformité. Nous mettons le point sur l'aide à la prise de décision, après le contrôle automatique (éventuellement partiel) de la conformité du Document Technique. La prise de décision finale est manuelle et s'appuie sur le rapport de conformité qui peut être généré automatiquement. Ce rapport d'identifier les éventuels aspects qui n'ont pas pu être vérifiés automatiquement.

Chapitre 5

Ce chapitre présente l'implémentation que nous avons réalisée d'un outil d'aide à la rédaction de dossier technique et d'un outil d'aide à la rédaction d'avis technique et l'expérimentation de ces deux outils. Nous décrivons les différentes technologies utilisées et les spécifications fonctionnelles et techniques de nos outils. Nous décrivons ensuite les expérimentations et validations que nous avons réalisées au sein du CSTB.

CHAPITRE 1

ETAT DE L'ART

1 Introduction 12
2 Modélisation des textes réglementaires 12
3 Contrôle de conformité dans le domaine du bâtiment 18
4 Ontologies et langages de représentation formelle 24
5 Modélisation de règles 35
6 Modélisation de processus à base d'ontologies 39
7 Conclusion 47

1 Introduction

Les réglementations sont un ensemble de textes (documents juridiques) qui cherchent à s'assurer que les politiques énoncées dans la législation pertinente sont réalisées

Dans le domaine du bâtiment, les réglementations, ou « codes du bâtiment », sont un ensemble de règles qui précisent le niveau minimum d'acceptabilité et de sécurité pour les projets de construction : des bâtiments (par exemple, des maisons privées) et des équipements utilisés dans le bâtiment (par exemple, installations photovoltaïques) (Satti, 2006). Ces réglementations jouent aussi le rôle d'encyclopédie complète des textes importants pour garantir la qualité d'un bâtiment, ses caractéristiques et minimiser son impact environnemental.

Parmi les règles de construction, on distingue : les textes officiels et les textes techniques. Les textes officiels sont soit à caractère obligatoire soit à caractère informatif.

- Les textes officiels à caractère obligatoire regroupent les lois, décrets, arrêtés et codes.
- Les textes officiels à caractère informatif regroupent les circulaires et les questions écrites.
- Les textes techniques regroupent les textes concernant les produits, procédures et équipements, et les règles de mise en œuvre. Il s'agit principalement des normes (NF, EN, ISO), avis techniques (ATec), documents techniques d'application (DTA), appréciations techniques d'expérimentation (ATEx) et documents techniques unifiés (DTU).

Nous nous focalisons, dans cet état de l'art, sur la représentation réglementaire, les différentes recherches sur le contrôle de conformité, la modélisation des contraintes réglementaires des processus. En fin de chaque section nous positionnons nos travaux par rapport aux différents travaux cités.

2 Modélisation des textes réglementaires

La représentation des contraintes réglementaire s'avère être une composante nécessaire dans l'élaboration de systèmes automatiques de vérification de conformité. Les sources actuelles de

réglementation sont pour la plupart textuelles. Elles nécessitent une interprétation humaine totale afin de les rendre accessibles dans une organisation bien structurée, sous forme électronique cohérente et non ambiguë (Fayyad, 2007). Les premiers travaux sur la formalisation de réglementation concernent l'extraction de connaissances à partir de sources textuelles. En effet, deux types d'approches, une automatique et une autre manuelle, ont été adoptés, pour analyser les réglementations, et ainsi les représenter dans un format exploitable par les machines.

2.1 Analyse automatique des réglementations

Projet E-POWER

Des premiers travaux se sont basés sur les techniques de traitement automatique du langage naturel (TALN), dans le projet européen E-POWER (EUROPEAN PROGRAMME FOR AN ONTOLOGY BASED WORK ENVIRONMENT FOR REGULATIONS AND LEGISLATION) [3], dont l'objectif était de développer une méthode formelle et des outils pour convertir des règlementations de leur format textuel en un format exploitable par les machines.

Le projet E-POWER a été appliqué à la description formelle du régime de pension en Belgique et aux Pays-Bas, ce qui a permis d'évaluer et de tester avec succès son applicabilité.

Les représentations de règlementation ont été basées sur des normes industrielles, le processus de traduction a été soutenu par des techniques de traitement de langage naturel visant à transformer les textes bruts (réglementations) en spécifications formelles (UML/OCL).

Travaux de Lau et al.

Dans (Lau et al., 2005), les auteurs ont développé une infrastructure pour la recherche de document juridique. Ils proposent de transformer des textes réglementaires en connaissances exploitables par la machine en analysant les structures hiérarchiques des documents et en ajoutant de nouveau méta-tags. La représentation XML est bien adaptée à la manipulation de données semi-structurées comme les documents juridiques et à l'extraction d'autres caractéristiques importantes (concept, date, etc.) pour les intégrer dans les connaissances juridiques. Cependant, les auteurs ont eu besoin de l'intervention d'experts du domaine pour arriver à ces résultats.

[3] http://cordis.europa.eu/search/index.cfm?fuseaction=proj.document&PJ_RCN=5179248

Travaux de Bolioli et al.

Dans (Bolioli et al., 2002) les auteurs proposent une classification des documents juridiques italiens basée sur leur étiquetage en XML. L'approche a permis de classifier d'importants termes qui identifient les textes réglementaires. Ces termes conceptuellement importants ont été pris en compte dans le processus d'extraction d'information et ont été formalisés dans les métadonnées XML.

Projet METALex

Le projet METALex vise à annoter des sources réglementaires (Boer et al., 2002). METALex propose un standard XML, l'une des premières normes ouvertes, particulièrement orientée vers la représentation de documents juridiques, mise en pratique aux Pays-Bas. Ce projet est destiné à couvrir toutes les décisions juridiques structurées et à les rendre exploitables pour la recherche et/ou pour différentes tâches de l'industrie. Il fournit un cadre générique pour l'encodage de la structure ainsi que le contenu des documents juridiques en XML. Il se concentre en particulier sur l'échange et les questions d'interopérabilité des documents juridiques.

NL2SBVR

Dans (Bajwa et al., 2011), les auteurs proposent un outil, NL2SBVR, qui permet de générer automatiquement des règles dans le langage contrôlé SBVR[4] (que nous détaillons plus loin dans ce chapitre) à partir de règles en langage naturel. Les auteurs ont testé cet outil sur des règles extraites du corpus « EU-Rent». L'outil a permis de traduire correctement plus de 80% des règles.

2.2 Analyse manuelle des réglementations

SPARCLE

Dans d'autres approches, les contraintes réglementaires sont écrites de manière normée avec l'aide d'experts du domaine, ce qui facilite leur traduction en langage formel. Dans (Brodie et al., 2006; Reeder et al., 2007), les auteurs aborde le problème de la gestion de politique de confidentialité dans les entreprises. Un outil « SPARCLE » a été créé pour aider les auteurs de politique de confidentialité en leur permettant de rédiger en langage naturel, d'analyser, de

[4] http://www.omg.org/spec/SBVR/1.0/

visualiser et de produire des politiques de confidentialités dans un langage XML lisible par les humains et interprétable par la machine : XACML.

Cet outil intègre un analyseur syntaxique (Neff et al., 2004) qui identifie les éléments de la politique de confidentialité des réglementations entrées en langage naturel. Il utilise une ou plusieurs grammaires abstraites conçues pour identifier les cinq éléments importants de la politique de confidentialité : catégories d'utilisateurs, actions, catégories de données, buts et conditions/obligations. L'analyseur syntaxique se base sur ces grammaires pour identifier les règles de politiques de confidentialité (écrites en langage naturel).

L'utilisation de ces grammaires a permis aux auteurs d'analyser, avec une précision comprise entre 88 % et 94 %, les politiques de confidentialité de grands services médicaux, des secteurs bancaires/finances et d'organisations gouvernementales.

Reeder indique que tout système de création de politique de confidentialité doit appliquer une terminologie cohérente, faire respecter la structure des règles, prévenir les conflits entre règles (Reeder et al., 2007).

Projet ISTforCE

Le projet ISTforCE (Turk et al., 2002) visait à développer une plateforme de collaboration ouverte, qui intègre des services et outils pour faciliter l'échange de données réglementaires.

Ce projet fournit une formalisation des représentations de l'accessibilité ainsi que d'autres textes juridiques connexes. Cette formalisation a été menée dans la perspective de les utiliser dans un service de contrôle de la conformité (CCS[5]).

La formalisation des textes réglementaires d'accessibilité dans ISTforCE se fait en trois étapes principales :

- La décomposition de textes juridiques, manuellement, en un ensemble de règles simples de conformité. Une base de textes réglementaires est ainsi créée. Elle est constituée de 6 documents juridiques relatifs à l'accessibilité en France. Ces textes ont été choisis et décomposés en 163 règles de construction par les experts du domaine.
- L'analyse de la possibilité de formalisation des règles. Les résultats du projet ont montré que 72 % des règles ont été non vérifiables. 6 % ont été partiellement vérifiables. 22 % d'entre elles pourraient être complètement vérifiées.

[5] http://istforce.eu-project.info/walkthrough.pps

- La formalisation des règles entièrement vérifiables et partiellement vérifiables par des algorithmes de contrôle de conformité. Ces algorithmes sont ensuite directement utilisés pour vérifier la conformité de la représentation des modèles au format « IFC[6] » d'un bâtiment aux réglementations en vigueur.

Malgré les nombreux avantages de l'approche formelle pour la formalisation de textes réglementaires d'accessibilité proposée dans le projet ISTforCE, la réutilisabilité de cette approche dans d'autres domaines, avec d'autres types de réglementations semble difficile. Ceci s'explique par une dépendance au langage de programmation utilisé (Delphi) et une interopérabilité faible (les algorithmes de contrôle de conformité ont été formalisés en Delphi). D'autre part, bien que reposant sur le format IFC du modèle BIM (Building Information Modeling) ayant une certaine expressivité sémantique, le projet ISTforCE n'exploite pas celle-ci.

SemEx

SemEx est une plateforme d'exploration sémantique et d'aide à l'acquisition des règles métier candidates à partir de leurs textes d'origine. Dans ses travaux, l'auteur s'est focalisé sur la transformation des contraintes réglementaires en langage naturel en SBVR (Guissé et al., 2012, 2011, 2009). D'abord, les experts du domaine désignent des règles dites candidates pour une transformation dans le langage contrôlé SBVR. Ces règles SBVR seront alors transmises à l'équipe technique pour une seconde étape de formalisation.

L'auteur ne s'occupe que de la première partie « la transformation des règles en langage naturel à SBVR ». L'approche a été testée en utilisant deux types de sources réglementaires : les conditions nécessaires pour bénéficier des avantages de cartes de fidélité dans les avions et les règlementations sur la sécurité des véhicules Audi.

La plateforme SemEx a été couplée à un outil d'extraction de termes, l'expert identifie les textes contenant potentiellement des règles candidates puis les transforme en SBVR. Dans cette étape de formalisation en langage semi-formel, l'auteur procède à quelques transformations sur les règles candidates :

- La première est une transformation lexicale, qui consiste à remplacer les termes issus de la réglementation par ceux du vocabulaire contrôlé (il utilise « Lexicalized ontology for à business rules management platform » (Omrane et al., 2011)).

[6] http://www.buildingsmart.org/standards/ifc

• La deuxième est la décontextualisation, on récrit les règle d'une façon à éviter toute ambigüité. Exemple : « Elle doit avoir une longueur égale à 6 m » devient « La ceinture de sécurité doit avoir une longueur égale à 6 m ».

• La troisième est la normalisation syntaxique, elle consiste à décomposer les textes réglementaires complexes en plusieurs phrases simples faciles à comprendre.

• La quatrième est la restauration sémantique, qui permet de restaurer les entités issues d'une transformation par normalisation syntaxique et/ou décontextualisation.

2.3 Positionnement

Puiseurs travaux ont traité la problématique de la représentation réglementaire, nous avons cité dans la partie ci-dessus les plus importants ainsi que les travaux qui se rapprochent le plus de notre problématique. La représentation réglementaire a été étudiée sous deux axes principaux : (I) analyser les réglementations automatiquement, (II) analyser les réglementations manuellement.

(I) Plusieurs approches, basées sur le TALN, ont été établies dans le premier axe. Le projet E-Power a permis de développer une méthode pour convertir les réglementations, automatiquement, de leurs formats textuels à un format interprétable par une machine. D'autres, comme (Lau et al., 2005), ont essayé de représenter les réglementations en XML. Boer et al., ont abordé le problème d'un autre point de vue, ils ont annoté les réglementations, en XML, dès leur création (Boer et al., 2002). Ceci a permis une indexation et une réutilisation facile des législations italiennes. Ceci a été exploité dans le projet METALex, où les réglementations ont été rendues exploitables en XML.

(II) Brodie & Reeder ont traité la problématique de l'analyse manuelle des réglementations. Ils ont participé à concevoir l'outil SPARCL. Cet outil, original, permet d'aider les experts à rédiger les politiques de confidentialité en langage naturel d'une part, et d'une autre part de les récrire en langage formel « XACML » (Brodie et al., 2006; Reeder et al., 2007).

Cependant, nous pensons que l'approche la plus prometteuse est l'utilisation du standard SBVR. Cela permet à la fois d'écrire les réglementions dans un langage contrôlé compréhensible par les humains et qui facilite, comme une représentation intermédiaire, leur formalisation dans un langage interprétable par la machine.

3 Contrôle de conformité dans le domaine du bâtiment

Dans (Eastman et al., 2009), les auteurs ont comparé plusieurs systèmes de contrôle de conformité dans le domaine du bâtiment, en se basant sur quatre critères : l'interprétation des réglementations, le modèle de bâtiment à contrôler, l'exécution des règles et l'élaboration des rapports de conformité.

Historiquement, Fenves a entrepris les premiers travaux sur la structure des réglementations dans des tables de décision (Fenves, 1966). Plus tard, les arbres de décision ont été appliqués dans le domaine de la construction, précisément dans la conception de bâtiments en acier (Nyman et al., 1973). Le système SASE (Standard Analysis Synthesis and Expression) (Fenves et al., 1987) a été développé afin de fournir une structure hiérarchique complète pour classer les familles de règlementations. Une étude importante de ces premières approches a été fournie dans (Fenves et al., 1995).

Dans la suite de cette section, nous présentons dans le détail les principaux systèmes de contrôle de conformité dans le domaine du bâtiment cités dans (Eastman et al., 2009).

3.1 Les travaux sur le contrôle de conformité à l'université de Standford

Le groupe de recherche du Centre CIFE de l'université de Standford a travaillé sur l'automatisation du contrôle de conformité dans le domaine de la construction. Leurs premiers travaux sur la vérification de conformité des bâtiments sont présentés dans (Han et al., 1998). Les auteurs mettent l'accent sur le traitement des codes de conception ; ils analysent les critères d'un modèle de bâtiment, la représentation des normes du bâtiment et la pertinence des dispositions en ce qui concerne les composants.

Ce travail est un point de départ pour une série de développements d'outils de raisonnement logique pour effectuer des contrôles de conformité. Le développement d'une infrastructure logicielle pour la gestion de l'information réglementaire est décrit dans la thèse de (Kerrigan and Law, 2003). Dans le cadre du projet REGNET qui s'intéresse aux réglementations environnementales, (Kerrigan et al., 2003) proposent un système de contrôle de conformité

basé sur le calcul des prédicats du premier ordre. Il comprend trois composantes principales (Kerrigan and Law, 2005) :

- Un référentiel documentaire contenant les règlementations. Ce référentiel comprend une série de concepts hiérarchisés qui permettent aux utilisateurs de parcourir les documents selon les termes qu'ils contiennent ;
- Un Framework XML pour représenter les règlements et les métadonnées associées.
- Un système de contrôle de conformité construit sous le Framework XML.

Le Framework de réglementation comprend des balises XML pour chaque niveau des textes réglementaires (partie, sous-partie, section ou du paragraphe) qui reflète la structure standard de la réglementation.

La transformation des réglementations de leurs formats de PDF et HTML en XML est réalisée par des systèmes d'analyse spéciaux en utilisant une grammaire ainsi qu'une sémantique de représentation et d'interprétation (Kerrigan and Law, 2003). Ce parseur utilise un modèle de correspondance pour identifier la structure d'un texte réglementaire et créer une structure XML explicite autour de ce texte. Exemple, la balise <logicOption> est utilisée pour construire une structure de questions-réponses. Elle contient un élément question pour capturer les entrées utilisateurs et un ou plusieurs éléments de réponse pour représenter les réponses possibles.

Une approche similaire, du même groupe de recherche, dédiée à l'analyse de la réglementation et à l'aide à la vérification de conformité eGov, est décrite dans (Lau et al., 2004) et (Lau et al., 2006). Les auteurs se sont concentrés en particulier sur les aspects pratiques d'implémentation d'un système de conformité.

D'autres chercheurs ont aussi travaillé sur différents aspects du problème de vérification de la conformité. Dans (Pan et al., 2008), les auteurs ont analysé les techniques de planification pour déterminer l'existence d'une voie accessible dans un environnement CAO.

Kam a mis particulièrement l'accent sur les technologies innovantes qui peuvent être utilisées pour la modélisation dans le domaine du bâtiment : les ontologies, la prise de décision, etc. (Kam, 2005).

Ce travail explique également la réorientation partielle des axes de recherche du groupe de recherche du Centre CIFE de l'université de Standford. Les chercheurs soulignent la nécessité d'augmenter l'expressivité de la connaissance réglementaire dans le domaine du bâtiment et de

travailler sur le développement d'ontologies compatibles et sur des systèmes de raisonnement basés sur la logique (par exemple (Cheng et al., 2007; Pan et al., 2008)).

3.2 Projet CORENET

Introduit en 1995 par le ministère du Développement national de Singapour, le projet Singapour CORENET[7] fut le premier effort de contrôle de conformité automatique dans le domaine du bâtiment. Ce projet a été implémenté par le Singapore Building and Construction Authority. Il consiste en trois modules : CORENET e-Submission, CORENET e-Plan Check et CORENET e-Info. Le module « e-Submission » suit à la trace les mesures à prendre pour les permis de construire, le module « e-Info » fournit des informations de construction diverses. Le module « e-Plan Check » a initialement été créé pour le développement de la conception en utilisant des dessins électroniques, puis en 1998, il a été utilisé pour opérer sur les données représentées au format IFC du modèle BIM (Building Information Modeling).

Ce projet vise à contrôler automatiquement la conformité des plans de construction de bâtiments. Le contrôle de conformité se réfère à des normes sur l'accessibilité, la sécurité des bâtiments, des normes sur les alarmes incendie, la résistance au feu, la santé publique, etc.

Une fois le contrôle de conformité effectué, E-plan check génère un rapport de conformité en identifiant les règles qui ont été appliquées. Plusieurs formats sont utilisés pour afficher le rapport de conformité (HTML, Word, etc.).

CORENET a été le premier système de contrôle de conformité dans le domaine de la construction. Il est utilisé de façon productive aujourd'hui et propose plusieurs exemples d'utilisation. Comme résultat, 92 % des règles sont applicables sur des plans de construction et plus de 2500 entreprises affirment utiliser ce système.

3.3 Projet AC3 « SMART code »

Le projet AC3 a été développé par ICC (the International Code Council)[8] aux Etats-Unis avec une équipe projet composée d'experts internationaux (AEC3, Digital Alchemy, Battelle Memorial Institute, et ERDC Illinois).

[7] http://www.corenet.gov.sg
[8] http://www.iccsafe.org/news/about/

Dans le cadre du projet AC3 une approche et un format ont été développés pour l'automatisation de la vérification de la conformité :

- SMARTcodes est un format interopérable de formalisation de réglementations concernant la conservation des énergies « International Energy Conservation Codes ». Cette formalisation peut identifier les conflits entre la modélisation de bâtiments en format BIM et les normes/réglementations énergétiques en vigueur (Nawari, 2011a).
- MCS (Model Checking Software) est un outil de contrôle de conformité développé par l'ICC qui exploite les réglementations écrites en SMARTcodes ; il supporte les modèles en BIM et contrôle leur conformité vis-à-vis des réglementations en vigueur. Une fois le contrôle de conformité terminé, une information graphique, correspondant aux résultats de la vérification, est affichée aux utilisateurs. Cette information graphique simplifie l'explication du « pourquoi » le/les bâtiment(s) en question n'est/ne sont pas conforme(s). L'exhaustivité et la transparence du processus de vérification sont, par conséquent, également gérés par MCS (Nawari, 2011b).

3.4 Projet HITOS

Le projet HITOS (Ernst Eberg et al., 2006) s'intéresse à l'évaluation des exigences de données dans le modèle IFC avec dRofus, et à la vérification de l'accessibilité des bâtiments en utilisant le logiciel SMC (Solibri Model Checker)[9] qui manipule des règles d'accessibilité issues de normes ISO pour vérifier l'accessibilité d'un bâtiment. SMC supporte des processus avec des données non seulement géométriques d'un objet de construction, un « bâtiment », mais aussi d'autres objets associés et leurs propriétés, comme les matériaux utilisés, espaces des portes, escalier, rampes et fenêtres. La majorité de règles définissent des relations entre les objets des composants.

SMC récupère toutes les règles et leurs données géométriques et recherche une correspondance (mapping) entre le modèle IFC du bâtiment et les valeurs exigées représentées dans ses règles de vérification.

SMC fournit un rapport de conformité textuel et graphique dans plusieurs formats. Il peut fournir aussi le degré de violation de contrainte normative selon trois niveaux : Critique, Modéré et Bas.

[9] http://www.solibri.com

3.5 Modèle C3R

Yurchyshyna a proposé un modèle C3R (Yurchyshyna, Faron-Zucker, Mirbel, et al., 2008), pour le contrôle semi-automatique de la conformité des projets de construction par rapport aux normes techniques du bâtiment. Les projets de construction utilisés sont représentés au format IFC du modèle BIM. Cependant, afin d'augmenter l'expressivité, une ontologie de domaine a été construite à partir des classes du modèle IFC.

Yurchyshyna s'est focalisée essentiellement sur les règles d'accessibilité. Celles-ci sont formalisées manuellement, avec l'aide d'experts du domaine, à partir de réglementations techniques d'accessibilité en requêtes SPARQL. La formalisation des règles techniques par des requêtes formelles de conformité comprend deux étapes (Yurchyshyna, 2009) :

- Les différentes contraintes réglementaires, relatives à l'accessibilité des bâtiments, sont extraites manuellement par des experts du domaine à partir de sources réglementaires puis transformées sous forme de requêtes SPARQL. Pour ce faire, une étape préliminaire a été la construction d'une ontologie légère en RDFS orientée conformité. Les classes et propriétés de cette ontologie sont extraites automatiquement du modèle IFC. Cette ontologie a été construite en collaboration étroite avec les experts du domaine ; des concepts non existant dans le modèle IFC, mais présents dans les règlementations ont été ajoutés comme des spécialisations des concepts issus du modèle IFC.

- Les normes de bâtiment ont été ensuite modélisées par un ensemble de requêtes de conformités :

- Celles qui expriment des faits qui doivent toujours être vérifiés sont représentées par une requête SPARQL dite positive, exprimant la contrainte d'applicabilité. Par exemple (Yurchyshyna, 2009), la contrainte « *La largeur minimale des portes doit être 90 cm* ».

- Celles qui décrivent des faits qui doivent être vérifiés seulement si certaines conditions sont vérifiées sont représentées par une requête SPARQL dite négative permettant de retrouver des éléments non conformes. Par exemple (Yurchyshyna, Faron-Zucker, Mirbel, et al., 2008) la contrainte « *Dans les bâtiments comportant plusieurs halls d'entrée, tous les halls d'entrée doivent être accessibles »* est représentée par deux requêtes SPARQL : la première requête teste l'existence de halls d'entrée et la deuxième renvoie les éventuels halls non accessibles.

Afin d'organiser les requêtes de conformité et d'y intégrer des connaissances expertes, elles sont annotées en RDF ; ces annotations contiennent :

- les caractéristiques du texte réglementaire à partir duquel la requête est construite,
- les caractéristiques d'extraction,
- la formalisation en SPARQL de la contrainte réglementaire,
- Le contexte d'application.

350 textes réglementaires issus de 9 réglementations ont été analysés. 5 % de ces règles sont totalement interprétables, 20 % sont non transformables en IFC, 40 % sont partiellement interprétables, et 35 % sont non interprétables.

Une représentation RDF d'un projet de construction est construite par transformation « XSLT » de sa représentation IFC, basée sur l'ontologie du contrôle de conformité (Yurchyshyna, 2009) ; seules les informations utiles au contrôle de conformité sont extraites. Cette représentation RDF est vérifiée automatiquement en l'interrogeant avec les requêtes SPARQL représentant les contraintes réglementaires. Le contrôle de conformité consiste en cet appariement des représentations des normes avec celles des projets de construction. L'ordre de traitement des requêtes a été défini en fonction des priorités existant entre les classes de requêtes et les connaissances supplémentaires spécifiant les requêtes.

3.6 Positionnement

Le contrôle automatique de la conformité des bâtiments est une des problématiques les plus étudiées dans le domaine de la construction. Plusieurs approches ont été proposées afin de vérifier la conformité des propriétés structurelles et dimensionnelles des bâtiments aux exigences réglementaires en vigueur. (Kerrigan and Law, 2003), ont participé au projet SASE et REGNET. Le groupe de recherche du Centre CIFE de l'université de Standford a travaillé sur l'automatisation du contrôle de conformité de plusieurs projets. Le premier, a proposé une approche pour analyser les critères d'un modèle de bâtiment et la pertinence de ce modèle (Han et al., 1998). Le second, REGNET, permettait de déterminer pour une condition donnée l'applicabilité de différentes réglementations. Cette vérification, semi-automatique, été basée sur un système de questions/réponses permettant à REGNET d'être un des premiers projets dans ce domaine. Ces deux projets produisent un rapport de conformité attestant que le bâti est conforme ou non aux contraintes réglementaires en vigueur.

Les projets CORNET, HITOS et AC3 ont utilisé les IFC comme modèle de représentation. CORNET est le projet le plus complet et le plus utilisé (2500 entreprises affirment l'utiliser).

Les travaux de thèse d'A. Yurchyshyna mené dans notre équipe ont également utilisé le modèle IFC comme base pour la construction d'une ontologie du domaine de la construction (Yurchyshyna, 2009). Elle utilise IFC pour représenter les modèles de bâtiments, les transformer en RDF, puis applique des règles de contrôle de conformité représentées en SPARQL.

Notre thèse s'inscrit dans la continuation de ces travaux. Comme nous le verrons dans les chapitres suivants, nous utilisons RDF comme modèle de représentation des connaissances à vérifier, SBVR et SPARQL comme modèles de représentation, respectivement semi-formel et formel, des réglementations en vigueur. Ceci a un double avantage :

- Nous avons réuni les différentes règles à appliquer sous forme de processus de vérification, réutilisable pour plusieurs autres scénarios de vérification de conformité. Exemple, le processus de vérification d'une tuile peut être appliqué sur tout type de tuile, du moment qu'il charge les requêtes adéquates ;
- Nos rapports de conformité sont plus riches et plus complets car ils contiennent toutes les informations sur les règles appliquées. Aussi, pour chaque règle échouée, nous utilisons les liens entre les requêtes SPARQL et règles SBVR pour identifier la cause de la non-conformité et établir un rapport précis.

Enfin, toutes les formalisations sont basées sur un vocabulaire contrôlé commun, l'ontologie OntoDT.

4 Ontologies et langages de représentation formelle

La notion d'ontologie dans son sens moderne remonte à la définition qu'en donne Aristote dans *Les Métaphysiques* : « l'étude de l'être en tant qu'être, indépendamment de ses déterminations particulières ». Dans *Les Catégories*, Aristote introduit les catégories comme des genres suprêmes de l'être, instruments de cette connaissance de l'être. Dans la continuité de ce discours, en ingénierie des connaissances, on entend par ontologie un système particulier de représentation des connaissances. La définition de (Gruber, 1993) fait

référence : « Une ontologie est une spécification *explicite* et *formelle* d'une conceptualisation *partagée* d'un domaine de connaissance ».

Dans son état de l'art Mechouche résume que l'objectif premier d'une ontologie est de modéliser un ensemble de connaissances dans un domaine donné, qui peuvent être concrètes ou abstraites (Mechouche, 2009). De nos jours, les ontologies sont de plus en plus utilisées dans tous les domaines dans les systèmes d'information. Ce succès est dû principalement à leurs capacités à représenter les connaissances selon des règles standards. Elles sont considérées comme un paradigme clé dans l'interopérabilité sémantique, le partage de connaissances, l'intégration de données. Elles permettent alors le partage des données hétérogènes, la représentation des connaissances d'un domaine de manière explicite et sans ambigüité, le raisonnement sur les connaissances représentées.

Les ontologies sont formalisées au moyen d'un système de représentation logique dont la sémantique est spécifiée de façon non ambigüe (Gruber, 1995; Guarino, 1995). Une ontologie est composée principalement de concepts, de relations et d'axiomes (Gruber, 1995; Studer et al., 1998).

- Les concepts sont des notions sémantiques qui peuvent être classifiées selon leur niveau d'abstraction, leur atomicité et leur niveau de réalité (élément réel ou fictif) ;
- Les relations représentent les différentes interactions entre les concepts, elles peuvent être des relations sémantiques ou des relations de subsomption ;
- Les axiomes sont utilisés pour définir les composants d'ontologie, les contraintes sur les valeurs des attributs et les arguments de relations ;
- Les fonctions représentent des cas particuliers de relation.

Selon (Mechouche, 2009), une ontologie doit répondre à un certain nombre de critères ; elle doit être :

- Correcte : (Noy and Mcguinness, 2008) démontrent qu'on peut trouver plusieurs ontologies de référence pour un domaine précis. Cependant, il est rare, voire impossible, de construire une ontologie identique à partir d'entités différentes. Alors, on ne peut pas définir la qualité d'une ontologie à partir de ses étapes de construction, mais à partir des usages pour lesquels elle a été conçue (Noy and Mcguinness, 2001). Une ontologie est considérée comme correcte si elle satisfait les besoins des applications l'utilisant,

• Documentée : une ontologie doit être documentée en langage naturel pour faciliter sa compréhension et son utilisation. En effet une documentation peut être une des barrières du partage effectif d'une ontologie inadéquate (Uschold et al., 1996).

• Réutilisable : une ontologie permet une représentation de connaissances sémantiquement interopérable, et sa réutilisation par plusieurs applications ou d'autres ontologies est tout à fait possible. En effet, un des principes de la construction d'ontologies est de réutiliser le plus possible des ontologies excitantes (Badra et al., 2008). Ces ontologies sont pour la plupart accessibles sur le web et représentent des domaines variés ainsi que des ontologies de haut niveau, comportant des concepts généraux subsumant les concepts existant dans différents domaines spécifiques.

• Rigoureuse : une ontologie doit définir des connaissances liées à son domaine d'utilisation.

• Explicite : les composantes principales d'une ontologie sont définies explicitement à l'aide d'axiomes.

• Consensuelle : l'organisation et la définition des concepts et des relations d'une ontologie doit être faite de manière consensuelle dans le domaine en question.

• Extensible : les extensions d'une ontologie doivent être anticipées pour éviter que d'un nouveau concept ne puisse la remettre en cause.

• Cohérente : l'ontologie et ses axiomes doivent être cohérents, aucune connaissance inférée de l'ontologie ne doit pouvoir entrer en contradiction avec ses autres concepts (Gruber, 1993).

4.1 Classifications des ontologies

Il existe trois classifications principales pour les ontologies. Dans (Heijst et al., 1997), les auteurs propose de classifier les ontologies selon deux dimensions :

- Selon la première dimension il regroupe les ontologies terminologiques, les ontologies d'information et les ontologies de modélisation des connaissances.
- Selon la deuxième dimension, il regroupe les ontologies d'application, les ontologies de domaine, les ontologies génériques et les ontologies de représentation.

Guarino conteste particulièrement la première dimension de la classification de Heijst, qu'il estime insuffisamment claire (Guarino, 1997). Il propose à son tour une classification, basée

sur le niveau de dépendance d'un point de vue ou d'une tâche particuliers. Il définit trois classes d'ontologies :

- Les ontologies de haut niveau, qui permettent de décrire des concepts généraux communs à plusieurs domaines ;
- Les ontologies de domaine et de tâche, permettant de décrire les concepts génériques d'un domaine particulier, tel que le bâtiment, ou une tâche générique d'un domaine.
- Les ontologies d'application décrivant les concepts dépendant à la fois d'un domaine et d'une tâche particulière.

Oberle a proposé de classifier les ontologies selon les buts pour lesquels elles sont conçues, leur niveau d'expressivité et leur spécificité (Oberle et al., 2006).

- Selon le *but*, dans cette classification, l'auteur distingue :
    - les ontologies d'application, utilisés lors d'un raisonnement basé sur une ontologie,
    - les ontologies de référence, utilisées pour faciliter la communication entre différents agents.
- Selon l'*expressivité*, dans cette classification, l'auteur distingue :
    - les ontologies lourdes (HeavyWeight), utilisées généralement pour effectuer des raisonnements complexes, qui sont très axiomatisées,
    - les ontologies légères (LightWeight), peu ou pas axiomatisées, qui se limitent à des structures taxonomiques simples.
- Selon la *spécificité*, dans cette classification, l'auteur distingue trois types d'ontologies :
    - les ontologies génériques ou de haut niveau, qui permettent de définir des concepts généraux à tous les champs d'application,
    - les ontologies noyau (Core Ontology), qui permettent de définir des concepts génériques d'un ensemble de domaines,
    - les ontologies de domaine, qui permettent de définir les concepts d'un domaine particulier.

On peut constater une certaine ressemblance entre les ontologies de haut niveau et les ontologies génériques, définies respectivement par Guarino et Oberle (Guarino, 1997; Oberle et al., 2006). Cette constatation s'étend aussi à leurs ontologies de domaine, de tâches, et aux ontologies de noyaux.

Dans nos travaux, en considérant la classification de (Guarino, 1997), nous avons construit une ontologie d'application, spécifique au domaine du photovoltaïque et à la tâche de contrôle de conformité. Cette ontologie constituera le socle de nos outils d'aide à la rédaction réglementaire et pourra être réutilisée dans d'autres systèmes.

4.2 Les bonnes pratiques pour la construction d'ontologies

Afin de construire une ontologie en bonne et due forme, il faut respecter un certain nombre de règles. Plusieurs méthodes de construction d'ontologies ont été proposées par différents auteurs. Dans (Corcho et al., 2003; Fernández-López and Gómez-Pérez, 2002), les auteurs ont comparé plusieurs de ces méthodes de construction d'ontologies. Cette multiplicité reflète à la fois la difficulté de construire des ontologies et la difficulté de s'entendre sur la nature et le contenu de ces dernières (Bachimont, 2004). De manière générale, la méthodologie choisie pour la construction d'une ontologie doit être guidée par le type de l'ontologie et ses objectifs d'utilisation. En effet, il existe deux façons de construire une ontologie, automatique (semi-automatique) ou manuelle (Bourigault and Aussenac-Gilles, 2003).

- Les techniques de construction d'ontologies automatique ou semi-automatique permettent de produire des ontologies légères, avec une expressivité réduite destinée principalement à des applications précises. Ces ontologies sont souvent non réutilisables et ne présentent aucun engagement ontologique.
- Les techniques de construction d'ontologie manuelle est souvent fastidieuse et nécessite beaucoup de temps et de ressources. Les constructeurs se basent sur d'autres ontologies excitantes et surtout sur le savoir-faire d'experts du domaine. Les ontologies produites manuellement sont richement axiomatisées, rigoureuses et réutilisables (Borgo and Leitão, 2004; Bouaud et al., 1995).

Dans (Aussenac-Gilles et al., 2000) les auteurs se focalisent sur une seule étape de la construction d'une ontologie : celle de « conceptualisation par l'analyse d'un corpus textuel ».

Wang and Xu ont proposé un exemple à suivre pour la construction d'une ontologie établie en quatre étapes (Wang and Xu., 2000) :

1. L'analyse et l'acquisition des connaissances du domaine,

2. La structuration de l'ontologie et l'identification des différentes composantes ontologiques (concepts, relation, attribut, axiomes, etc.),
3. La représentation de l'ontologie dans un langage formel,
4. L'évaluation de l'ontologie et de ses performances.

Dans (Noy and Mcguinness, 2008) les auteurs proposent un processus de construction d'ontologie itératif, en sept étapes :

1. Détermination de l'étendue du domaine de l'ontologie. On définit le domaine de l'ontologie, le but de sa construction, les questions auxquelles l'ontologie devrait permettre de répondre ainsi que les entités qui vont l'utiliser et la maintenir.
2. Réutilisation des ontologies existantes. Cette étape est nécessaire si l'ontologie est susceptible d'être utilisée par d'autres systèmes qui interagissent avec d'autres applications.
3. Identification des termes importants de l'ontologie. Il s'agit des termes dont la sémantique doit être détaillée.
4. Définition des classes et des hiérarchies de classes. Ceci est fait soit selon une approche top-down qui consiste à définir d'abord les concepts les plus généraux du domaine puis leur spécialisation, soit selon une approche bottom-up, qui consiste à définir d'abord l'ensemble des classes spécifiques et à ensuite les regrouper pour définir des concepts plus généraux. On peut aussi combiner ces deux approches en une seule (middle-out).
5. Définition des propriétés des classes.
6. Définition des facettes des propriétés. Il s'agit de définir les valeurs pour des propriétés particulières.
7. Création d'instances. Il s'agit de créer des instances de classes pour le peuplement de l'ontologie.

Dans (Ben Mustapha et al., 2006) les auteurs proposent une approche pour la construction d'ontologie « from scratch » :

- Une ontologie est construite à partir de corpus documentaires textuels ou multimédia ou de bases de données semi-structurées.
- L'ontologie est validée par des experts qui vérifient formellement sa cohérence.
- L'ontologie est diffusée et validée par ses utilisateurs (la validation s'effectue une fois que les utilisateurs ont testé l'ontologie).

Dans nos travaux de thèse, nous avons construit une ontologie, où sont définis les concepts principaux qui interviennent dans le domaine du Photovoltaïque. Comme nous le verrons dans les chapitres suivants, nous avons construit, *manuellement*, une ontologie du *domaine* à partir de connaissances recueillies lors d'interviews d'experts du CSTB et en réutilisant des ressources ontologiques existantes.

4.3 Langages de représentation des connaissances

Les langages de représentation des connaissances sont destinés à représenter et organiser le savoir humain pour l'utiliser et le partager. Plusieurs outils de représentation de connaissances, simples, existent (les taxonomies, les thésaurus). D'autres permettent de représenter des connaissances plus complexes (graphes conceptuels, systèmes à base de règles). Le développement du web sémantique a permis une évolution de ces outils en introduisant les ontologies.

En effet, le web sémantique propose un certain nombre de langages qui ont été développés et définis par le W3C, comme des standards de représentation des connaissances (Gandon et al., 2012). Ces langages sont représentés (**Figure 1**) dans une architecture en plusieurs couches, dont les couches les plus basses assurent l'interopérabilité syntaxique (URI, UNICO, XML). Les couches supérieures correspondent aux langages de description des données (RDF), de modélisation d'ontologies (RDFS et OWL), de requêtes (SPARQL), de règles (RIF).

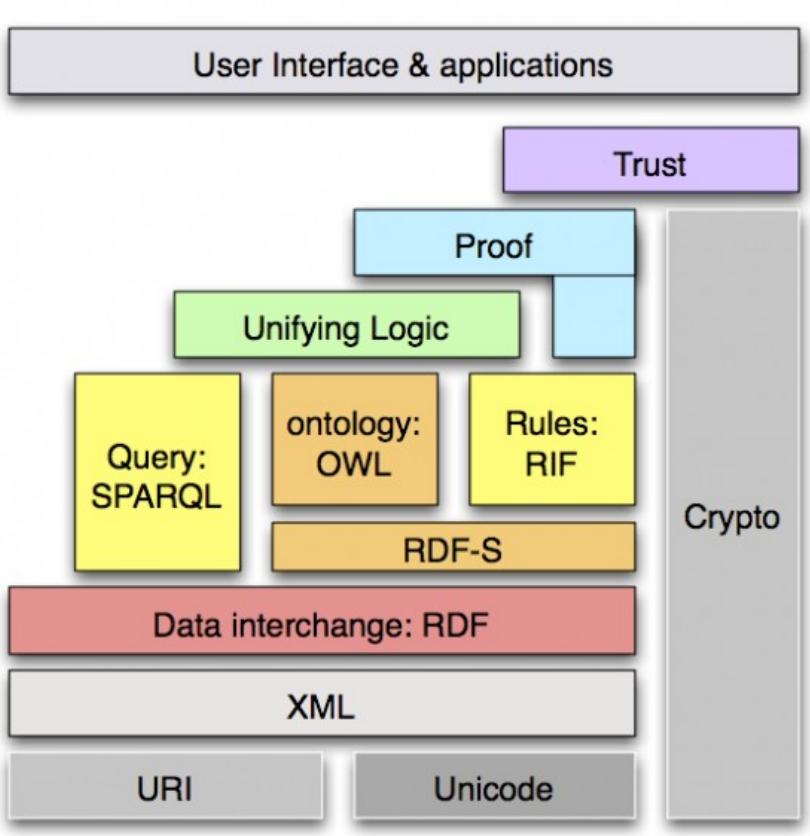


Figure 1 Architecture des langages de représentation des connaissances du W3C

XML

XML (eXentesible Markup Language) est un langage informatique de balisage générique. Il permet de créer, de manipuler et de décrire la structure des documents. Il facilite l'échange automatisé de contenus entre systèmes d'information hétérogènes. Néanmoins, il n'impose aucune contrainte sémantique. En effet, l'interopérabilité syntaxique n'est pas suffisante pour qu'un système comprenne le contenu des données XML et les manipule d'une manière significative.

XML fournit une syntaxe pour d'autres langages de plus haut niveau, dont ceux du web sémantique : RDF, RDFS et OWL. L'utilisation d'espaces de nommage (Namespace) permet d'identifier les noms des balises XML utilisées.

RDF

RDF (Resource Description Framework) (Klyne and Carroll, 2004) est un modèle de représentation sémantique, considéré comme le premier langage garantissant l'interopérabilité sémantique. C'est LE modèle de représentation de données pour le web sémantique. Les modèles de représentation RDF sont composés de plusieurs ressources web identifiées par un URI (Uniform Resource Identifier) (Berners-Lee et al., 2005). Un URI permet d'identifier de manière universelle une ressource qui peut être aussi bien : (1) une donnée présente sur le Web, (2) un objet du monde réel auquel on souhaite associer un identifiant, ou encore (3) une relation (l'appartenance, la filiation, …) (Passant, 2009).

La structure fondamentale de RDF, est un graphe (on parle de « graphe RDF »), composé de triplets composés chacun d'une `ressource`, d'une `propriété`, et d'une `valeur`.

- La `ressource`, i.e. le `sujet`, est une entité accessible via une URI,
- La `propriété`, i.e. le `prédicat`, est une URI qui définit la relation entre la `ressource` et la `valeur`,
- La `valeur`, i.e. l'`objet`, peut être une `ressource` ou une `valeur littérale` (entier, chaîne de caractères, etc.).

RDF est considéré comme la base du web sémantique. Le web Sémantique est, alors, considéré comme un *graphe* dont les ressources sont interconnectées via des propriétés, là où le Web connecte des documents via des liens hypertextes.

RDF est muni d'une syntaxe XML selon la recommandation du W3C, ainsi que d'une syntaxe Turtle que nous utiliserons dans les exemples de cette thèse.

RDFS

Les schémas RDF (RDFS) définissent le vocabulaire utilisé dans les descriptions RDF (Brickley and Guha, 2004). En effet, RDFS est un modèle permettant de représenter des hiérarchies de classes et des hiérarchies de propriétés, et de typer les sujets et valeurs des propriétés. Il étend RDF en décrivant plus précisément les ressources utilisées pour étiqueter les graphes RDF.

RDFS est ainsi un langage simple de représentation d'ontologies dites légères.

OWL

OWL (Ontology Web Language) est LE langage de représentation des ontologies sur le web. C'est une extension de RDFS dont une syntaxe est RDF/XML. Il offre une plus grande capacité d'interprétation du contenu Web, grâce à un vocabulaire sémantiquement plus riche. En effet, OWL est issu des travaux sur la représentation d'ontologie dans les logiques de description : il descend de OIL (Fensel et al., 2000), DAML+OIL (Horrocks, 2002) et DAML-Ont (McGuinness et al., 2003).

Une ontologie OWL est composée de métadonnées, d'axiomes et de faits. Les axiomes permettent la définition complète ou partielle de classes, de relations ainsi que la spécification de propriétés. Les faits sont des descriptions d'instances de classes de l'ontologie.

Il existe trois sous-langages d'OWL 1 à l'expressivité croissante : OWL Lite, OWL DL et OWL Full.

- OWL Lite est le sous-langage d'OWL le plus simple. Il permet de modéliser des ontologies légères avec quelques constructeurs supplémentaires à RDFS comme les propriétés algébriques des propriétés et les contraintes de cardinalité.
- OWL DL permet le plus haut niveau d'expressivité et garantit la complétude des raisonnements et leur décidabilité.
- OWL Full est une extension de OWL DL qui garantit la compatibilité avec RDFS : une ressource peut être à la fois une classe, un individu et une propriété. Cependant, OWL Full ne garantit pas la décidabilité.

OWL 2 introduit de nouveaux composants ontologiques tels que les axiomes de classes disjointes et d'union disjointe de classes, de nouvelles propriétés permettant d'exprimer de nouvelles restrictions de valeur, de cardinalité ainsi que des propriétés d'annotation, la notion de chaîne de propriétés, de nouveaux types de données et la notion de facettes. OWL2 regroupe quatre sous-langages :

- OWL 2 DL : regroupe comme toutes les primitives de OWL 2 et impose certaines restrictions d'utilisation comme OWL DL ; correspond à la logique de description « SROIQ ».
- OWL 2 EL : correspond à la logique de description EL (Existential Langage) ne fournit que la quantification existentielle.

- OWL 2 QL (Query Langague) : est un sous-ensemble des primitives de OWL 2 spécifique aux opérations de réponse à des requêtes, qui peut être implémenté en réécrivant les requêtes dans un langage relationnel comme SQL.
- OWL 2 RL (Rule Language) : est un défini à partir de OWL 2 en imposant certaines restrictions pour empêcher d'inférer l'existence d'individus non déclarés explicitement dans la base ; il peut être implémenté par un système à base de règles (implications de la logique du premier ordre).

SPARQL

SPARQL[10] est un langage de requêtes, inspiré de SQL, qui permet d'interroger des données RDF. Il est adapté à la structure spécifique du langage RDF, et s'appuie sur les triplets qui les constituent. Il permet, l'ajout/ la surpression, la recherche et la modification des données au format RDF. SPARQL peut également être utilisé pour interroger des vocabulaires RDFS ou OWL (écrits en RDF). Une requête SPARQL se compose de trois parties :

- un opérateur définissant le type de requête :
    - SELECT, permet de sélectionner différents éléments selon un patron de requête particulier.
    - ASK, produit « TRUE » ou « FALSE », en identifiant si oui ou non le patron recherché est présent dans le graphe interrogé
    - DESCRIBE, renvoie sous forme d'un graphe RDF une description de la ressource passée en paramètre
    - CONSTRUCT, permet de construire un graphe RDF à partir d'un autre graphe.
- un graphe requête, à apparier avec les données RDF à interroger
- un modifieur des résultats de la requête (par exemple, ORDER BY).

RIF

Rule Interchange Format (RIF) est une recommandation W3C visant à rendre interopérables les langages de règles existants en proposant un format de représentation et d'échange de règles. Le noyau de RIF est le dialecte RIF-CORE correspondant à la logique de Horn sans symbole de fonction. RIF-Core est utilisé comme langage de condition des règles de RIF. Le

[10] http://www.w3.org/TR/rdf-sparql-query/

dialecte RIF-BLD (Basic Logic Dialect) permet de représenter des programmes logiques. Le dialecte RIF-PRD (Production Rule Dialect) permet de représenter des règles de production dont l'application déclenche des actions prédéfinis.

Comme nous le verrons dans les chapitres suivants, nous utilisons OWL DL pour formaliser l'ontologie du domaine OntoDT, nous utilisons RDF pour représenter les Dossiers Techniques du CSTB et les processus de vérification de la conformité aux réglementations, et nous utilisons SPARQL pour représenter les contraintes réglementaires relatives au domaine du bâtiment.

5 Modélisation de règles

Plusieurs travaux ont été entrepris sur la modélisation de règles et plusieurs langages de représentation formelle ont été proposés, comme SILK (Grosof, 2009), N3-Logic (Berners-Lee et al., 2007), SWRL[11], ou SPIN[12]. Le standard RIF du W3C a été développé pour faciliter l'interopérabilité entre ces différents langages de règles mais à ce jour il n'est pas adopté en tant que langage de représentation de règles. Dans le cadre de nos travaux, nous nous sommes orientés vers les langages adoptés par la communauté du web sémantique : SPARQL et SPIN. Nous avons également adopté SBVR comme langage de représentation semi-formel basé sur un vocabulaire contrôlé.

5.1 SBVR

Les langages contrôlés ont été proposés comme une solution intermédiaire entre langage naturel et langage formel, surtout pour améliorer la collaboration entre les ingénieurs et les experts du domaine (Dubauskaitė and Vasilecas, 2009; Von Halle et al., 2006).

L'objectif principal du standard SBVR (Semantics of Business Vocabulary and Business Rules) de l'OMG est de fournir un méta-modèle qui permette d'établir des interfaces d'échanges de données pour les outils qui créent, organisent, analysent, formalisent les règles métier. Le méta-modèle SBVR vise à faciliter la validation, l'analyse, l'alignement, et la

[11] http://www.w3.org/Submission/SWRL/
[12] http://www.w3.org/Submission/spin-sparql/

formalisation de règles métier pour différents outils de différents constructeurs (Spreeuwenberg and Healy, 2009).

Les règles SBVR reposent sur la logique des prédicats pour représenter le « quoi » de règles métier, plutôt que le « comment », c'est-à-dire le sens des règles métier, et non leur exécution. SBVR n'est pas un formalisme exécutable, il s'adresse particulièrement aux experts métier. Il utilise le langage que tous les experts métier comprennent : le langage naturel. La mise en œuvre de données SBVR se fait en deux étapes : la mise en place d'un vocabulaire métier et l'écriture des règles métier qui se basera sur les termes et les concepts définis dans le vocabulaire.

Notre manuscrit de thèse contient de nombreuses formulations logiques en SBVR. Elles sont entièrement écrites en utilisant des polices particulières répertoriées ci-dessous. Ces polices ont été désignées par l'OMG pour représenter les différentes parties d'une règle SBVR. Plus précisément, il y a quatre styles de polices ayant un sens formel :

- La police « **terme** » est utilisée pour la désignation d'un concept, qui fait partie d'un vocabulaire utilisé ou défini (par exemple, le terme **industriel**). Ce style est appliqué à la désignation où il est défini et où il est utilisé.
- La police « **Nom** » est utilisée pour la désignation d'un nom. Cette police est souvent utilisée pour désigner les noms propres (par exemple, **université de Nice**, **CSTB**).
- La police « ***verbe*** » est utilisée pour les désignations de types d'information – le plus souvent un verbe, une préposition ou une combinaison des deux. Une telle désignation est définie dans le contexte d'une forme d'expression.
- La police « **mot-clé** » est utilisée pour les symboles linguistiques utilisés pour construire des énoncés. Il y a des mots-clés dans SBVR qui sont d'un grand intérêt pour la construction des règles. Ces mots-clés sont combinés avec des termes et des verbes pour créer des règles. La catégorisation des mots-clés est la suivante :
    - Mots-clés de quantification :
        - **Chacune**
        - **Au moins un/plusieurs**
        - **Tout au plus**
        - **Exactement plus d'un/une**
        - **Au moins (X) et au plus (Y)**
        - **Certains**
    - Mots-clés modaux :
        - Mots-clés modaux intégrés :
            - Opératifs :
                - **Il faut**

                - **Ne doit pas**
                - **Peut**
            - Structurels :
                - **Toujours**
                - **Jamais**
                - **Parfois**
                - **Peut**
        - Mots-clés modaux préfixes :
            - Structurels :
                - **Il est nécessaire que**
                - **Il est impossible que**
                - **Il est possible que**
            - Opératoires :
                - **Il est obligatoire que**
                - **Il est permis que**
                - **Il est interdit que**
- Mots-clés logiques :
    - **Ce n'est pas le cas que (X)**
    - **(X) et (Y)**
    - **(X) ou (Y)**
    - **(X) ou (Y), mais pas les deux**
    - **Si (X) alors (Y)**
    - **(Y) si (X)**
    - **(X) si et seulement si (Y)**
    - **Pas (X) et (Y)**
    - **Ni (X) ni (Y)**
- Mots-clés autres :

    « **un** », « **une** », « **le** », « **autre** », « **qui** », « **quoi** », etc.

La construction de règles SBVR se fait en deux étapes. Dans la première, il faut d'abord spécifier leur vocabulaire contrôlé. Ce vocabulaire est représenté par les termes métier. Dans la seconde, les règles candidates pour une formalisation SBVR sont réécrites d'une façon claire et précise, sans perdre la sémantique qu'elles reflètent, et en tenant compte des composants (termes, relations, faits) du vocabulaire contrôlé.

5.2 SPARQL et SPIN

Le langage SPARQL a été utilisé dans plusieurs travaux pour représenter des règles. En effet, une requête de la forme CONSTRUCT, conçue pour construire un nouveau graphe RDF à partir d'un graphe RDF existant, peut être vue comme une règle d'inférence appliquée en chainage avant, le graphe existant étant la prémisse de la règle et celui à construire étant la

conclusion. Plus généralement, le langage SPARQL peut donc être utilisé pour représenter des règles dont les prémisses et conclusion sont des patrons de graphes, c'est-à-dire des graphes RDF comportant des variables. La **Figure 2** montre la représentation SPARQL d'une règle permettant d'inférer une relation de grand-parentalité à partir de relations de parentalité :

```
CONSTRUCT {
  ?this ex:grandParent ?grandParent .
}
WHERE {
  ?parent ex:child ?this .
  ?grandParent ex:child ?parent .
}
```

Figure 2 Exemple d'une règle représentée en SPARQL[13]

Dans la continuité de ce mouvement, le langage SPIN (SPARQL Inferencing Notation) a été conçu pour représenter en RDF des requêtes SPARQL. Cela permet de façon uniforme d'interroger (en SPARQL) le contenu de règles ainsi représentées mais aussi d'éventuelles annotations (RDF) sur ces règles. SPIN utilise un vocabulaire contrôlé permettant de décrire les requêtes SPARQL en RDF. Certaines propriétés de ce vocabulaire permettent notamment de spécifier l'ordre d'exécution des règles et de spécifier au moteur d'inférence le nombre maximal d'exécutions d'une règle. La **Figure 3** présente l'exemple de règle précédent représentée en SPIN :

```
[ a     sp:Construct ;
        rdfs:comment "infer grandParent
        relationship"^^xsd:string ;
        sp:templates
                ([
                sp:object sp:_grandParent ;
                sp:predicate ex:grandParent ;
                sp:subject spin:_this
                 ]) ;
        sp:where([
                sp:object spin:_this ;
                sp:predicate ex:child ;
                sp:subject sp:_parent
                 ]
                 [
                sp:object sp:_parent ;
                sp:predicate ex:child ;
                sp:subject sp:_grandParent
                 ])
    ]
```

Figure 3 Règle SPARQL (Figure 2) représentée en SPIN[14]

[13] http://www.w3.org/Submission/2011/SUBM-spin-sparql-20110222/

[14] http://www.w3.org/Submission/2011/SUBM-spin-sparql-20110222/

5.3 Positionnement

Comme nous le décrivons dans le chapitre 3, nous représentons les contraintes réglementaires par des requêtes SPARQL. Certaines contraintes ayant des conditions d'application (représentées également sous la forme de telles requêtes SPARQL) peuvent être vues comme des règles. Il ne s'agit cependant pas là de règles d'inférence pouvant être représentées par une seule requête SPARQL de la forme CONSTRUCT ; il s'agit de règles opératives, représentées par un couple de deux requêtes de la forme ASK. Nous exprimons l'association entre ces deux requêtes dans la description RDF d'un processus de vérification de conformité, qui associe également la représentation SBVR de la règle opérative.

Cette association d'une description RDF de règles SPARQL est proche de celle utilisée dans SPIN. Cependant, nos règles ne pourraient pas être formalisées en une seule requête SPIN, car elles sont des règles opératives composées de deux contraintes réglementaires (prémisse et conclusion) ou d'une seule.

Nous avons également privilégié l'utilisation conjointe de SBVR, pour modéliser nos contraintes règlementaires en langage semi-formel pour les raisons suivantes (Bouzidi, Faron-Zucker, Fies, and Le-Thanh, 2011) :

- SBVR représente une assurance de la qualité des textes représentés.
- Il permet d'assister les experts du domaine dans la définition d'un modèle sémantique et de structurer toutes les connaissances métier en vocabulaire et règles métier.
- SBVR facilite la représentation des connaissances du domaine de façon suffisamment claire, précise et non-ambigüe.

6 Modélisation de processus à base d'ontologies

Lin a comparé plusieurs modèles de processus basés sur des ontologies (Lin, 2008). Dans cette partie nous passons en revue un certain nombre de ces modèles ayant des motivations différentes : (1) l'ontologie BWW pour la modélisation fondamentale des systèmes d'information, (2) UEML2 comme langage intermédiaire entre les divers langages de

modélisation existants, (3) MIT process handbook pour organiser des connaissances sur les processus, (4) les ontologies TOVE pour soutenir la modélisation de processus métier, (5) PSL pour décrire des procédés de fabrication, (6) PIF pour échanger des modèles de processus métier à travers différents formats et schémas, pour décrire des services web, (7) POP pour mettre en correspondance les processus métier et les outils de modélisation, (8) une ontologique permettant de modéliser les processus de propagation des maladies infectieuses et (8) les ontologies OWL-S, WSMO et SAWDSL pour décrire et annoter des services web.

6.1 BWW (Bunge-Wand-Weber)

Bien que l'ontologie BWW ne soit pas particulièrement créée comme une ontologie de processus, elle contient les concepts utiles pour la modélisation de processus ; elle contient en effet les concepts thing, property, state, law, event, transformation, system. L'ontologie BWW a été représentée dans différentes langages, en représentation formelle, en descriptions textuelles (Opdahl and Henderson-Sellers, 2002) et en représentation semi-formelle (modèle d'entités-relations) (Rosemann and Green, 2002).

6.2 UEML (Unified Enterprise Modeling Language)

UEML vise à fournir à l'industrie un langage de modélisation unifié et extensible (Panetto et al., 2004). UEML 2.0 et UEML 2.1 sont développés dans le projet INTEROP-NoE[15]. Le travail sur UEML2 dans ce projet vise à caractériser les correspondances entre les langages de modélisation d'entreprise afin de permettre l'échange de modèles. Cette approche exige une analyse ontologique détaillée des langages de modélisation métier. L'ontologie d'UEML est basée sur l'ontologie BWW ; elle comprend de nouvelles classes, des propriétés, des états et des transformations pour la description des langages de modélisation. L'ontologie UEML est représentée en OWL et permet de représenter tous les types de processus métier à travers des extensions appropriées.

[15] http://www.interop-noe.org/

6.3 Le projet MIT process handbook

Le MIT process handbook a développé de riches bibliothèques en ligne pour le partage et la gestion de nombreux types de connaissances métier. L'activité essentielle du MIT process handbook est d'organiser les connaissances liées à la représentation de processus dans des librairies spécifiques. Deux notions spécifiques sont abordées : la spécialisation des processus basée sur les notions d'héritage de la programmation orientée objet et la gestion des dépendances. Différents types de spécialisations et différents types de dépendances sont identifiés. Le MIT process handbook (Malone et al., 2003) contient des modèles génériques d'activités commerciales typiques (par exemple l'achat, la fabrication et la vente) ainsi que d'autres exemples de cas spécifiques de pratiques commerciales. Les connaissances relatives aux processus sont classées et représentées en taxonomies de processus métier.

La sémantique n'est pas formellement définie, mais elle peut être représentée en PIF (Lee et al., 1998) afin de partager les descriptions de processus.

6.4 TOVE (TOronto Virtual Enterprise)

Les ontologies TOVE[16] sont des représentations des bonnes pratiques métiers d'une entreprise à caractère commercial ou public. TOVE s'étend sur l'activité, l'état, le temps et la causalité. Elle comporte : (I) une ontologie générique décrivant les propriétés des ressources de l'entreprise, telles que les machines, l'électricité ou les matières premières, les capitaux, les compétences humaines et les informations, (II) une ontologie des structures d'organisation couvrant les rôles de la communication, (III) une ontologie de produits qui inclut des fonctionnalités, des paramètres, des assemblages, des versions et des révisions. Elle comprend également un ensemble d'extensions pour couvrir des concepts tels que le coût et la qualité.

Cependant, la principale composante de TOVE est la définition des classes et propriétés des processus (Grüninger et al., 2000).

KIF (Knowledge Interchange Format)[17] est le langage de représentation du vocabulaire logique de l'ontologie TOVE. La partie non logique du vocabulaire se compose d'expressions (constantes, fonctions, relations) qui se réfèrent à des concepts dans des domaines particuliers.

[16] http://www.eil.utoronto.ca/enterprise-modelling/tove/index.html
[17] http://www-ksl.stanford.edu/knowledge-sharing/kif/

6.5 PSL (Process Specification Language)

Le projet PSL[18] de du NIST aborde la question de l'interopérabilité en créant un langage standard pour la spécification de processus afin d'intégrer des applications de processus multiples (Grüninger and Menzel, 2003). PSL définit un langage de processus, il fournit aussi une caractérisation mathématique de la sémantique de sa terminologie ainsi que des axiomes pour l'interprétation des termes d'ontologie.

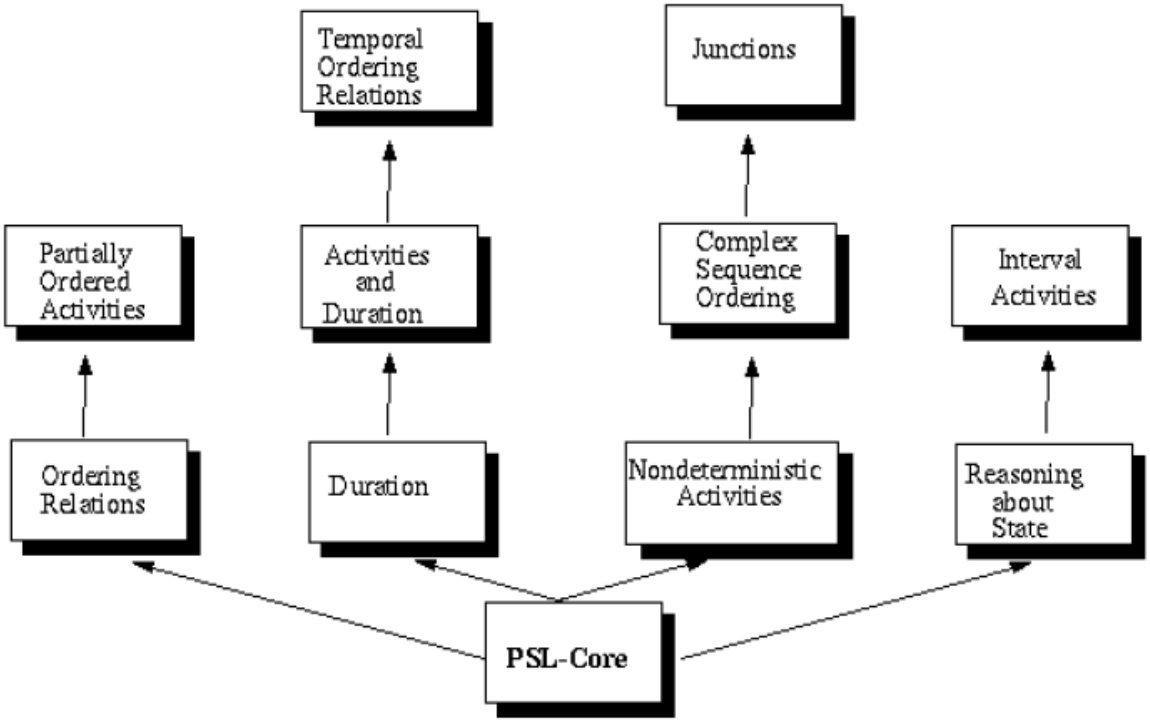


Figure 4 Modules PSL pour les classes génériques d'activités et leurs relations

PSL-Core (**Figure 4**) (Grüninger and Menzel, 2003) spécifie la sémantique des primitives de l'ontologie PSL correspondant à une représentation formelle des activités. Il comprend quatre classes principales : Object, Activity, ActivityOccurrence et Timepoint, et quatre relations principales : Participates-in, Before, BeginOf, EndOf. D'autres extensions peuvent être ajoutées à PSL-Core, telles que les rôles des ressources, des actions du processus.

[18] http://www.mel.nist.gov/psl/psl-ontology/pslcore_page.html

6.6 PIF (Process Interchange Format)

PIF (Lee et al., 1998) vise à développer un format d'échange facilitant l'échange automatique entre une grande variété de systèmes de modélisation des processus métier et des systèmes de support tels que les logiciels de workflow, les outils de représentation graphique de flux, les systèmes de simulation de procédés, et des référentiels de processus. PIF est un langage de modélisation de processus formel qui adopte la syntaxe de KIF (Lee et al., 1998).

Les principales classes et leurs relations de PIF sont présentées dans la (**Figure 5**) (Lee et al., 1998). Il y a trois classes de base : Activity, Actor et Resource. D'autres classes sont liées à ces classes principales à travers des relations. PIF soutient le point de vue structurel par la relation de décomposition de l'activité.

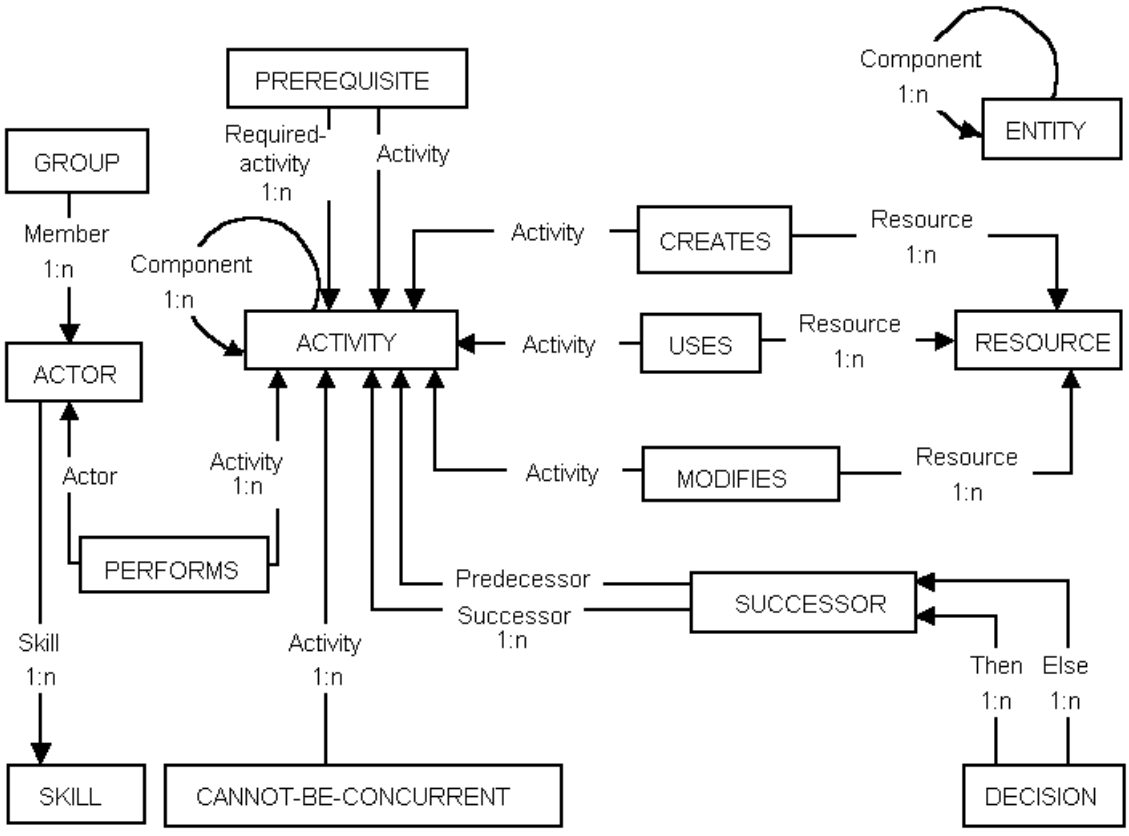


Figure 5 Les classes et relations PIF

Le projet PIF a été fusionné avec le projet PSL au NIST, de sorte que PIF a été incorporé dans le noyau PSL et ses extensions.

6.7 POP (Process, Organization, Product & others)

POP est une contribution du projet européen ATHENA[19]. C'est un format d'échange contenant un ensemble de modèles de base. Il y a cinq dimensions dans POP : les processus, les organisations, les produits, les décisions et les infrastructures.

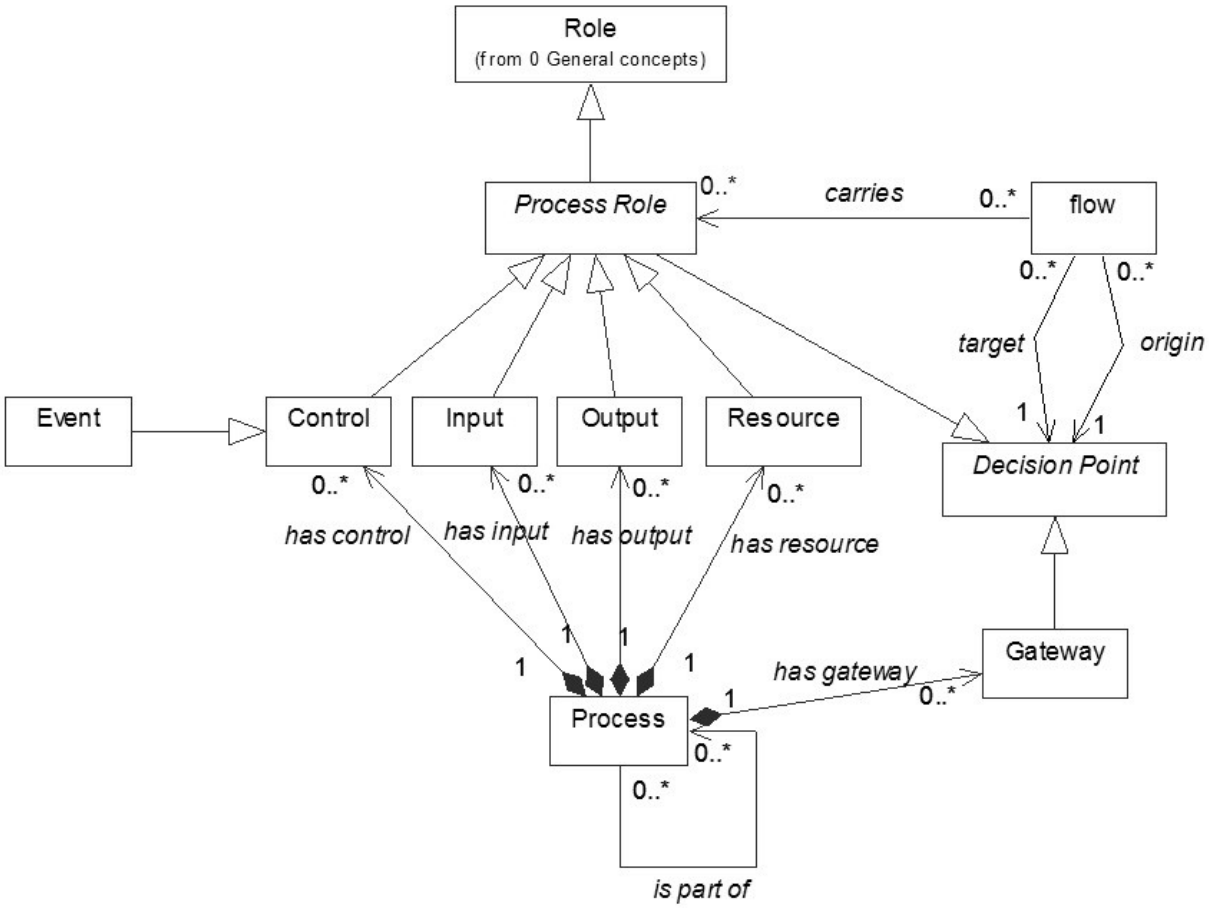


Figure 6 Méta-modèle POP

La **Figure 6** présente le méta-modèle POP (Berio, 2003) en diagramme de classes UML pour spécifier les concepts et leurs interrelations. Par conséquent, POP peut être décrit dans le profil UML 2.0 comme une implémentation des langages de modélisation.

[19] http://www.athena-ip.org/

6.8 Ontologie des processus de propagation des maladies infectieuses

Dans (Camara et al., 2012), les auteurs présentent une approche ontologique permettant de modéliser les processus de propagation des maladies infectieuses. Ils ont d'abord procédé à une analyse, à l'issue de laquelle ils ont identifié trois dimensions : l'échelle de la population, l'échelle des individus et l'échelle biologique. Chacune des dimensions comprend un ensemble de processus déterminant la propagation d'une maladie dans une population. Cette analyse a permis de diviser la population en plusieurs états distincts : les *Susceptibles*, les *Infectés* et les *Rétablis* (SIR), l'I*mmunité à la naissance* (M) qui précède la susceptibilité et l'*Exposition* (E) entre la susceptibilité et l'infection.

Les auteurs ont ensuite entamé la construction d'une ontologie de propagation des maladies infectieuses structurée sur trois couches ; une ontologie fondationnelle, une ontologie noyau et des ontologies spécifiques :

- L'ontologie fondationnelle détermine les concepts (et leurs relations), les processus, évènements, les objets et les états. Ces définitions abstraites seront ensuite réutilisées pour modéliser l'ontologie principale.
- L'ontologie de noyau contient les concepts généraux et communs au domaine et les relations qui les lient. L'auteur utilise l'ontologie des maladies infectieuses « The Infectious Disease Ontology » (IDO-Core) [20]. Cette ontologie couvre les concepts généraux du domaine des maladies infectieuses et leurs relations. IDO est rattachée à l'ontologie de fondement « Basic Formal Ontology » (BFO) [21].
- Les ontologies spécifiques, ou ontologies de sous-domaines, modélisent des aspects particuliers des maladies infectieuses. L'auteur utilise dans cette partie deux ontologies : l'ontologie du domaine de la propagation (les aspects statiques) et l'ontologie du processus de propagation (les aspects dynamiques).

Cette approche a été appliquée à la modélisation du processus de propagation de la bilharziose au Sénégal. L'utilisation d'un tel modèle pour prévoir et contrôler la propagation des maladies infectieuses constitue un défi prometteur pour les systèmes de veille épidémiologique.

[20] http://infectiousdiseaseontology.org/page/Main_Page
[21] http://www.ifomis.org/bfo

6.9 SAWSDL, OWL-S et WSMO

Il existe plusieurs langages pour annoter sémantiquement des services web dans le but de rendre leur contenu compréhensible par une machine et ainsi d'améliorer la découverte de services ou la composition de services.

• OWL-S[22] est un langage permettant de spécifier les architectures de services web avec des métadonnées sémantiques basées sur OWL (Toch et al., 2005). Les motivations des applications d'OWL-S sont l'automatisation des processus de découverte, d'invocation et de composition des services web et l'interopérabilité. Toutefois, OWL-S est critiqué pour son ambigüité conceptuelle, son manque concis d'axiomatisation due à une conception vague (Mika et al., 2004)

• WSMO[23] (Dimitrov et al., 2007) est un modèle conceptuel pour décrire les divers aspects liés aux services web sémantiques. L'objectif de WSMO est de résoudre le problème de l'intégration d'applications distribuées. Il suit le standard MOF (Meta-Object Facility).

• SAWSDL (Kopecký et al., 2007) est une recommandation du W3C qui vise à fournir un mécanisme d'annotation pour ajouter de la sémantique aux descriptions WSDL de services web. Le mécanisme d'annotation est accompli à travers des attributs de WSDL pour faire référence à des concepts de modèles sémantiques. Dans la spécification SAWSDL, plusieurs annotations sémantiques peuvent être associées à des éléments WSDL, aucune restriction n'est faite sur le langage de formalisation de l'ontologie.

6.10 Positionnement

Nous avons présenté dans cette partie différents modèles de processus. Certains d'entre eux sont spécifiés comme un méta-modèle de langage de modélisation de processus, tels que PIF, POP. BWW et TOVE fournissent des modèles de processus basés sur des ontologies et des taxonomies normalisées. Egalement, G. Camara propose une ontologie construite dans le but de modéliser les processus de propagation de maladies infectieuses, afin de les contrôler. WSMO, OWL-S et SAWSDSL sont des langages pour annoter sémantiquement des services

[22] http://www.w3.org/Submission/OWL-S/
[23] http://www.wsmo.org/

web pour augmenter l'expressivité de leur description et améliorer ainsi la découverte ou la composition de services.

Dans nos travaux nous proposons un modèle déclaratif de processus de vérification de contraintes basé sur une ontologie légère RDFS et nous utilisons un moteur de processus qui interprète ces représentations RDF de processus de vérification. Le moteur repose sur un modèle de processus, qui permet de décrire une séquence d'exécution de vérification de contraintes dans laquelle il s'agit d'exécuter séquentiellement une ou plusieurs requêtes (Bouzidi, Faron-Zucker, Fies, and Le-Thanh, 2011). Nous détaillons ce modèle dans le chapitre 4.

7 Conclusion

Les recherches sur la modélisation et les représentations des contraintes réglementaires dans le domaine du bâtiment ont été menées en parallèle des recherches sur le contrôle des contraintes de conformité à satisfaire.

Les premiers systèmes de contrôle de la conformité des projets de construction ont été orientés sur la vérification des aspects d'un bâtiment : ses dimensions physiques et géométriques, ses caractéristiques topologiques, etc. Cette vérification a été principalement basée sur la modélisation des contraintes de conformité issues de la réglementation.

De ce fait, la modélisation réglementaire est devenue un enjeu important. Cette problématique a été abordée jusqu'ici sous deux approches différentes. La première vise à analyser les réglementations automatiquement et à se confronter à la complexité du langage naturel : projet E-POWER, travaux de (Lau et al., 2005), (Bolioli et al., 2002), (Turk et al., 2002).

Dans la seconde, les contraintes réglementaires sont écrites de manière normée avec l'aide d'experts du domaine, ce qui facilite leur traduction en langage formel (Reeder et al., 2007).

Notre approche constitue une troisième voie, elle prend en compte la réglementation écrite en langage naturel et celles écrites de manière normée. Nous représentons en RDF l'association d'une règle SBVR et de requêtes SPARQL représentant la prémisse et la conclusion d'une même contrainte réglementaire. En effet, avec l'aide d'experts du domaine, nous extrait de réglementations textuelles des règles opératives en SBVR, basées sur un vocabulaire contrôlé, avant de les formaliser en SPARQL.

Par ailleurs, divers efforts ont été entrepris pour appliquer contrôler la conformité de bâtiments, en utilisant un modèle de représentation des bâtiments : IFC, (Yurchyshyna, 2009), (Pauwels et al., 2011), (Eastman et al., 2009), CORNET, AC3, HITOS. D'autres visent à déterminer pour des conditions données du bâtiment l'applicabilité des différentes réglementations (Kerrigan et al., 2003).

Dans nos travaux, nous effectuons des vérifications automatiques sur des dossiers techniques afin de les valider au regard des réglementations en vigueur. Pour cela, nous avons organisé en différents processus les requêtes SPARQL représentant des contraintes réglementaires. Nous proposons un modèle déclaratif de processus de vérification de contraintes basé sur une ontologie légère RDFS et nous utilisons un moteur de processus qui interprète ces représentations RDF de processus de vérification (Bouzidi, Faron-Zucker, Fies, Zarli, et al., 2012).

CHAPITRE 2
MODÉLISATION DU PROCESSUS DE RÉDACTION DES DOSSIERS TECHNIQUES

1 Processus courant de rédaction d'un dossier technique 50
2 Modélisation du dossier technique 52
2.1 Analyse des besoins 53
2.2 Collecte des données 54
2.2.1 L'interview d'experts 54
2.2.2 L'utilisation de corpus textuel 55
2.2.3 Thesaurus REEF 55
2.3 Création d'une ontologie légère 56
2.3.1 Alignement de l'ontologie du Dossier technique avec celle du REEF 57
2.3.2 Organisation hiérarchique des concepts non issus du thesaurus REEF 58
2.4 Définition de concepts 58
2.5 Résultat 60
3 Modélisation du processus de rédaction d'un dossier technique 61
3.1 Identification des étapes du processus de rédaction d'un dossier technique 61
3.2 Modélisation de la dynamique du processus de rédaction d'un dossier technique 62
4 Conclusion 66

1 Processus courant de rédaction d'un dossier technique

Les avis techniques sont rédigés par le CSTB à la demande d'un industriel. L'industriel commence par envoyer une demande d'avis technique aux services concernés au sein du CSTB. Les instructeurs du CSTB lui transmettent alors le règlement de l'avis technique ainsi qu'une trame pour l'élaboration du dossier technique. Il s'agit tout simplement d'un fichier Word contenant des chapitres, du texte ainsi que des instructions. Cette trame contient toutes les informations nécessaires pour qu'un procédé, produit ou matériau soit étudié par un groupe spécialisé qui aura la charge de délivrer l'avis technique. L'industriel est sensé renseigner la trame et envoyer un document complet au CSTB qui transmet en retour un devis d'avis technique à l'industriel. Une fois le devis accepté par l'industriel, les instructeurs du CSTB procèdent à la vérification du contenu du dossier technique et peuvent demander à l'industriel de justifier les éléments renseignés dans son dossier par des essais expérimentaux qui serviront par exemple de preuve de la résistance du produit aux différents risques extérieurs. Des visites de chantier et/ou d'usine peuvent être planifiées pour s'assurer du bon déroulement des procédures de fabrication et/ou d'installation du produit. Pour être valide, le dossier technique doit finalement contenir une description détaillée du procédé et le domaine d'emploi visé par le demandeur, les prescriptions de mise en œuvre, les résultats expérimentaux, les références d'emploi.

Une fois le dossier complété, les instructeurs du CSTB le transfèrent au groupe spécialisé concerné. Ce dernier a la tâche de vérifier la conformité des informations selon les normes en vigueur et de délivrer un avis. Les instructeurs du CSTB remettent alors en forme l'avis technique et le publient sur le site du CSTB et en transmettent une copie à l'industriel.

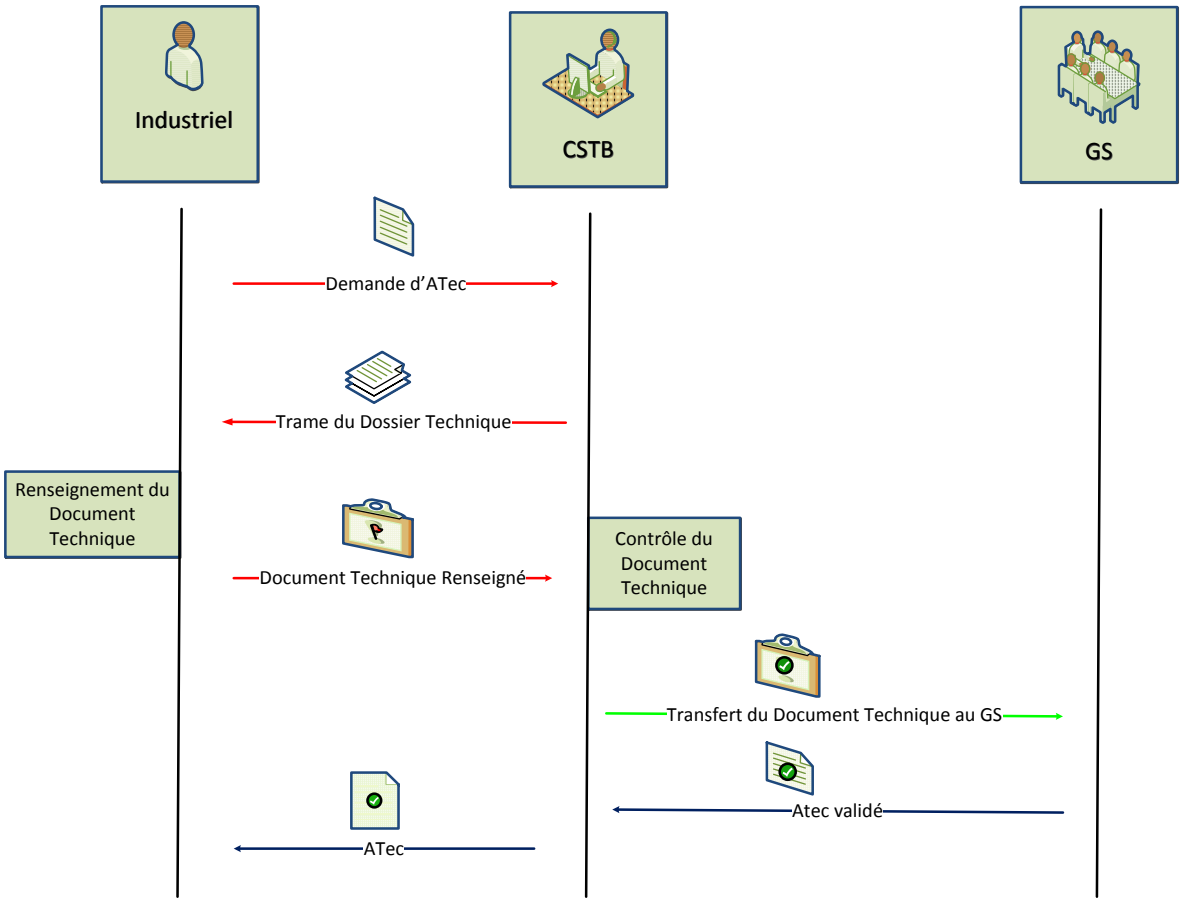


Figure 7 Processus de création d'un avis technique

Ce processus que nous venons de décrire et qui est résumé sur la **Figure 7** est idéal. Dans la pratique, il n'est bien souvent pas respecté. A travers nos échanges avec les experts du CSTB, il apparait très clairement que les industriels n'arrivent pas à renseigner correctement la trame du document. En effet, la trame du Dossier Technique est un long formulaire Word général pour un domaine précis. Nous nous sommes rendu compte lors de notre étude de la trame du groupe spécialisé Photovoltaïque (GS21), que celle-ci contenait des informations communes à tous les produits qui peuvent être utilisés dans le domaine du photovoltaïque. De ce fait, les industriels n'arrivent pas à la renseigner correctement pour un produit ou procédé particulier.

De ce fait, un dialogue s'instaure entre le CSTB et l'industriel demandeur, de trop nombreux échanges sont nécessaires avant d'arriver à une version acceptable du dossier technique ; le processus est mal anticipé et jugé trop administratif. Comme conséquence directe, cela conduit à un traitement des dossiers long, de l'ordre de 6 à 8 mois. Sans accompagnement du CSTB, la plupart des demandeurs d'avis technique ne parviennent pas à remplir correctement leur dossier technique. Or le CSTB souhaite ne pas mobiliser de ressources hors d'un quelconque devis. Idéalement, le rôle du CSTB devrait se limiter à vérifier le contenu de ce

dossier et non à assister l'industriel et l'industriel devrait prendre en charge la tâche de renseigner la trame du dossier technique correctement, avant de l'envoyer au CSTB.

2 Modélisation du dossier technique

Nous proposons de modéliser un dossier technique et le processus de sa rédaction selon une approche basée sur des ontologies afin de construire un modèle sémantiquement riche du dossier technique. Nous avons étudié les documents techniques émis par le groupe spécialisé photovoltaïque du CSTB dans le but de construire une ontologie du domaine qui capitalise le savoir-faire des experts. Cette ontologie vise à modéliser un dossier technique du domaine pour faciliter la création de nouveaux dossiers techniques.

Diverses méthodes ont été proposées et développées pour construire des ontologies. Nous utilisons dans nos travaux une méthode de construction manuelle d'ontologies à partir de textes et d'interviews. De nombreux travaux ont été entrepris pour répondre à la problématique de la construction manuelle d'ontologies, notamment Methontology (Corcho et al., 2003), On-To-Knowledge (Noy and Mcguinness, 2008; Staab et al., 2001), ainsi que d'autres décrits dans le Chapitre 1 (section 3.2). Cette méthode de construction d'ontologie s'applique particulièrement à un domaine (ex. le domaine du bâtiment) englobant plusieurs sous-domaines (tels que le domaine du photovoltaïque, du thermique, de l'accessibilité, etc.).

L'idée commune à ces solutions est de réutiliser autant que possible des ontologies existantes liées au domaine à modéliser. C'est l'approche que nous avons choisi d'adopter : nous n'abordons pas le problème de l'extraction de connaissances à partir de texte avec des techniques de TALN, nous construisons une ontologie à partir de connaissances recueillies lors de l'étude de documents techniques et d'interviews d'experts du CSTB et en réutilisant des ressources ontologiques existantes. Plus précisément, avec l'aide d'experts du domaine photovoltaïque, nous avons identifié les connaissances à représenter pour modéliser un dossier technique de ce domaine et à quel niveau de granularité.

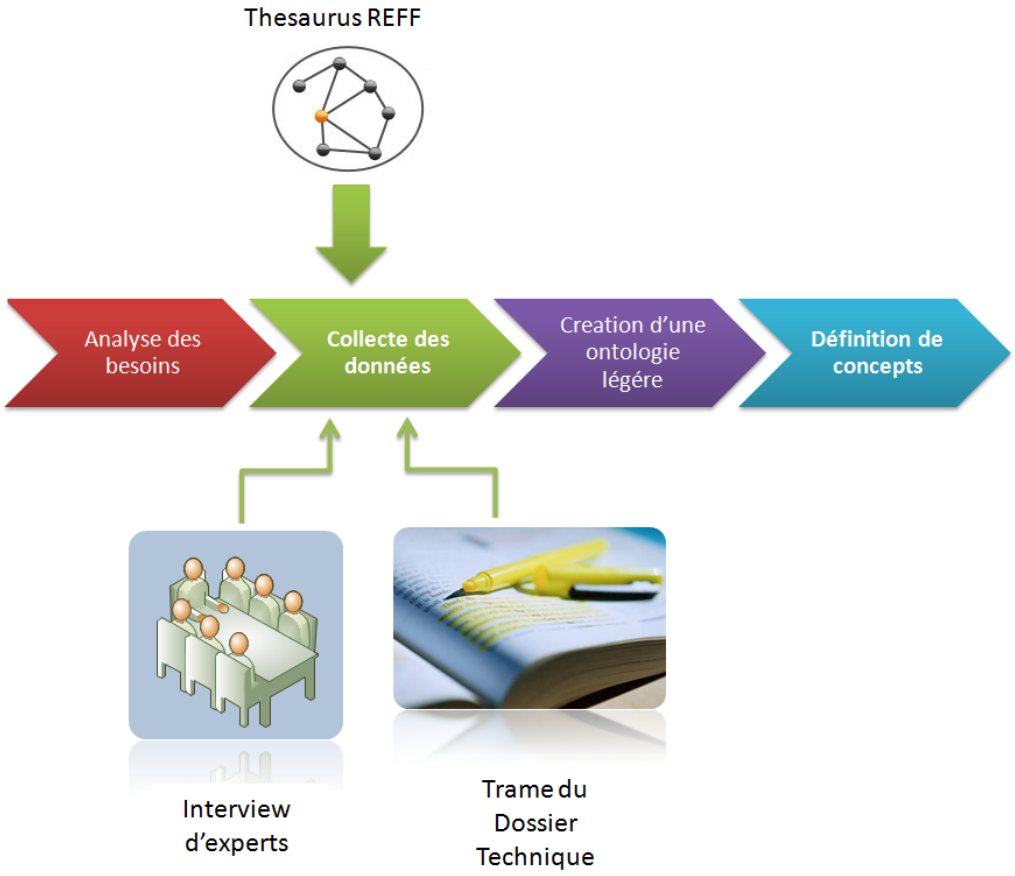


Figure 8 Etapes de construction d'une ontologie du dossier technique

Les documents techniques contiennent des éléments linguistiques qui peuvent être utiles pour la construction d'ontologies, notamment les termes candidats, les relations et les axiomes. Notre méthode de construction d'ontologie comporte les étapes décrites ci-après. Elle est résumée sur la **Figure 8**.

2.1 Analyse des besoins

Dans cette étape, nous avons identifié nos objectifs opérationnels en identifiant le but d'utilisation de l'ontologie à construire ainsi que les questions auxquelles elle devra permettre de répondre.

OntoDT est une ontologie du domaine du bâtiment, destinée à être utilisée principalement par un système d'aide à la rédaction de documents réglementaires, que nous avons construit plus précisément pour guider le processus de rédaction de documents techniques et la rédaction d'avis techniques sur ces dossiers.

A travers ce système d'aide, OntoDT est utilisée de façon transparente par les industriels qui renseignent des formulaires pour construire leur dossier technique. Elle permet ainsi au système d'aide de guider l'industriel dans sa production d'un dossier technique en lui posant à chaque étape les bonnes questions, en fonction de ce qu'il a renseigné aux étapes précédentes.

OntoDT est ensuite utilisée par un système d'aide à la rédaction d'avis techniques sur les dossiers techniques et plus précisément pour automatiser le contrôle de la conformité d'un dossier technique par rapport aux textes réglementaires en vigueur.

2.2 Collecte des données

C'est l'étape de conceptualisation dans la construction d'une ontologie. Selon (Gómez-Pérez et al., 1996) il existe plusieurs techniques pour l'effectuer : les interviews avec des experts du domaine, et/ou l'utilisation de corpus textuels. Dans note cas nous avons utilisé les deux méthodes conjointement.

2.2.1 L'interview d'experts

Deux entretiens avec les experts du GS 21, le groupe spécialisé du domaine du photovoltaïque, nous ont permis d'obtenir une ébauche de spécification ainsi qu'un inventaire détaillé des termes candidats, pour identifier les concepts et relations d'OntoDT.

La première interview a eu lieu à mi-étape de la construction d'ontologie. Elle a permis (1) de clarifier la structuration des connaissances du domaine, à l'aide de la connaissance des experts sur le renseignement du dossier technique, indépendamment des contraintes réglementaires et (2) de clarifier les connaissances du domaine photovoltaïque à modéliser. Les explications des expert métier on porté sur :

(I) les différents composants entrant dans la fabrication d'un module photovoltaïque,

(II) les différentes étapes essentielles au renseignement d'un dossier technique,

(III) les différentes parties obligatoires qu'un industriel doit renseigner,

(IV) les différents types de modules photovoltaïques et leurs nomenclatures,

(V) les contraintes structurelles et dimensionnelles à représenter et leur niveau de granularité.

La seconde interview a permis de valider l'ontologie construite.

La seconde interview a été effectuée une fois l'ontologie OntoDT construite et nous a permis de valide celle-ci. A l'issue de cette interview, nous avons été amenés à modifier, légèrement, la structuration de l'ontologie proposée. En effet, avec l'aide des experts du domaine, nous avons ajouté de nouveaux concepts permettant de factoriser des concepts, déjà existant, issus de la trame du dossier technique.

2.2.2 L'utilisation de corpus textuel

Cette étape est indépendante ou complémentaire de la précédente (Condamines, 2005). Dans notre cas, elle se révèle cruciale car c'est à partir de la trame du dossier technique que nos connaissances sont extraites. Mais seul un travail en collaboration avec les experts du domaine a pu apporter un résultat cohérant.

Nous avons d'abord conduit une analyse informelle de la trame du dossier technique afin d'identifier et d'extraire les principaux termes rencontrés et ébaucher une première modélisation. Ensuite, nous avons formellement choisi les termes à retenir et identifié les concepts de notre ontologie. Cette étape a nécessité l'intervention d'experts afin de fournir une terminologie fiable et rigoureuse commune à tous les acteurs.

Cette phase d'extraction et d'analyse de termes candidats peut se faire à l'aide d'outils de traitement de la langue tel que par exemple TERMINAE[24] qui implante une méthodologie permettant d'extraire les concepts ainsi que leurs propriétés et relations. Cependant, étant donnée la taille limitée de notre corpus textuel nous avons réalisé cette étape manuellement.

2.2.3 Thesaurus REEF

Nous avons ensuite comparé les termes que nous avons extraits avec les termes du thésaurus du domaine de construction mis au point pour le REEF, l'encyclopédie complète des textes techniques et réglementaires de la construction développée par le CSTB. Les ingénieurs de la connaissance et les documentalistes du CSTB ont travaillé, dans le cadre du REEF, sur l'établissement d'une terminologie unifiée pour décrire et rechercher l'information technico-réglementaire. Ils reposent pour cela sur le modèle SKOS[25]. La **Figure 9** suivante présente un extrait du thesaurus REEF.

[24] http://lipn.univ-paris13.fr/fr/article/1314-terminae/166
[25] http://www.w3.org/2004/02/skos/

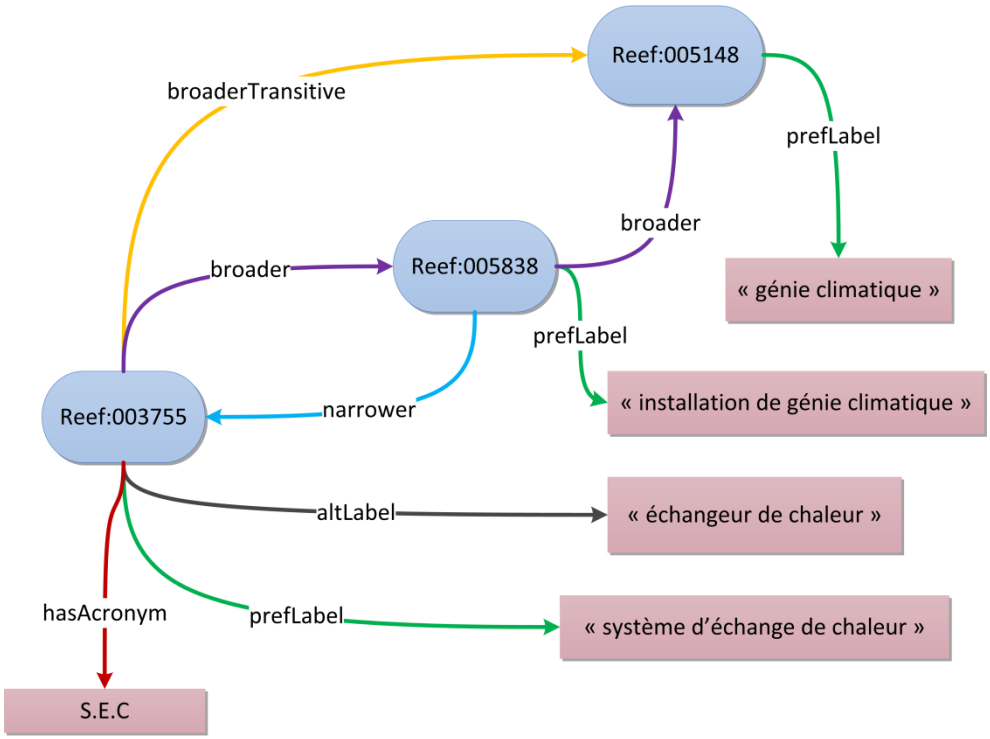


Figure 9 Extrait du thesaurus REEF

L'avantage de réutiliser le thésaurus du CSTB est non seulement de nous permettre de lier/intégrer facilement les avis techniques dans le REEF (Bus et al., 2009), mais aussi d'exploiter toute la richesse et la sémantique exprimée dans cette taxonomie et d'avoir une base pour construire notre ontologie du bâtiment.

Nous avons ainsi adopté une approche similaire à celle de (Hernandez and Mothe, 2006) : réutiliser un thésaurus de domaine qui exige des efforts de conception importants pour le développement de nouvelles ressources à un niveau formel plus élevé. Ce principe est intéressant dans la mesure où il nous évite la construction d'une nouvelle ontologie ex nihilo. La conception d'ontologies à partir de thésaurus présente l'avantage de considérer tous les termes identifiés par les experts comme étant représentatifs du domaine.

2.3 Création d'une ontologie légère

A ce stade, l'ontologie créée comporte les concepts identifiés comme pertinents pour la création de documents techniques mais est encore plate. Il s'agit de regrouper et d'organiser hiérarchiquement les concepts. Après avoir identifié les concepts et relations de l'ontologie, nous avons donc défini sa structure générale en utilisant les relations hiérarchiques dans le thesaurus REFF et en organisant les concepts non issus du REEF avec l'aide des experts du CSTB.

2.3.1 Alignement de l'ontologie du Dossier technique avec celle du REEF

L'ontologie qui peut être construite à partir du thesaurus REEF est légère. En effet, le REEF est composé d'une arborescence de concepts du domaine du bâtiment, liés entre eux par des relations « narrower» et « broader » (**Figure 9**). Cette ontologie contient des concepts généraux de l'industrie du bâtiment et il lui manque des termes spécifiques de l'industrie photovoltaïque. Ces termes sont extraits à partir de la trame du dossier technique et des résultats des interviews effectuées avec les experts du CSTB. L'exploitation du thesaurus REEF comprend ainsi trois étapes :

I. Extraction de relations hiérarchiques explicites dans le thesaurus REEF. Tout d'abord, nous avons organisé hiérarchiquement les concepts issus du thésaurus REEF sur la base des relations SKOS *narrower* et *broader* entre les termes du thesaurus. Ils sont sélectionnés comme candidats pour une relation RDFS *subClassOf*.

II. Elimination des redondances dans les relations hiérarchiques du thésaurus. N'étant pas formalisés, des redondances dans la structure du thésaurus peuvent exister. Notre ontologie peut être confrontée à la transitivité de type : si A est plus spécifique que B et B plus spécifique que C, alors A est plus spécifique que C. A, B, C étant des concepts, la relation hiérarchique entre A et C ne doit pas être explicite. Pour supprimer ce type de relations redondantes, nous avons vérifié la pertinence de chaque relation *subClassOf*.

III. Fusion de l'ontologie extraite du REEF avec l'ontologie OntoDT en construction.

La fusion d'ontologies consiste en la création d'une nouvelle ontologie à partir de deux ontologies ou plus. L'ontologie résultante unifie et remplace les ontologies d'origine. Les approches les plus courantes utilisent l'union ou l'intersection pour relier les ontologies résultantes à l'ontologie originale. Dans l'approche par union, l'ontologie résultante contient l'union des entités provenant des ontologies originales et suppose résolues les différences de représentation d'un même concept. Dans l'approche par intersection, l'ontologie résultante ne contient que les parties communes des ontologies originelles.

Dans notre cas, nous avons réalisé l'union de l'ontologie OntoDT avec l'intersection de celle-ci avec l'ontologie du REEF : tous les concepts d'OntoDT sont conservés dans l'ontologie

résultat et seuls nous intéressent les concepts du REEF qui peuvent être alignés avec des concepts d'OntoDT.

Plusieurs approches et outils pour la fusion d'ontologie ont été proposés dont PROMPT (Noy and Musen, 2000), (McGuinness et al., 2000) que nous avons utilisé. Cet outil identifie un ensemble d'opérations pour la fusion d'ontologies (fusion des classes, fusion des liens) et un ensemble de conflits possibles consécutifs à l'application de ces opérations (conflits de nom, redondance dans la hiérarchie des classes).

2.3.2 Organisation hiérarchique des concepts non issus du thesaurus REEF

Il existe pour cela trois principales méthodes de regroupement (Aimé, 2011):

- La méthode (bottom-up) consiste à agréger les classes les plus spécifiques pour créer des classes plus générales.
- La méthode (top-down) consiste à diviser les classes générales en classes plus spécifiques.
- La méthode (middle-out) combine la spécialisation et la généralisation de classes.

Nous avons adopté une méthode de regroupement bottom-up : des groupes de concepts ayant le même contexte sémantique ont été identifiés, ensuite une factorisation maximale des relations entre ces concepts a été réalisée.

2.4 Définition de concepts

Nous avons enrichi l'ontologie OntoDT en traduisant des connaissances véhiculées dans les documents techniques, exprimées de manière informelle en langage naturelle, en connaissances exploitables par la machine, représentées dans un langage formel. Plus précisément, nous avons traduit la structure de la trame du document technique en un ensemble d'axiomes définissant certains des concepts d'OntoDT. Selon (Staab and Mädche, 2000), les axiomes représentent les connaissances n'ayant pas un caractère strictement terminologique, ils permettent de fixer la sémantique des concepts et de leurs relations.

Par exemple, Le module verre polymère est composé de verre polymère, d'une cellule photovoltaïque, d'un film polymère, etc. La représentation de cette connaissance passe par l'énoncé d'un axiome. Nous avons ainsi défini les concepts d'OntoDT dans le langage OWL,

en utilisant l'éditeur d'ontologie Protégé[26]. La **Figure 10** ci-dessous montre le concept VerrePolymere défini dans l'éditeur Protégé et la **Figure 11** présente la définition de la classe VerrePolymere dans le langage OWL produite (pour une meilleure lisibilité, nous avons traduit les données RDF/XML produites par Protege en données RDF/N3) :

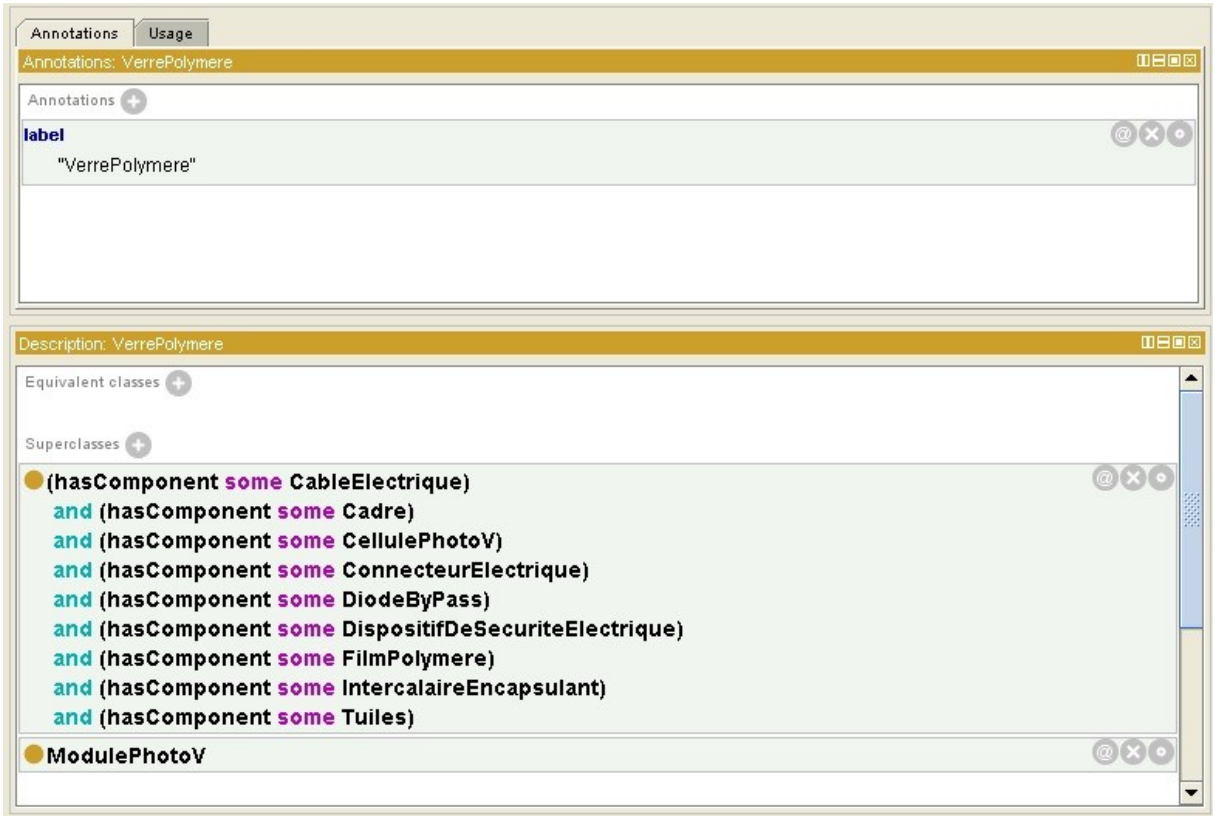


Figure 10 Définition de la classe VerrePolymere dans l'éditeur Protege

[26] http://protege.stanford.edu

```
@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix odt: <http://www.cstb.fr/ontologies/odt#> .

odt:VerrePolymere a owl:Class;
    rdfs:label "Verre Polymere";
    rdfs:subClassOf odt:ModulePhotoV
                    [ a owl:Class;
                       owl:intersectionOf (
                          [ a owl:Restriction;
                           owl:onProperty odt:hasComponent;
                           owl:someValuesFrom odt:CableElectrique ]
                          [ a owl:Restriction;
                           owl:onProperty odt:hasComponent;
                           owl:someValuesFrom odt:Cadre ]
                          [ a owl:Restriction;
                           owl:onProperty odt:hasComponent;
                           owl:someValuesFrom odt:CellulePhotoV ]
                          [ a owl:Restriction;
                           owl:onProperty odt:hasComponent;
                           owl:someValuesFrom odt:ConnecteurElectrique ]
                          ...
                          [ a owl:Restriction;
                           owl:onProperty odt:hasComponent;
                           owl:someValuesFrom odt:Tuiles ] ) ].
```

Figure 11 Définition OWL de la classe VerrePolymere

Cette définition stipule que la classe `odt:VerrePolymere` est une sous-classe de la classe `odt:ModulePhotovoltaique` et de la classe définie comme l'intersection de définitions de classes représentant chacune l'ensemble des individus ayant un composant particulier.

2.5 Résultat

OntoDT est constituée de 138 classes et 48 propriétés OWL Lite ; 27 classes sont définies. 35% de ces classes correspondent à des termes du thésaurus REEF. Les 65% restants sont des concepts plus spécifiques : dans son état actuel, il manque à REEF de nombreux termes

spécifiques au domaine du photovoltaïque. Toutefois, il demeure en constante évolution (Bouzidi, Faron-Zucker, Fies, Bourdeau, et al., 2011).

3 Modélisation du processus de rédaction d'un dossier technique

La problématique majeure du processus de rédaction de dossier technique est sa lenteur. Les industriels comprennent mal la trame du dossier technique et n'arrivent pas à la renseigner correctement. Côté CSTB, les instructeurs se retrouvent souvent avec des dossiers incomplets et sont obligés de les retourner pour des rectifications. Cette situation est coûteuse en temps et en ressource pour le CSTB ; une des tâches des instructeurs est devenue la vérification du dossier technique, pour assister les industriels.

3.1 Identification des étapes du processus de rédaction d'un dossier technique

C'est dans ce contexte que nous avons décidé de modéliser ces étapes d'interaction entre instructeur et industriel via une méthodologie puis un outil pour guider les industriels tout au long de leur processus de rédaction d'un dossier technique.

Pour cela, nous avons commencé par identifier à la fois les parties à renseigner ainsi que les types d'informations attendues. Nous avons constaté que les documents techniques ne sont pas tous identiques et que les industriels, selon le produit, ne sont pas obligés de renseigner les même parties de la trame ni toute la trame du dossier technique.

Nous avons également constaté qu'une décomposition de cette trame en plusieurs formulaires s'imposait. Ces formulaires sont liés les uns aux autres selon des contraintes spécifiques à la nomenclature des produits à renseigner, aux différents critères dimensionnels et structurels.

Chaque formulaire contient des composants qui représentent la nomenclature du même produit, le composant lui-même peut être formé d'un ou plusieurs éléments. Pour mieux expliquer, prenons l'exemple du module verre polymère. Selon la trame, ce module est composé de verre polymère, d'une cellule photovoltaïque, d'un film polymère, ainsi que d'autres composants. Chacun de ces composants possède des caractéristiques structurelles et

dimensionnelles intrinsèques. Ces composants et leurs caractéristiques sont représentés dans notre ontologie respectivement par des concepts et propriétés et nous avons défini par des axiomes les interdépendances entre les différents concepts modélisant les interactions ou interdépendance entres les pièces des modules photovoltaïques.

Nous avons ainsi obtenu une liste de formulaires relatifs aux composants des différents produits à certifier issue de la trame du dossier technique DT PV GS21. Ces formulaires sont liés entre eux selon la définition dans l'ontologie OntoDT des concepts représentant les composants qui sont décrits dans ces formulaires. Par exemple, le formulaire de description d'un verre polymère renvoie à un formulaire pour décrire une cellule photovoltaïque, à un autre pour décrire un film polymère, etc.

3.2 Modélisation de la dynamique du processus de rédaction d'un dossier technique

Le processus de rédaction d'un dossier technique que nous proposons est basé sur la modélisation des composants à décrire et permet que l'industriel n'ait plus à renseigner une trame de dossier technique générique, mais interagisse plutôt avec un système qui s'adapte aux informations qu'il renseigne, raisonne et lui propose une liste de formulaires à renseigner en adéquation avec le produit à décrire.

A l'aide d'une interface dynamique, l'industriel est guidé dans son processus de renseignement du dossier technique, une séquence de formulaires lui est présentée, adaptée dynamiquement aux informations qu'il fournit. L'industriel n'a qu'à renseigner les éléments que le système lui présente ; c'est ce dernier qui s'assure que toutes les étapes de renseignement sont respectées. L'avantage d'un tel système est une aisance au niveau du renseignement côté industriel et une meilleure compréhension des demandes côté CSTB.

Ce processus de rédaction interactive du dossier technique repose sur l'interrogation de l'ontologie ontoDT pour sélectionner les formulaires pertinents associés aux composants élémentaires du produit à décrire dans le dossier technique.

Le processus commence par la présentation d'un premier formulaire général permettant à l'industriel de définir le cadre de son dossier technique. Il s'agit à ce premier niveau de faire des choix de composant à décrire dans des listes déroulantes. Les choix proposés correspondent à des concepts de l'ontologie OntoDT dont les labels correspondent au vocabulaire défini dans le thésaurus du REEF.

A partir du composant choisi, l'ontologie est interrogée pour demander au fur et à mesure des renseignements plus précis sur les sous-composants : en fonction des composants identifiés dans les définitions de concepts de l'ontologie, un enchainement de formulaires adéquats est produit à l'utilisateur. Les renseignements fournis à travers les formulaires concernent la nomenclature du produit, les différents critères structurels et dimensionnels à renseigner. La **Figure 12** ci-dessous présente le principe général de fonctionnement de l'outil d'aide à la rédaction de dossiers techniques (Bouzidi, Faron-Zucker, Fies, Corby, and Le-Thanh, 2012).

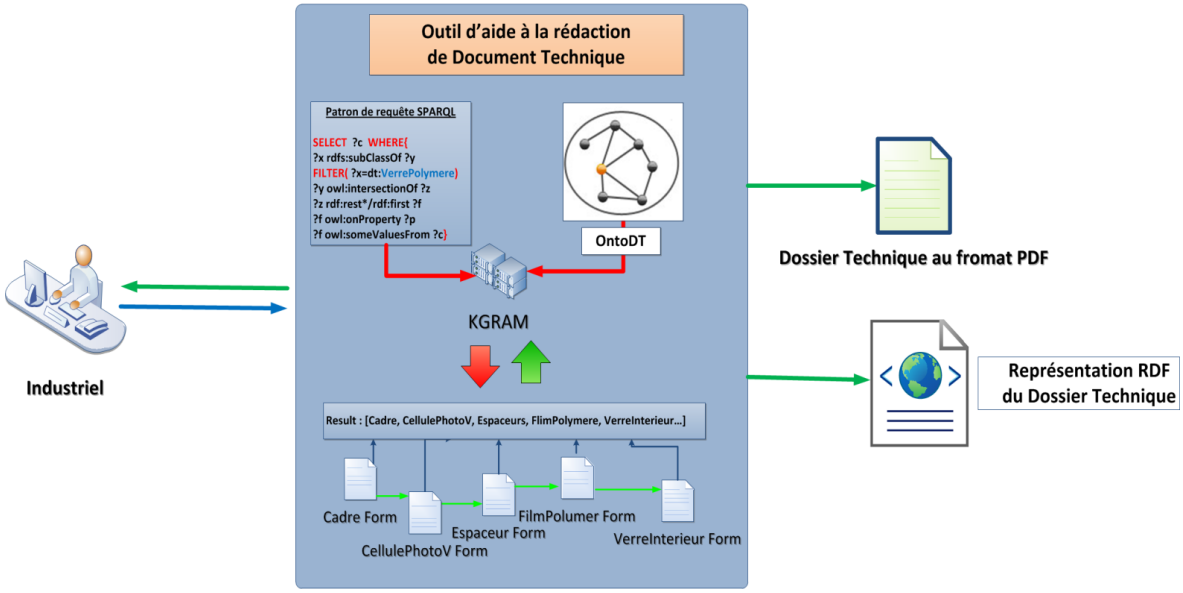


Figure 12 Outil d'aide à la rédaction de Dossier Technique

La dynamique de l'enchainement des formulaires repose sur le patron de requête SPARQL présenté sur la **Figure 13** qui est instancié à chaque étape pour interroger l'ontologie et déterminer le formulaire suivant. En effet, l'instanciation de ce patron de requête permet pour chaque composant du produit à décrire de retrouver la liste des composants (`?composant`) qui entrent dans sa définition OWL.

```
SELECT ?composant WHERE{
  ?classeAInstancier rdfs:subClassOf ?y
  ?y owl:intersectionOf ?z
  ?z rdf:rest*/rdf:first ?f
  ?f owl:onProperty odt:hasComponent
  ?f owl:someValuesFrom ?Composant}
```

Figure 13 Extrait du patron de requête SPARQL pour interroger l'ontologie OntoDT sur les classes définies

Par exemple, la requête ci-dessous (**Figure 14**) est une instanciation du patron de requête où la variable `?classeAInstancier` a été remplacée par l'URI de la classe `VerrePolymere` pour interroger OntoDT sur la définition de celle-ci (**Figure 11**). Cette définition, nous l'avons vu, stipule que la classe `VerrePolymere` est sous-classe d'une classe définie comme l'intersection des classes d'individus ayant tel ou tel composant. Le graphe de la requête est apparié à celui de la définition de la classe `VerrePolymere` en appariant la variable `?z` à l'intersection de classes et la variable `?f` à chacune des classes entrant dans cette intersection ; le chemin de propriétés `rdf:rest*/rdf:first` permet d'accéder à chacune de ces classes et la requête produit les composants de ces sous-classes, c'est-à-dire les composants de la classe `VerrePolymere`.

```
SELECT ?Composant WHERE{
  odt:VerrePolymere rdfs:subClassOf ?y
  ?y owl:intersectionOf ?z
  ?z rdf:rest*/rdf:first ?f
  ?f owl:onProperty odt:hasComponent
  ?f owl:someValuesFrom ?Composant}
```

Figure 14 Instanciation du patron de requête SPARQL pour interroger la définition de VerrePolymere

Le résultat de cette requête SPARQL est présenté sur la **Figure 15** suivante :

```
<sparql xmlns='http://www.w3.org/2005/sparql-results#'>
    <head><variable name='Composant'/></head>
    <results>
          <result>
              <binding name='Composant'>
                    <uri>&odt;CableElectrique</uri>
              </binding>
          </result>
          <result>
              <binding name='Composant'>
                    <uri>&odt;Cadre</uri>
              </binding>
          </result>
          <result>
              <binding name='Composant'>
                    <uri>&odt;CellulePhotoV</uri>
              </binding>
          </result>
          ...
          <result>
              <binding name='Composant'>
                    <uri>&odt;Tuile</uri>
              </binding>
          </result>
    </results>
</sparql>
```

Figure 15 Extrait du résultat de requête SPARQL

Ce résultat exprime qu'un verre polymère est composé d'un cadre, d'une cellule photovoltaïque, etc. Si l'un des concepts résultats est défini par d'autres concepts représentant des sous-composants, le système exécutera à nouveau le patron de requête en l'instanciant avec le concept défini. Ce processus se répètera jusqu'à identifier tous les composants et sous-composants.

Le résultat de ces requêtes est une liste générée à la volée de concepts représentant les éléments constituants du produit à décrire et auxquels sont associés des formulaires à renseigner. Les formulaires à renseigner sont ainsi sélectionnés et orchestrés grâce aux requêtes qui s'exécutent sur l'ontologie du dossier technique.

Avec les informations renseignées par l'industriel dans la suite de formulaires qui lui sont présentés par le système, une description RDF du dossier technique est générée, interopérable et réutilisable dans d'autres systèmes. La **Figure 16** suivante présente un exemple d'une telle représentation RDF. Nous verrons dans le chapitre 4 que c'est ce qui nous permet de pouvoir automatiser la vérification de la conformité du dossier technique par rapport aux normes du domaine photovoltaïque. Une autre version du dossier technique est générée au format PDF à destination des instructeurs du CSTB.

```
dt:Module1 odt:aDemandeur dt:BOUZIDI;
      odt:aModule dt:VerrePolymere1.
dt:VerrePolymere1 a dt:VerrePolymere;
      odt:aComposant dt:CellulePhotoV1, dt:FilmPolymere1.
dt:BOUZIDI a odt:Industriel;
      odt:fax "04 92 96 50 63";
      odt:module "Verre Polymer" ;
      vcard:name "Khalil BOUZIDI";
      vcard:ADR "Office 429 Wimmics Team Polytech' Nice Sophia 930 Route
des Colles BP 145 F-06903 Sophia Antipolis Cedex France ";
      vcard:EMAIL "wimmics@lists-sop.inria.fr";
      vcard:TEL "04 92 96 50 63".
dt:FilmPolymere1 a odt:FilmPolymere;
      rdfs:label " Polyéthylène téréphtalate";
      odt:aEpaisseur "0.35";
      odt: tensionElectriqueMax "1255".
dt:CellulePhotoV1 a odt:CellulePhotoV;
      odt:dimensionsHor "150";
      odt:adimensionsVert "150";
      odt:aepaisseur "220";
      odt:anombreCell "36";
      odt: distanceMinimaleEntreCellulesHor "3.5";
      odt:distanceMinimaleEntreCellulesVer "3.5";
      odt:distanceMinimaleAuBordHor "18";
      odt:distanceMinimaleAuBordVer "18".
```

Figure 16 Extrait de la représentation RDF/N3 d'un dossier technique produite par l'outil d'aide à la rédaction de dossier technique

Nous avons implanté un outil d'aide à la rédaction de dossier technique, décrit en détails dans le chapitre 5. Nous avons soumis ce prototype à des tests fonctionnels auprès des instructeurs du service photovoltaïque du CSTB pour valider la pertinence de notre approche ; cela est décrit dans le chapitre 6.

4 Conclusion

Notre première contribution est la facilitation de la rédaction d'un dossier technique afin de permettre à un industriel de disposer plus rapidement d'un avis technique sur un produit. Nous avons adopté une approche ontologique pour la modélisation du domaine concerné. Ces connaissances sont modélisées sous la forme de concepts représentant les différents éléments du dossier technique. Elles sont exploitées à travers des requêtes SPARQL pour déterminer dynamiquement un enchainement de formulaires adaptés au produit que l'industriel décrit. Le résultat de ces requêtes permet d'assurer que l'enchainement des formulaires est cohérent avec les entrées des utilisateurs, qu'aucun composant du produit ne manque et que tous les éléments spécifiques à son produit sont renseignés. Notre système repose sur les langages et techniques du web sémantique et assure ainsi l'interopérabilité des connaissances manipulées et produites.

CHAPITRE 3
MODELISATION DES GUIDES PRATIQUES

1 Introduction 68
2 Modélisation des contraintes réglementaires 71
2.1 Ontologie OntoDT 71
2.2 Modélisation des contraintes réglementaires sous la forme de règles SBVR 74
2.2.1 Reformulation préalable 76
2.2.2 Extraction de règles à partir de tableaux 77
2.3 Modélisation des contraintes réglementaires en RDF et SPARQL 79
2.3.1 Représentation des contraintes réglementaires en RDF et SPARQL 79
2.3.2 Construction de représentations RDF et SPARQL à partir de règles SBVR 80
3 Modélisation du processus de vérification des contraintes réglementaires 83
3.1 Modèle de processus 84
3.2 Processus élémentaire de vérification de contraintes 85
3.3 Processus complexe de vérification de contraintes 86
4 Résultats et limitations 87
5 Conclusion 89

1 Introduction

Les règles et règlements sont écrits par ou pour des humains et ne sont lus et appliqués que par des humains. De ce fait, ils sont parfois incomplets ou contradictoires, leur structure est souvent arbitrairement complexe et leur mise en œuvre requiert une véritable expertise du domaine. Dans ce contexte général, nous avons travaillé à la modélisation des contraintes exprimées dans les textes réglementaires du domaine du bâtiment ainsi qu'à la modélisation des processus de vérification de la conformité aux réglementations, qui repose sur la vérification du respect de ces contraintes. Plus précisément, nous nous sommes intéressés à la modélisation des textes réglementaires utilisés par les experts du CSTB pour émettre des avis techniques sur des dossiers techniques soumis par les industriels, ainsi qu'à la modélisation du processus expert de vérification de la conformité du dossier technique aux contraintes réglementaires qui sous-tend l'émission d'un avis technique. Ce faisant, notre objectif final est de développer un outil d'aide à la rédaction d'avis techniques à destination des experts du CSTB.

Pour ce faire, nous avons exploité les guides pratiques publiés par le CSTB qui capturent sous une forme plus opérationnelle les contraintes réglementaires contenues dans les textes réglementaires (arrêtés, circulaires), normatifs (Documents Techniques Unifiés ou DTU, Normes Françaises ou NF) ou codificatifs (Avis Techniques ou ATec). Les guides pratiques ne remplacent pas ces textes de référence ; ils constituent un complément réglementaire offert par le CSTB aux industriels pour leur permettre une lecture plus facile des règles techniques de construction. Ils visent à simplifier la lecture et l'application des réglementations. Ils recueillent des détails d'exécution présentant un large éventail de situations possibles de mise en œuvre. Les acteurs visés par ces guides sont les maîtres d'ouvrage, maîtres d'œuvre, bureaux d'études des entreprises. L'objectif principal de ces documents est de répondre aux problématiques de sécurité et de santé auxquels sont confrontés les industriels en particulier lors de la phase de conception.

On y trouve de nombreux cas concrets, schémas, tableaux de synthèse, les principales exigences réglementaires qui doivent être respectées (sécurité, hygiène, accessibilité, performance) en fonction des produits et de la nature des travaux. Aussi, les solutions proposées face au risque pouvant menacer la sécurité ou la santé des utilisateurs sont commentées et clairement illustrées par des exemples de réalisation. Les guides pratiques sont tous organisés de la même façon et encouragent ainsi un parcours type de lecture : définition

des risques, diagnostic, solution technique, choix, mise en œuvre, compatibilité avec les règles existantes, contrôles, exemples d'utilisation et liens utiles.

Les guides pratiques constituent pour notre travail autant d'informations extraites des textes de référence, dont la modélisation permettrait d'automatiser une partie de la tâche de vérification de la conformité aux réglementations en vigueur des dossiers techniques soumis par les industriels au CSTB. Les guides pratiques regroupent en effet toutes les caractéristiques structurelles et variables dimensionnelles relatives à la validation d'un produit.

Comme nous le verrons dans la suite de ce chapitre, nous proposons une modélisation des contraintes réglementaires extraites des guides pratiques dans les langages SBVR et SPARQL et une organisation de cette base de contraintes selon des critères de classification issus de la structure commune des guides pratiques :

1. Organisation par source documentaire : l'idée est de classifier les contraintes modélisées selon le type et la référence du document réglementaire dont elles sont issues : normatifs ou codificatifs référencés dans les guides pratiques. Exemple : Guide pratique « Couverture en tuiles » issu des DTU 40.211, 40.211, 40.22, etc.
2. Organisation par domaine d'application : chaque contrainte, ou ensemble de contraintes, est relative à un domaine d'application précis. Exemple : Accessibilité, sécurité, photovoltaïque, etc.
3. Organisation par thématique : cette organisation est la plus importante ; on peut alors classifier les contraintes selon les solutions techniques qui les référencent. Exemple : Tuile, installation électrique, etc.

La **Figure 17** suivante présente les différentes organisations possibles des guides pratiques.

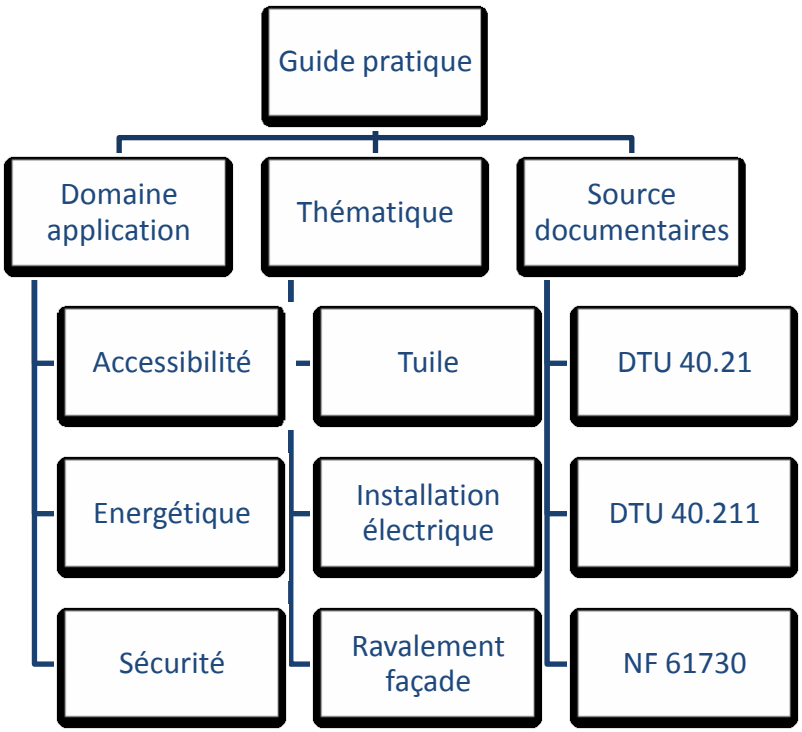


Figure 17 Organisation des guides pratiques

Dans notre travail, nous avons exploité le guide pratique « Les couvertures en tuiles » comme modèle d'étude. Ce guide expose les différents types de tuiles avec leurs caractéristiques. Il définit leurs conditions de mise en œuvre ainsi que divers critères de vérification des tuiles, tels que la pente, le support ou la climatologie de l'endroit où elles doivent être installées. Le suivi de ces consignes se fait de façon drastique car le non-respect d'une exigence conduit à une non-validation du dossier technique.

Nous présentons dans la partie suivante la modélisation des contraintes réglementaires que nous proposons, qui repose sur l'enrichissement de l'ontologie OntoDT présentée dans le chapitre précédent avec de nouvelles définitions de concepts. Les contraintes réglementaires sont représentées en langage naturel contrôlé par des règles SBVR et formellement par des requêtes SPARQL annotées en RDF qui reposent sur le vocabulaire et les définitions dans OntoDT. Nous présenterons ensuite dans la partie 3 le modèle de processus que nous proposons pour capturer le savoir-faire des experts du CSTB en matière de vérification de contraintes réglementaires. Nous définissons récursivement un processus comme la combinaison de processus plus spécifiques, des processus élémentaires étant définis comme la composition de contraintes représentées à la fois en SPARQL et RDF et en SBVR.

2 Modélisation des contraintes réglementaires

Nous représentons les contraintes réglementaires à la fois dans un langage contrôlé, SBVR, compréhensible par les humains, et dans un langage formel, SPARQL, permettant leur utilisation dans des traitements automatiques. Nous présentons dans cette partie notre méthode d'extraction manuelle de règles SBVR à partir des informations textuelles contenues dans les guides pratiques, puis la formalisation de ces contraintes réglementaires en SPARQL et notre méthode de transformation des règles SBVR en requêtes SPARQL (Bouzidi, Faron-Zucker, Fies, Corby, and Le Thanh, 2012a). Les règles SBVR et les requêtes SPARQL reposent sur un vocabulaire commun, l'ontologie OntoDT, qui est également celui sur lequel repose la représentation RDF des dossiers techniques (chapitre 2) dont nous souhaitons automatiser la vérification de la conformité à la réglementation en vigueur (chapitre 4). Nous commençons par présenter comment nous avons enrichi OntoDT pour la représentation des contraintes réglementaires.

2.1 Ontologie OntoDT

Dans le guide pratique « Les couvertures en tuiles », nous avons recensé neuf types de tuiles (**Figure 19**), chacune avec des caractéristiques dimensionnelles ou structurelles différentes les unes des autres. Les types de tuile et leurs caractéristiques ont été représentés respectivement par des classes et propriétés dans OntoDT.

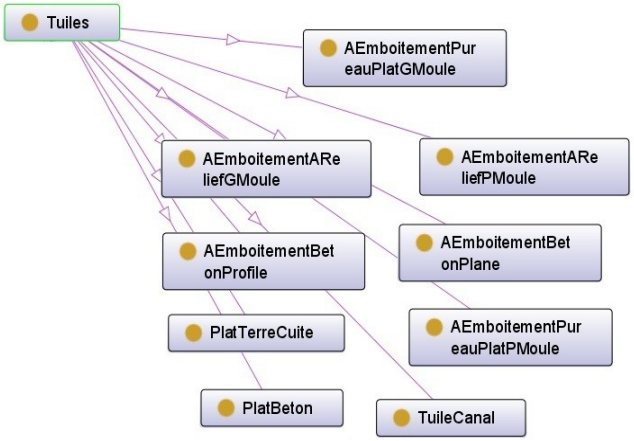


Figure 19 Types de tuiles

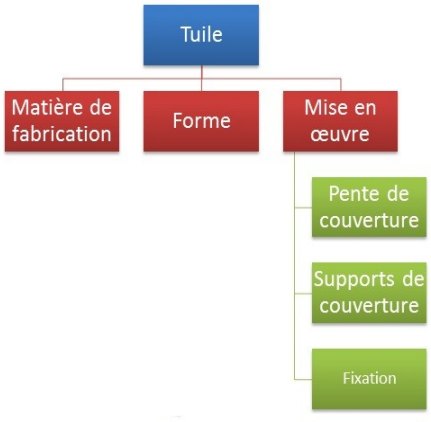


Figure 18 Caractéristiques d'une tuile

Chaque type de tuiles est défini par une matière de fabrication, une forme et une mise en œuvre dont la description comprend l'indication d'une pente, d'un support, et d'un type de fixation (**Figure 18**).

- Pentes : Les inclinaisons minimales admissibles pour les pentes des tuiles sont fonction des zones d'application et de leurs situations d'installation. C'est ce que montre le **Tableau 1**.

Tableau 1. Pentes de tuiles minimales admissibles, issues du DTU 40.21

Type de tuile	Situation	Zones d'application		
		Zone I (m.p.m.)	Zone II (m.p.m.)	Zone III (m.p.m.)
Grand moule	Protégée Normale Exposée	0,30 0,35 0,50	0,30 0,45 0,60	0,45 0,50 0,70
Petit moule	Protégée Normale Exposée	0,35 0,45 0,60	0,45 0,50 0,70	0,50 0,60 0,75

- Support : Les tuiles reposent sur un litonnage en bois dont les éléments sont fixés à raison d'une pointe ou d'une agrafe à chaque intersection d'un chevron et d'un liteau. Le **Tableau 2** présente les dimensions des liteaux en fonction de certaines valeurs d'entraxe et de charge totale.

Tableau 2. Dimensions des liteaux en bois, issues du DTU 40.21

Section (mm) (h × l) *)	Entraxe maximal des appuis en mètres suivant la charge exprimée en daN/m²		
	100	150	200
22 × 25	0,45	0,43	0,40
25 × 25	0,55	0,50	0,45
25 × 32	0,60	0,54	0,50
25 × 38	0,64	0,57	0,52
25 × 50	0,74	0,66	0,62
32 × 32	0,79	0,70	0,62
32 × 38	0,83	0,74	0,68
38 × 38	1,00	0,89	0,82
38 × 50	1,10	0,98	0,90

*) *h* : hauteur du liteau ; *l* : largeur du liteau. L'utilisation de liteaux d'une hauteur inférieure à 22 mm n'est pas admise.

- Fixation : La fixation est destinée à assurer le maintien de l'assemblage des tuiles entre elles lorsque les effets du vent risquent d'en déranger l'ordonnancement. La fixation des tuiles, si elle n'est pas totale, se fait d'une manière répartie, par clouage et/ou crochetage et/ou pannetonnage. La fixation minimale des tuiles, en partie courante, doit être exécutée suivant les cas indiqués dans le **Tableau 3**.

Tableau 3. Fixation des tuiles en partie courante, issues du DTU 40.21

Pentes (m.p.m.)	**Régions 1 et 2 : Sites protégé et normal** 1)	**Régions 1 et 2 : Site exposé** 1) **Région 3 : Tous sites** 2)
$p \leq 1,00$	aucune	1/5 2)
$1,00 < p \leq 1,75$	1/5 2)	1/5 2)
$p > 1,75$	toutes	toutes
1) ***Les régions et sites d'exposition au vent considérés sont ceux prévus par les règles NV 65.*** 2) *Une tuile fixée toutes les cinq tuiles posées.*		

Nous utilisons le langage OWL pour capturer ces caractéristiques structurelles et dimensionnelles dans la définition de chaque classe représentant un type de tuiles. Par exemple, les **Figure 20** et **Figure 21** suivantes présentent la définition de la classe représentant les tuiles plates en terre cuite dans l'éditeur Protege et la définition de classe OWL produite.



Figure 20 De la classe des tuiles plates en terre cuite dans l'éditeur Protegé

```
@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix odt: <http://www.cstb.fr/ontologies/odt#> .

odt:PlatTerreCuite a owl:Class ;
rdfs:label "PlatTerreCuite" ;
rdfs:subClassOf odt:Tuiles,
                [ a owl:Class ;
                owl:intersectionOf (
                   [ a owl:Restriction ;
                        owl:onProperty odt:aUneForme ;
                        owl:someValuesFrom odt:Plat ]
                   [ a owl:Restriction ;
                        owl:onProperty odt:aUneMatiere ;
                        owl:someValuesFrom odt:TerreCuite ]
                   [ a owl:Restriction ;
                        owl:onProperty odt:aUneMiseEnOeuvre ;
                        owl:someValuesFrom odt:MiseEnOeuvre ] ) ] .
```

Figure 21 Définition OWL de la classe des TuilePlateTerreCuite

OntoDT permet ainsi de représenter de façon déclarative des connaissances extraites des guides pratiques, sans fixer de sémantique opérationnelle, i.e. sans présumer de la façon dont les définitions de concepts seront utilisées.

2.2 Modélisation des contraintes réglementaires sous la forme de règles SBVR

Il existe des travaux visant à reconnaître automatiquement les phrases décrivant des règles métier (Martínez-Fernández et al., 2008). Les auteurs reposent sur une analyse linguistique des textes pour détecter les phrases contenant une structure conditionnelle. Une phrase contenant une telle structure conditionnelle est marquée comme candidate pour exprimer une règle métier.

Par ailleurs, les normes techniques peuvent être interprétées de différentes manières. C'est pourquoi l'intervention d'un expert du domaine est primordiale pour identifier les blocs textuels contenant, potentiellement, des contraintes réglementaires. Ces deux approches sont complémentaires, un expert du domaine pourrait être assisté par un système de reconnaissance automatique basé sur des techniques de TAL pour identifier les phrases exprimant des règles métiers. Cependant, nous n'avons pas abordé dans le cadre de cette thèse les approches à base de TAL ; elles sont complémentaires de nos travaux.

Avec l'aide d'experts du CSTB, nous avons identifié, manuellement, à partir du guide pratique « Les couvertures en tuiles », des contraintes réglementaires et nous avons ensuite cherché à les représenter. Soulignons que dans ce cas particulier des guides pratiques du CSTB, de nombreuses réglementations sont synthétisées sous la forme de tableaux et un travail préliminaire a consisté non pas à extraire à partir de textes des contraintes réglementaires mais plutôt à formuler en langage naturel contrôlé des contraintes réglementaires à partir de l'interprétation de tableaux (Bouzidi, Faron-Zucker, Fies, and Le-Thanh, 2011).

SBVR est un standard de l'OMG qui vise à représenter de manière déclarative des règles de conformité complexes telles que les règles de conformité à la réglementation ; il permet de construire une représentation logique d'une règle métier avec une interface en langage naturel contrôlé (Bajwa et al., 2011). Dans notre cas particulier de la représentation des contraintes réglementaires extraites du guide pratique des couvertures en tuiles, nous reposons sur un vocabulaire contrôlé consistant en les labels des classes et propriétés de l'ontologie OntoDT, auxquels s'ajoutent des mots-clés réservés SBVR.

Considérons l'exemple de règle SBVR opérative suivante :

> **Si un module photovoltaïque *est* mis en œuvre en Zone 1 alors la vitesse du vent *est* inférieure à 103 km/h.**

Les termes faisant référence à des classes ou propriétés dans OntoDT sont reliés à ces dernières à travers leur mise en forme : bleu et souligné ; les valeurs sont en rouge et les mots-clés SBVR sont en jaune. Le tableau suivant précise la nature des équivalences ontologiques.

Représentation en SBVR	Equivalent Ontologique
Si	Condition
un module photovoltaïque	Concept (Module)
est	Quantification existentielle
mis en œuvre	Propriété (aMiseEnOeuvre)
en	Symbole d'information
Zone 1	Concept (Zone1)
alors	Conséquent
La vitesse du vent	Propriété (aVitesse)
***est* inférieure à**	Opérateur (<)
103 km/h	Valeur

La construction de règles SBVR à partir du guide pratique des couvertures en tuiles s'est avérée être un travail délicat, nécessitant des reformulations préalables ou des formulations à partir de tableaux. C'est ce que nous détaillons dans la suite de cette partie.

2.2.1 Reformulation préalable

La représentation sous forme de règles SBVR d'un fragment de texte identifié comme exprimant une contrainte réglementaire ne consiste pas uniquement à mettre en forme les différents éléments linguistiques d'une phrase pour faire apparaître sa structure logique et ses correspondants dans le vocabulaire contrôlé. Une étape de désambiguïsation et de reformulation est le plus souvent nécessaire. Considérons par exemple l'extrait de la norme EN 61730 suivante auquel le guide pratique des couvertures en tuiles fait référence :

> *Châssis principal :*
>
> *Les dimensions du châssis principal doivent être :*
> *Largeur intérieur : (847 ± 5) mm*
> *Hauteur intérieur :(1910 ± 5) mm*

Cette formulation cache des conditions qui sont difficiles à interpréter par des utilisateurs qui ne sont pas experts du domaine. La désambiguïsation de ce texte conduit à la reformulation suivante :

La largeur maximale d'un châssis principal doit être inférieure ou égale à 853 mm et sa largeur minimale doit être supérieure ou égale à 842 mm. La hauteur maximale d'un châssis principal doit être inférieure ou égale à 1915 mm et sa hauteur minimale doit être supérieure ou égale à 1905 mm.

Les références à l'ontologie sont ensuite identifiées ; dans l'exemple il s'agit des concepts **châssis** et **châssis principal**. La structure logique de la phrase est mise en évidence par la coloration de ses éléments pour aboutir à la règle SBVR suivante :

Si le châssis *est* un châssis principal, alors la largeur minimale doit être supérieure ou égale à 842 mm et la hauteur minimale doit être supérieure ou égale à 1905 mm et la largeur maximale doit être inferieure ou égale à 853mm et la hauteur maximale doit être inferieure ou égale à 1915 mm.

Considérons un second exemple extrait de la norme EN 61730 :

Classe A: Accès général, tension dangereuse, applications de puissance dangereuses

Les modules assignés pour utilisation dans cette classe d'application peuvent être utilisés dans des systèmes fonctionnant à plus de 120 V en courant continu. Les modules requis pour la sécurité dans la présente partie de l'EN 61730 et l'EN 61730-2 dans cette classe d'application sont considérés comme satisfaisant aux exigences de la classe II de sécurité.

Les composants ontologiques identifiés dans ce fragment de la norme sont **Module**, **Classe de sécurité**, **Classe d'application**, **Classe de sécurité**.

Ce texte réglementaire peut être représenté par la règle SBVR suivante :

Si la classe d'application *est* A alors le module doit avoir un système qui *fonctionne* à plus de 120 volts et une classe de sécurité 2.

Ces exemples montrent bien les possibilités limitées d'automatisation de la formalisation en SBVR des textes auxquels nous sommes confrontés et la nécessité d'une connaissance experte du domaine.

2.2.2 Extraction de règles à partir de tableaux

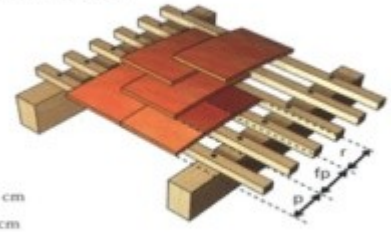
Comme nous l'avons déjà mentionné précédemment, les guides pratiques du CSTB contiennent des représentations de contraintes réglementaires issues notamment des DTU

sous la forme de tableaux. Par exemple, le tableau suivant est extrait du guide des « Couvertures en tuiles » :

Pentes des couvertures en tuiles plates de terre cuite

Pentes et recouvrements des tuiles plates

Pose sans écran
DTU 40.23, tableau 1



Emploi admis et recouvrement « r » ≥ 7 cm
Emploi admis et recouvrement « r » ≥ 8 cm
Emploi admis et recouvrement « r » ≥ 9 cm
Emploi non admis

r : recouvrement
fp : faux pureau
p : pureau

Pentes valables pour les rampants de longueur maximale de 8 m (en projection horizontale)

Pente de couverture	Zone I (1)			Zone II (1)			Zone III (1)		
	Situation (1)			Situation (1)			Situation (1)		
	Protégée	Normale	Exposée	Protégée	Normale	Exposée	Protégée	Normale	Exposée
70 %	8 cm			8 cm					
75 %	8 cm			8 cm					
80 %	7 cm	8 cm		7 cm			9 cm		
85 %	7 cm	8 cm		7 cm			9 cm		
90 %	7 cm	7 cm		7 cm	8 cm		8 cm		
95 %	7 cm	7 cm		7 cm	8 cm		8 cm		
100 %	7 cm	7 cm	8 cm	7 cm	7 cm		8 cm	9 cm	
105 %	7 cm	7 cm	8 cm	7 cm	7 cm		8 cm	9 cm	
110 %	7 cm	7 cm	7 cm	7 cm	7 cm	8 cm	8 cm	8 cm	
115 %	7 cm	7 cm	7 cm	7 cm	7 cm	8 cm	8 cm	8 cm	9 cm
120 %	7 cm	7 cm	7 cm	7 cm	7 cm	7 cm	8 cm	8 cm	9 cm
≥ 125 %	7 cm	7 cm	7 cm	7 cm	7 cm	7 cm	8 cm	8 cm	8 cm

1. Les zones et situations de concomitance vent/pluie sont définies dans le chapitre « Climatologie », page 102.

Figure 22 Extrait du guide pratique des couvertures en tuiles

La compréhension de ce tableau fait appel à la connaissance de conditions implicites qui nécessitent une connaissance approfondie du domaine. Le travail de reformulation effectué sur ce genre de données comporte deux étapes :

- La première étape consiste à transformer les lignes du tableau en phrases afin d'identifier toutes les conditions d'application. Par exemple, dans le tableau (**Figure 22**), la première colonne représente les différentes inclinaisons possibles pour une tuile plate en terre cuite. En effet, l'inclinaison des pentes change en fonction de leurs recouvrements et de leurs zones d'implantation. Nous avons ce faisant identifié les principales références aux concepts d'OntoDT, tels que **pente de couverture**, **recouvrement**, **zone** d'application ;

- La seconde étape consiste à réécrire les phrases écrites lors de la première étape en réutilisant les classes et propriétés d'OntoDT. Ceci nous a permis d'arriver à une reformulation de la contrainte réglementaire exprimée dans la première ligne du tableau sous la forme de la règle SBVR suivante :

  Si la pente de couverture est égale à 70 %, alors la mise en œuvre doit être en Zone 1, en situation Protégée, avec un recouvrement supérieur à 8 cm.

Cet exemple montre bien les possibilités limitées d'automatisation de la formalisation en SBVR des contraintes réglementaires présentées dans de tels tableaux et la nécessité d'une connaissance experte du domaine pour interpréter ces tableaux.

2.3 Modélisation des contraintes réglementaires en RDF et SPARQL

SBVR décrit les concepts et les exigences sans se préoccuper de leur implémentation. Pour cela, nous avons traduit les règles SBVR en annotations RDF liées à des requêtes SPARQL identifiées prédéfinies. C'est cette formalisation, qui permettra d'automatiser le processus de vérification de la conformité d'un dossier technique aux textes réglementaires.

2.3.1 Représentation des contraintes réglementaires en RDF et SPARQL

Les règles SBVR que nous avons écrites pour formaliser les contraintes réglementaires extraites des guides pratiques sont toutes des règles opératives de la forme « Si <antécédent> alors <conséquent> », indiquant une obligation : elles stipulent que si l'antécédent est satisfait par les données à vérifier, alors le conséquent de la règle doit l'être également. Il ne s'agit donc pas ici de règles d'inférence mais de l'association de deux énoncés à vérifier, la contrainte étant que si le premier énoncé est vérifié, le second doit l'être également. En d'autres termes, la sémantique opérationnelle de ces règles est de vérifier le respect d'une contrainte pour une condition donnée vérifiée.

Nous représentons ces règles en RDF, à l'aide d'un petit vocabulaire dédié que nous présenterons en détail dans la section 3. Chaque règle est représentée par une ressource

anonyme de type `Test` et est le sujet de deux propriétés `if` et `then` qui lui associent ses antécédent et conséquent à vérifier. C'est ce que montre la **Figure 23** :

```
@prefix kg: <http://ns.inria.fr/edelweiss/2010/kgram/> .
[ a kg:Test ;
  kg:if <Antécédent> ;
  kg:then <Conséquent> ] ) .
```

Figure 23 Représentation RDF d'une règle opérative

Quant aux contraintes proprement dites, les antécédents et conséquents, nous les représentons par des requêtes SPARQL de la forme ASK. Une requête SPARQL ASK permet en effet de déterminer si un graphe requête (une contrainte) a une correspondance dans un graphe RDF (une description de dossier technique) – la réponse à une telle requête est une valeur booléenne. Sur la **Figure 23** ci-dessus, `Antécédent` et `Conséquent` désignent donc les URI de deux requêtes SPARQL.

2.3.2 Construction de représentations RDF et SPARQL à partir de règles SBVR

Nous avons défini quatre étapes pour transformer une règle SBVR en requêtes SPARQL et annotation RDF. Cette transformation repose bien sûr sur l'utilisation d'un vocabulaire commun, l'ontologie OntoDT, dans les règles SBVR et les requêtes SPARQL. Considérons par exemple la règle SBVR suivante :

> **Si la pente de couverture *est* égale à 70 % alors la mise en œuvre doit être en Zone 1 en situation Protégée avec un recouvrement supérieur à 8 cm.**

Sa formalisation doit permettre de contrôler la pente de recouvrement adaptable pour une tuile plate en terre cuite, installée en zone géographique 1 en situation protégée. La **Figure 24** suivante montre la formalisation de cette règle dans le langage SPARQL.

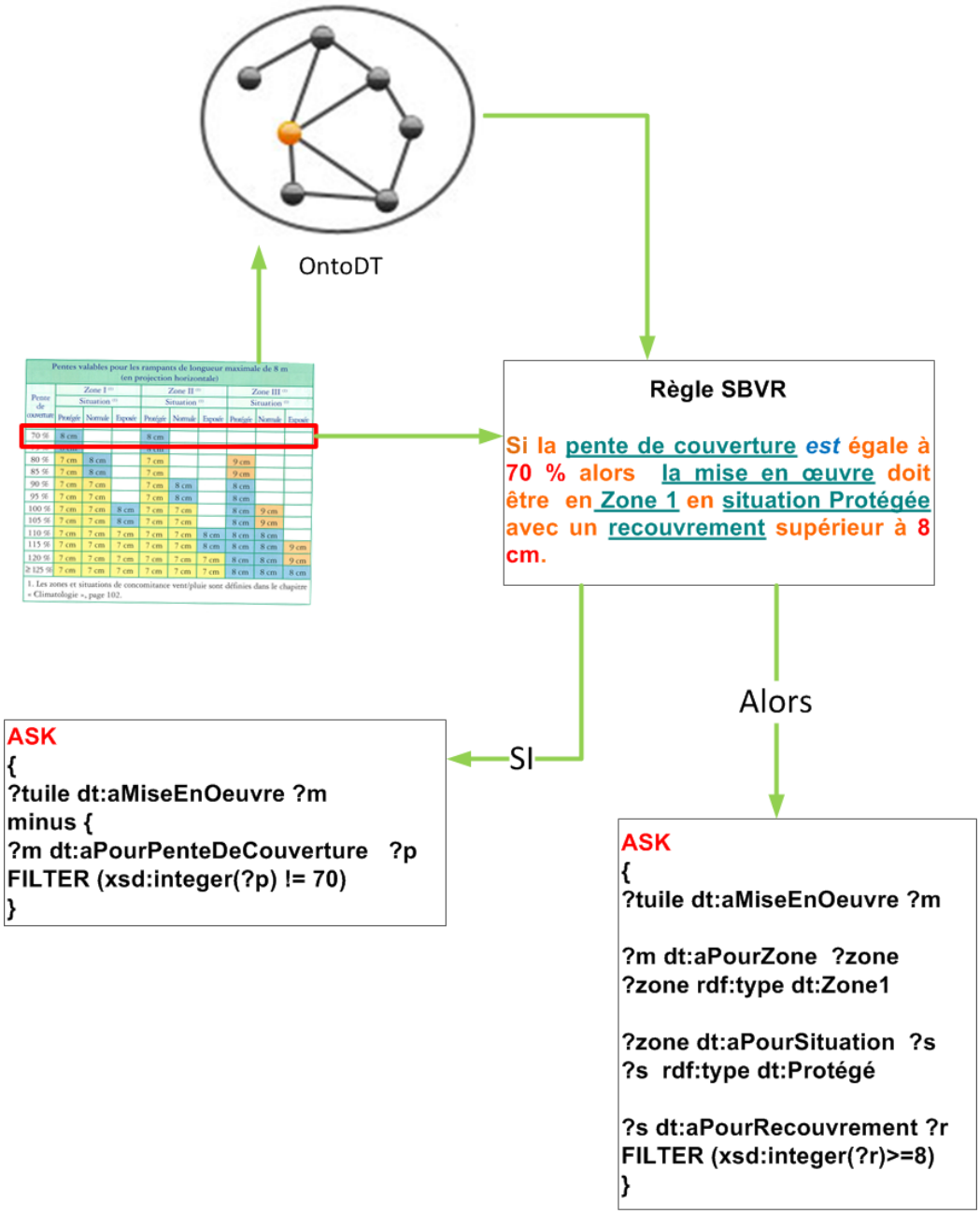


Figure 24 Transformation des règles SBVR en requêtes SPARQL

Etapes 1 : Nous reformulons la règle SBVR en la décomposant en énoncés SBVR qui pourront être facilement traduits dans le modèle de triplets de SPARQL et nous distinguons antécédent et conséquent. Reprenons l'exemple de règle SBVR de la **Figure 24** ; elle peut être décomposée comme suit :

- Antécédent :
  - La pente de couverture est égale à 70 %
- Conséquent :
  - (1) La tuile *a* une mise en œuvre
  - (2) La mise en œuvre *a* une zone
  - (3) La zone *est* de type Zone 1
  - (4) La mise en œuvre *a* une situation
  - (5) La situation *est* de type Protégée
  - (6) La situation *a* un recouvrement supérieur à 8 cm

Au regard de la définition de la classe des tuiles plates en terre cuite dans l'ontologie OntoDT (**Figure 21**), nous avons explicité que la tuile est le sujet de la propriété d'avoir une mise en œuvre, que la mise en œuvre est le sujet des propriétés d'avoir une zone et une situation et que le sujet d'avoir un recouvrement est la situation.

Etape 2 :

Nous procédons ensuite à la transformation en triplets SPARQL de chacun des énoncés SBVR élémentaires dans l'antécédent et le conséquent de la règle. Nous utilisons pour cela les classes et propriétés de l'ontologie OntoDT. La transformation de notre exemple est présentée dans le **Tableau 4**.

Tableau 4 : Transformation des règles SBVR en pattern de graphes SPARQL

Enoncé SBVR	Enoncé SPARQL
La pente de couverture est égale à 70 %	`minus { ?m dt:aPourPenteDeCouverture ?p FILTER (xsd:integer(?p) != 70)`
La tuile *a* une mise en œuvre	`?tuile dt:aMiseEnOeuvre ?m`
La mise en œuvre *a* une zone	`?m dt:aPourZone ?zone`
La zone *est* de type Zone 1	`?zone rdf:type dt:Zone1`
La mise en œuvre *a* une situation	`?m dt:aPourSituation ?s`
La situation *est* de type Protégée	`?s rdf:type dt:Protégé`
La situation *a* un recouvrement supérieur à 8 cm	`?s dt:aPourRecouvrement ?r FILTER (xsd:integer(?r)>=8)`

Etapes 3 :

Nous construisons deux requêtes SPARQL pour représenter l'antécédent et le conséquent de la règle SBVR, en effectuant la conjonction des énoncés SPARQL produits à l'étape précédente. C'est ainsi que la règle SBVR que nous avons prise pour exemple est traduit dans les deux requêtes SPARQL que l'on peut observer sur la **Figure 24**.

Etapes 4 :

Finalement, nous construisons une représentation RDF de la règle, selon le modèle présenté sur la **Figure 23**, où les valeurs des propriétés `if` et `then` sont les URI des deux requêtes SPARQL construites :

```
@prefix kg: <http://ns.inria.fr/edelweiss/2010/kgram/> .
@prefix dt: <http://www.cstb.fr/ontologies/data#>.

[ a kg:Test ;
      kg:if dt:P70 ;
      kg:then dt:Check70 ] .
```

Ces règles opératives ainsi formalisées sont les briques de la formalisation du processus de vérification des contraintes réglementaires que nous décrivons dans la partie suivante.

3 Modélisation du processus de vérification des contraintes réglementaires

Dans le but de développer un outil d'assistance pour les experts du CSTB en charge de contrôler des dossiers techniques et d'émettre des avis techniques sur ces dossiers, nous avons cherché à modéliser non seulement les règles opératives extraites des réglementations mais aussi le savoir-faire de ces experts dans l'utilisation particulière qu'ils ont de ces réglementations (Bouzidi, Faron-Zucker, Fies, Corby, and Le Thanh, 2012b).

Avec l'aide d'experts du CSTB, nous avons pu définir un ordre de traitement des réglementations, en fonction des priorités existantes entre elles : (I) les réglementations d'un même domaine, (II) les contraintes d'une même réglementation, et (III) les connaissances expertes spécifiant les règles d'une même contrainte :

- Au sein du même domaine : les réglementations qui décrivent un équipement constitué d'une multitude de composants sont prioritaires par rapport à celles qui décrivent des composants uniques. Par exemple, le guide pratique des couvertures en tuiles est prioritaire à la norme NF12600 qui décrit une méthode d'essai d'impact et de classification du verre plat.
- Au sein de la même réglementation : les règles qui permettent de vérifier un équipement sont prioritaires par rapport aux règles qui permettent de vérifier les éléments qui font partie de sa nomenclature. Par exemple, dans le guide pratique des couvertures en tuiles, les règles permettant de vérifier les contraintes matérielles d'une tuile sont prioritaires par rapport à celles qui permettent de vérifier les contraintes relatives au support de la tuile.
- Au sein de la même contrainte : une contrainte peut contenir plusieurs règles. Les règles prioritaires sont celles qui sont les plus spécifiques, c'est-à-dire qui décrivent des exigences d'application plus strictes. Un composant n'est considéré comme non conforme qu'une fois que la vérification de toutes les règles d'une contrainte donnée a échoué. Cependant, nous avons identifié, avec l'aide d'experts CSTB, les parties à vérifier en priorité en nous basant sur leurs expériences personnelles et savoir-faire. Ceci nous a permis d'établir un ordre de priorité qui n'est pas toujours exhaustif, mais qui reflète la façon dont ils procèdent pour vérifier la conformité d'un composant.

3.1 Modèle de processus

La représentation déclarative en RDF que nous proposons des processus de vérification de contraintes réglementaires repose sur un petit vocabulaire léger général que nous avons défini pour décrire une séquence d'exécution de requêtes ou de règles (dans notre cas des requêtes représentant des contraintes), possiblement conditionnelles.

Le schéma RDFS de ce modèle est composé de 4 propriétés : `body`, `if`, `then` et `else`, et de 5 classes : `Load`, `Query`, `Pipeline`, `Pipe` et `Test`. La classe principale `Pipeline` désigne un processus et la classe `Pipe` l'appel à un processus. La classe `Query` désigne une requête. L'exécution de requêtes ou de règles peut être conditionnelle : cela est représenté par la classe `Test`, et les propriétés `if` et `then` dont les valeurs sont des processus à exécuter. La description d'un processus peut donc faire récursivement intervenir d'autres processus (`Pipe`),

y compris le processus lui-même. Cette notion de récursivité est utilisée dans la modélisation des processus complexes où chaque processus fait appel à un ou plusieurs processus élémentaires.

La syntaxe abstraite d'un processus est définie par la grammaire suivante :

```
PIPELINE ::= EXP +
EXP ::= Load(Name) | Query(Name) | Test(Query(Name), Exp, Exp)
| Pipe(Name)
```

3.2 Processus élémentaire de vérification de contraintes

Nous avons organisé les requêtes SPARQL représentant des contraintes réglementaires en différents processus élémentaires de vérification de la conformité. Pour chaque composant élémentaire dans OntoDt, nous représentons en RDF un processus dit *élémentaire* de vérification de contrainte, contenant d'une part une suite de couples de requêtes SPARQL à exécuter pour vérifier la conformité de ses caractéristiques (tels que décrits dans la partie 2.3 de ce chapitre), ainsi que la règle SBVR correspondante (telle que décrite dans la partie 2.2). Par exemple, la représentation du processus élémentaire de contrôle de conformité des pentes de recouvrement admissibles une tuile contient 13 règles SBVR et 13 couples de requêtes SPARQL de vérification de la conformité de la pente (**Figure 25**).

```
@prefix kg: <http://ns.inria.fr/edelweiss/2010/kgram/> .
@prefix odt: <http://www.cstb.fr/ontologies/odt#>.
@prefix dt: <http://www.cstb.fr/ontologies/data#>.

dt:processus1 a kg:Pipeline ;
      odt:composant odt:Pente ;
      kg:body (dt:rule1 ; dt:rule2 ; ... dt:rule13).
dt:rule1 a kg:Test ;
      kg:if dt:P70 ;
      kg:then dt:Check70;
      odt:hasSBVRrule dt:sbvr1 ;
dt:sbvr1 odt:hasValue "Si la mise en oeuvre est dans une Zone 1 en
       situation Protégée et un recouvrement supérieur ou égale 8 cm
       ou dans une Zone 2 avec une situation Protégée et un
       recouvrement supérieur ou égale à 8 cm alors la pente de
       couverture est égale à 70 % (Extrait du guide pratique "des
       couvertures en tuiles" en application des DTU 40.2, 40.211,
       40.22, 40,23 et DTU 40.24, 40.241, 40.25) " .
dt:rule2 a kg:Test ;
...
```

Figure 25 Exemple d'association d'une règle SBVR et des requêtes SPARQL qui l'opérationnalisent

3.3 Processus complexe de vérification de contraintes

Un processus complexe de vérification d'un dossier technique est défini récursivement à partir des processus élémentaires associés aux composants dans OntoDT, qui entrent dans sa définition (**Figure 26**).

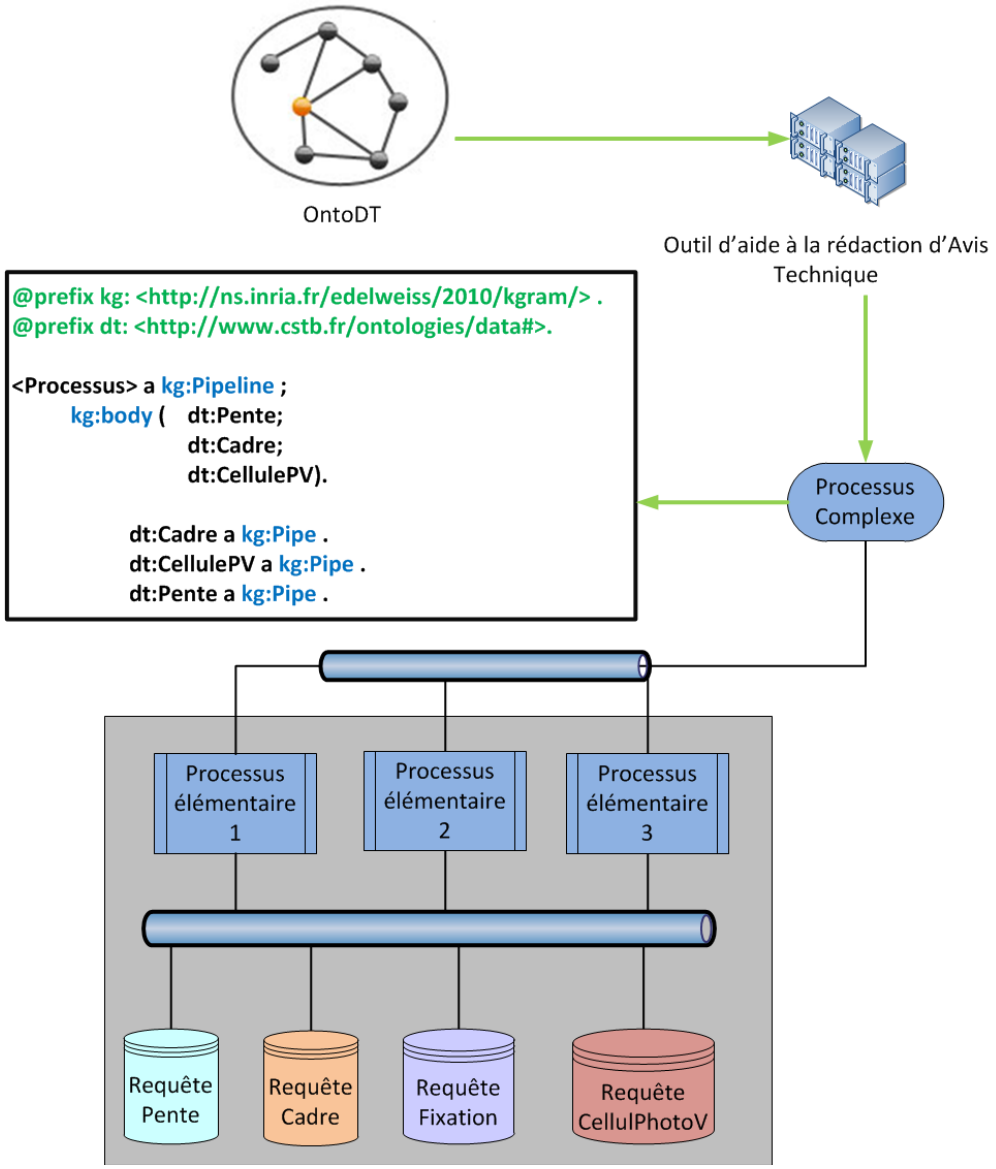


Figure 26 Processus complexe de vérification de contraintes réglementaires

Par exemple, le processus complexe de vérification de la conformité d'un module `VerrePolymere` est composé des processus élémentaires de vérification de la conformité de ses composants `Cadre`, `CellulePhotoV`, `FilmPolymere`, etc. La représentation RDF de cet énoncé est donnée sur la **Figure 27**.

```
@prefix kg: <http://ns.inria.fr/edelweiss/2010/kgram/> .
@prefix dt: <http://www.cstb.fr/ontologies/data#>.

<processus> a kg:Pipeline ;
      kg:body (   dt:Pente;
                  dt:Cadre;
                  dt:CellulePV).
      dt:Cadre a kg:Pipe .
      dt:CellulePV a kg:Pipe .
      dt:Pente a kg:Pipe .
```

Figure 27 Processus complexe de vérification de la conformité d'un verre polymère

La représentation RDF du processus complexe de vérification d'un dossier technique est construite automatiquement en même temps que la représentation RDF du dossier technique (Chapitre 2). En effet, à partir du choix d'un composant par l'industriel, l'ontologie est interrogée pour demander la liste des éléments qui entrent dans la définition de ce composant. En fonction des composants identifiés dans les définitions de classes de l'ontologie, l'outil d'aide à la rédaction du dossier technique génère automatiquement une représentation formelle en RDF du processus complexe de vérification de la conformité du dossier technique en cours de construction.

4 Résultats et limitations

Nous avons travaillé à la modélisation d'un ensemble de règlementations sur les couvertures en tuiles, issues d'un guide pratique de 104 pages qui synthétise les principales exigences de 7 DTU actuellement appliquées en France.

Même si l'interprétation d'experts peut aider à saisir la signification de la plupart des contraintes, certaines d'entre elles peuvent être écrites en SBVR, mais restent non transformables en SPARQL. Ce sont les contraintes qui contiennent des informations difficiles à comprendre et à désambiguïser (par exemple, « *sur une courte distance* »), ou des connaissances communes (Yurchyshyna, 2009) (par exemple, « *Le recouvrement des tuiles faîtières se fait dans le sens opposé aux vents de pluie dominants* », dans « Faîtages scellés » GP Couvertures en tuiles, page 75.

Cependant, les guides pratiques constituent un complément réglementaire offert aux professionnels du domaine du bâtiment pour faciliter la lecture des règles techniques. Ces règles, issues des textes de référence (DTU, NF, etc.), ont été auparavant désambiguïsées et réécrites par les experts des domaines, de manière plus facile à comprendre et plus simple à interpréter. Ainsi, les règles contenant des informations difficiles à comprendre ou qui contiennent des connaissances communes se sont avérées très rares.

Par contre, la limitation actuelle des guides pratiques est qu'ils recueillent des détails d'exécution présentant un large éventail de situations possibles de mise en œuvre. Ce type de règles, facilement représentable en SBVR, constitue la plus grande partie des règles que nous n'avons pas pu traduire en SPARQL. Or ces règles représentent la plus grande partie des contraintes réglementaires identifiées dans les guides pratiques. Nous présentons quelques exemples des différents types de formulations de mise en œuvre non transformable en SPARQL dans le tableau suivant :

Texte réglementaire	Equivalent en SBVR
« Que les arêtiers soient scellés ou à sec, les tuiles sont tranchées biaises par rapport à la ligne d'arêtier » *Arêtiers p76*	**Si les arêtiers *sont* scellés ou à sec alors les tuiles *sont* tranchées biaises à la ligne d'arêtier.**
« Dans le cas des tuiles canal, on exécute généralement un solin en mortier ou un bardelis encastré et scellé dans le mur » *Pénétration continue p. 87*	**Si tuile canal alors on *exécute* un solin en mortier ou un bardelis encastré et *scellé* dans le mur**
« Les bois support de couverture sont fixés à la charpente, généralement sur les chevrons à l'aide de pointes en acier lisses torsadées ou crantées » *Fixation et supports p. 41*	**Si le bois support de couverture *sont fixé* à la charpente avec des pointes en acier lise ou ils *sont fixé* en crantées alors le support de couverture *est* de type bois**
« Les liteaux doivent reposer sur trois appuis au moins en partie courante et doivent être fixés à raison d'une pointe d'agrafe » *Pose et dimensionnement p. 42*	**Les liteaux doivent reposer sur trois appuis au moins en partie courante et doivent être fixée à raison d'une pointe d'agrafe**
« Les abouts de faîtage doivent être obturés » *Faitage scellés p. 75*	**Les abouts de faitage doivent être obturés**

Finalement, avec l'aide d'experts du CSTB, nous avons modélisé en SBVR 100% des exigences réglementaires issues du guide pratique « couvertures en tuiles » et interprétables en SPARQL. Nous avons ainsi construit une base de 164 règles SBVR et de 177 requêtes SPARQL. Nous estimons que toute la réglementation du guide pourrait être modélisable en SBVR mais seulement 30% formalisable avec requêtes SPARQL (Bouzidi, Faron-Zucker, Fies, Zarli, et al., 2012).

5 Conclusion

Notre première contribution consiste en l'utilisation combinée des standards SBVR, RDF et SPARQL pour reformuler d'une part dans un langage contrôlé et d'autre part dans un langage formel les contraintes réglementaires présentes dans les guides pratiques du CSTB. SBVR représente une assurance de la qualité du texte des contraintes réglementaires présentées à l'utilisateur et SPARQL permet l'automatisation de la vérification de ces contraintes. Ces représentations reposent sur l'ontologie de domaine OntoDT que nous avons développée.

En second lieu, nous avons organisé en différents processus élémentaires les requêtes SPARQL représentant des contraintes réglementaires. À chaque composant élémentaire dans OntoDT correspond un processus élémentaire de vérification de sa conformité. Les processus complexes de vérification d'un dossier technique sont définis récursivement et construits automatiquement.

CHAPITRE 4
MODELE D'AIDE A LA REDACTION D'AVIS TECHNIQUES

1 Introduction 92
2 Automatisation de la vérification de la conformité des dossiers techniques 93
2.1 Orchestration des vérifications de contraintes par un moteur de processus 93
2.2 Vérification des contraintes par interrogation de la représentation du Dossier Technique ... 95
3 Aide à la prise de décision 95
3.1 Explication du résultat des vérifications automatiques 96
3.2 Aide à la vérification manuelle de contraintes 98
4 Conclusion 100

1 Introduction

Un de nos objectifs final est, nous l'avons dit, de développer un système d'aide à la rédaction d'avis techniques. Pour cela nous avons établi un modèle de vérification de la conformité d'un dossier technique sur lequel un avis technique est demandé. C'est ce modèle que nous décrivons dans ce chapitre. Il repose sur le mécanisme de raisonnement de base consistant en l'appariement de la représentation formelle du dossier technique en RDF et de la représentation en SPARQL des textes réglementaires en vigueur auxquels le Dossier Technique doit se conformer ; en d'autres termes, il s'agit de l'interrogation de la représentation RDF du dossier technique grâce aux requêtes SPARQL représentant les contraintes réglementaires. L'orchestration de ces vérifications repose quant à elle sur l'interprétation de la représentation RDF du processus de vérification du dossier technique, grâce à un moteur de processus que nous avons développé et dont le principe est l'interrogation de la représentation RDF d'un processus grâce à des requêtes SPARQL prédéfinies. Ainsi, l'ensemble de notre modèle de vérification de la conformité d'un dossier technique repose sur les modèles du Web sémantique.

Dans la perspective d'un outil non pas de vérification automatique mais bien d'aide à la vérification, où l'expert en charge de l'avis technique reste bien au centre de cette tâche, nous nous sommes concentrés sur l'explication des résultats du processus automatique de vérification qui peut être réalisé. Pour cela, notre modèle intègre la notion de rapport de conformité et l'identification des caractéristiques non conformes est automatisée.

Enfin, nous l'avons vu, une partie seulement des contraintes réglementaires peut être modélisée par des requêtes SPARQL (de l'ordre de 30% des contraintes relatives aux couvertures en tuiles) et dans le cas où une contrainte ne peut être vérifiée, notre modèle prévoit de produire la représentation SBVR des contraintes qui restent à vérifier manuellement par l'expert.

Nous commençons par décrire dans la partie suivante notre modèle de vérification automatique de la conformité d'un dossier technique qui repose sur l'interprétation de la représentation RDF du processus de vérification du dossier par un moteur sémantique de processus. Puis nous présenterons dans la partie 3 la partie de notre modèle relative à l'explication des résultats des vérifications automatiques et à la présentation des contraintes qui doivent être vérifiées manuellement.

2 Automatisation de la vérification de la conformité des dossiers techniques

Une partie du processus de vérification de la conformité d'un dossier technique par rapport aux réglementations qui s'appliquent peut être automatisée en exploitant d'une part la représentation formelle du dossier technique, telle que la représentation RDF que nous proposons dans le chapitre 2, produite par l'outil d'aide à la rédaction d'un dossier technique, et d'autre part la représentation formelle des contraintes réglementaires et des processus de vérification de ces contraintes, telle que la représentation RDF des processus et la représentation en SPARQL des contraintes que nous proposons dans le chapitre 3.

2.1 Orchestration des vérifications de contraintes par un moteur de processus

Nous l'avons vu dans le chapitre précédent, une représentation RDF du processus de vérification d'un dossier technique est construite automatiquement en même temps que la représentation RDF du dossier technique, en interrogeant l'ontologie OntoDT et la base des représentations RDF des processus élémentaires associés à chaque composant représenté par une classe dans OntoDT : le processus dit complexe de vérification de la conformité d'un dossier technique est composé des processus élémentaires nécessaires à la vérification de la conformité de chaque composant élémentaire qui participe à la représentation du produit décrit dans le dossier.

Nous avons alors développé un moteur de processus qui interprète la représentation RDF d'un processus pour construire dynamiquement et exécuter une séquence de requêtes SPARQL et permettre ainsi de superviser, coordonner et exécuter séquentiellement un ensemble de requêtes. Ce moteur repose sur la sémantique opérationnelle des classes du vocabulaire des processus décrit dans le chapitre 3. Cette sémantique est résumée comme suit :

- Le moteur de processus interprète la classe `Pipe` en invoquant un sous-processus particulier ;

- Il interprète la classe `Load` en chargeant la ou les ressources contenant les données RDF à traiter et d'éventuelles ontologies – dans notre cas, la représentation du dossier technique à contrôler et l'ontologie OntoDT.
- Il interprète la classe `Query` en chargeant et exécutant une requête SPARQL.
- Il interprète la classe `Test` en exécutant une ou plusieurs requêtes selon qu'une condition spécifique est vérifiée ou non.

Le moteur de processus repose sur un ensemble de patrons de requêtes SPARQL prédéfinis, tels que celui présenté sur la **Figure 28** ci-dessous qui permet de lister tous les composants d'un processus et leurs types, qu'ils soient la valeur de la propriété `body` ou bien appartenant à la liste valeur de la propriété `body`.

```
SELECT ?q ?t WHERE {
  ?p rdf:type kg:Pipeline
  {?p kg:body ?q
   ?q rdf:type ?t
   minus {?q rdf:type rdf:List}
  }
  UNION
  {?p kg:body ?a
   ?a rdf:rest*/rdf:first ?q
   ?q rdf:type ?t
  }
}
```

Figure 28 Patron de requête SPARQL permettant de retrouver les composants d'un processus et leurs types

Par exemple, une fois chargée la représentation RDF de processus de la **Figure 25** du chapitre 3, l'instanciation du patron de requête de la **Figure 28** par instanciation de la variable `?p` par l'URI du processus permet d'interroger la représentation RDF du processus et de retrouver les trois processus associés aux classes `Pente`, `Cadre` et `CellulePV`. Récursivement, le moteur de processus interrogera la représentation de ces processus, jusqu'à déterminer leurs composants de type `Test`, c'est-à-dire les couples de requêtes SPARQL à exécuter.

Nous l'avons vu dans le chapitre 3, c'est en effet un composant de type `Test` qui représente une règle de vérification de contrainte, la contrainte proprement dite étant représentée par la requête conséquent de la règle, valeur de la propriété `kg:then` et les conditions d'application de la règle étant représentées par la requête antécédent de la règle, valeur de la propriété `kg:if`. Le patron de requête présenté sur la **Figure 29** est utilisé dans le cas d'un composant de processus de type `Test` et son instanciation par instanciation de la variable `?q` permet d'identifier les actions à effectuer et leurs types, si une condition est vérifiée.

```
SELECT * WHERE {
   ?q kg:if ?qi
   OPTIONAL {?q kg:then ?qt . ?qt rdf:type ?tt}
   OPTIONAL {?q kg:else ?qe . ?qe rdf:type ?te}
}
```

Figure 29 Patron de requête SPARQL pour retrouver les conséquents d'une règle

2.2 Vérification des contraintes par interrogation de la représentation du Dossier Technique

Les requêtes SPARQL représentant des contraintes réglementaires qui font partie de la description des processus élémentaires qui participent au processus complexe de vérification d'un dossier technique sont exécutées au fur et à mesure qu'elles sont identifiées dans l'interprétation de la représentation RDF du processus complexe par le moteur de processus.

Ces requêtes, nous l'avons vu dans le chapitre 3, sont toutes de type ASK et renvoient donc une valeur booléenne selon qu'un appariement a pu ou non être trouvé entre le graphe requête SPARQL représentant la contrainte et le graphe RDF représentant le dossier technique. Le résultat de l'exécution d'un processus de vérification de contraintes n'est cependant pas une simple valeur booléenne, qui serait la conjonction des résultats des requêtes ASK participant au processus et indiquerait si oui ou non un dossier technique est conforme aux réglementations qui s'appliquent. Nous construisons la liste des couples de requêtes SPARQL représentant une contrainte associées chacune au résultat de son application sur la représentation RDF. C'est l'exploitation de cette liste qui nous permet de guider l'instructeur du CSTB dans sa prise de décision lors de la rédaction d'un avis technique sur le dossier.

3 Aide à la prise de décision

Pour assister les instructeurs CSTB dans la rédaction d'avis techniques sur des dossiers techniques, notre modèle intègre deux types d'aide à la prise de décision. D'une part, les résultats du contrôle automatique de la conformité d'un dossier technique aux règlementations qui s'appliquent sont expliqués à l'utilisateur, notamment en produisant la formulation SBVR correspondant aux requêtes SPARQL qui ont échoué ; et d'autre part les règlementations qui n'ont pas pu être vérifiées automatiquement faute de formalisation SPARQL sont présentées à

l'utilisateur sous la forme de règles opératives SBVR. L'ensemble constitue ce que nous appelons un rapport de conformité du dossier technique, qui aidera l'expert dans sa prise de décision lors de l'émission d'un avis technique.

3.1 Explication du résultat des vérifications automatiques

L'explication des résultats des vérifications automatiques de contraintes, déclenchées par le moteur de processus qui interprète la description RDF d'un dossier technique, repose sur la présentation de la représentation SBVR des contraintes réglementaires qui ne sont pas vérifiées par le dossier technique, c'est-à-dire dont la formalisation en SPARQL ne s'apparie pas avec la représentation RDF du dossier technique. L'interprétation et l'exécution du processus complexe de vérification d'un dossier technique produit la liste des couples de requêtes SPARQL représentant chaque contrainte à vérifier dans le processus, chacune de ces requêtes étant associées au résultat de son application sur la représentation RDF du dossier technique.

La conformité du dossier technique à une contrainte correspond au cas où le résultat de l'application des deux requêtes SPARQL représentant les conditions d'application de la contrainte et la contrainte proprement dite sur la représentation RDF du dossier technique est le couple `(true, true)`. La **Figure 30** suivante montre un exemple d'une telle situation. Dans ce cas le label de la règle est retrouvé pour indiquer en langage contrôlé, dans un rapport de conformité, que dossier technique est conforme à cette contrainte. Sur la **Figure 31**, il s'agit des lignes de texte en vert.

La non-conformité du dossier technique à une contrainte correspond au cas où le résultat de l'application des deux requêtes SPARQL représentant la contrainte et ses conditions d'application sur la représentation RDF du dossier technique est le couple `(true, false)`. Ce serait par exemple le cas d'un dossier technique dont la représentation RDF serait identique à celle sur la **Figure 30** où la valeur de la propriété `odt:aPourRecouvrement` serait 6. Dans ce cas, nous interrogeons la base des représentations RDF des processus élémentaires grâce à un patron de requêtes SPARQL pour retrouver le label et la représentation SBVR de la règle qui a échoué et indiquer dans le rapport de conformité que cette contrainte n'est pas vérifiée. Sur la **Figure 31**, il s'agit des lignes de texte en rouge. Le rapport de conformité ainsi généré, capture le « pourquoi » de la prise de décision négative.

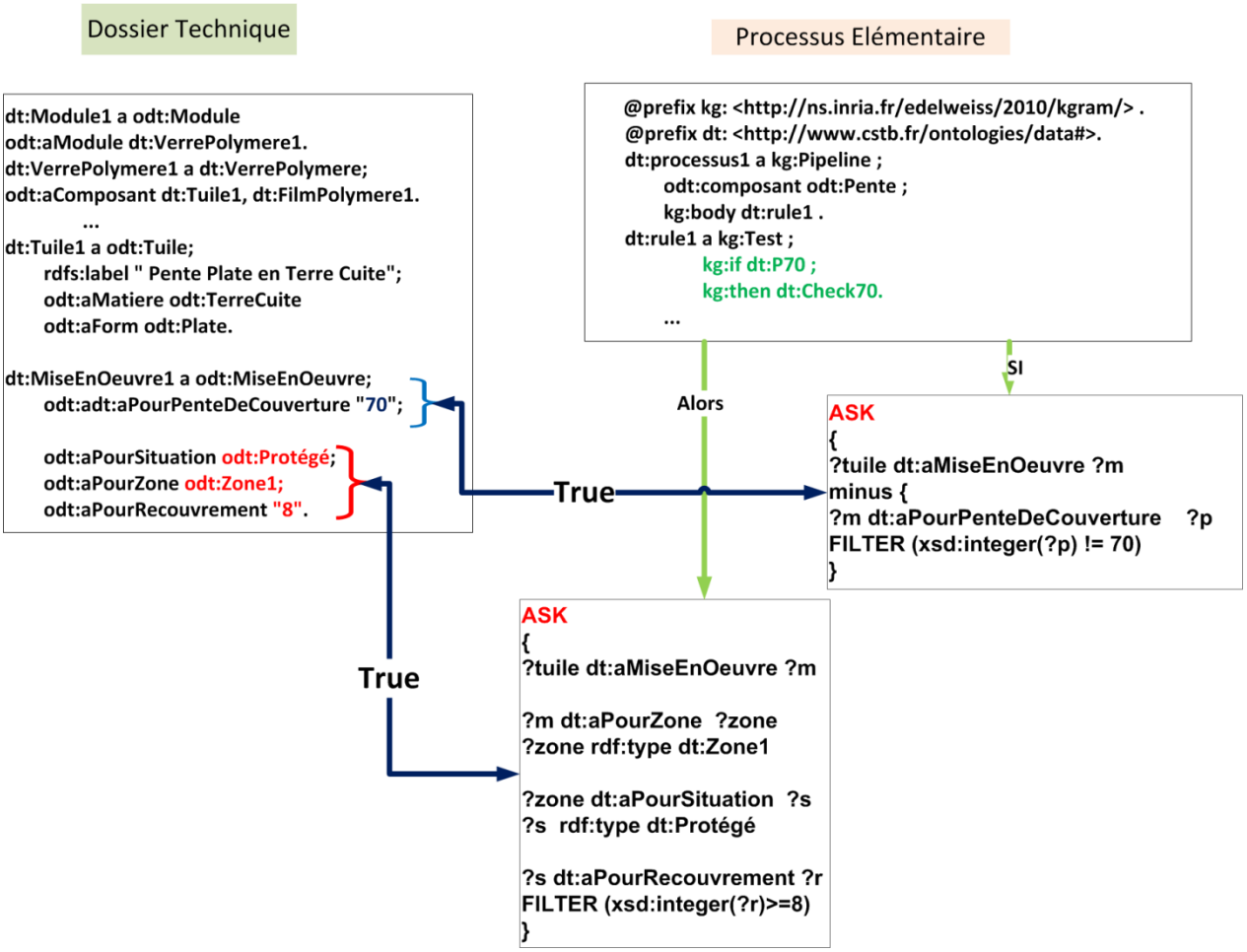


Figure 30 Exemple d'un dossier technique conforme à une contrainte élémentaire

Notons que le cas où le résultat de l'application de la requête SPARQL représentant les conditions d'application d'une règle sur la représentation RDF du dossier technique est la valeur false, correspond à une règle qui ne s'applique pas au dossier technique.

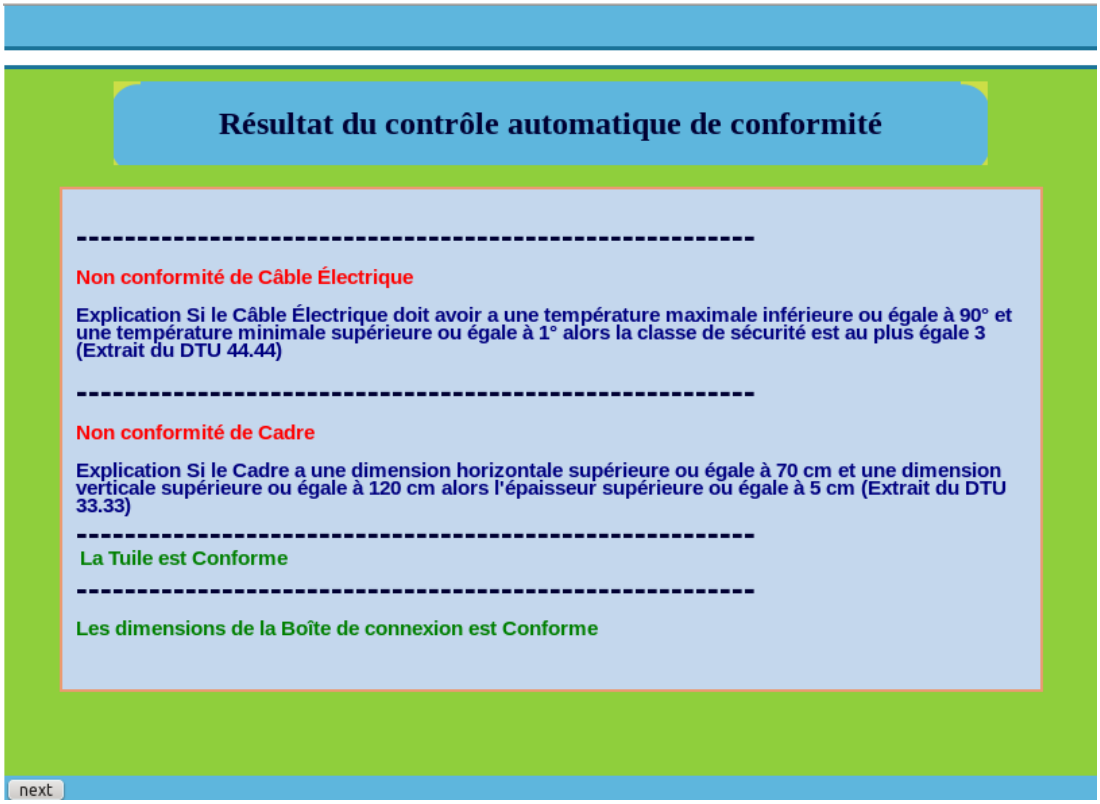


Figure 31 Exemple d'un rapport de conformité

3.2 Aide à la vérification manuelle de contraintes

Nous l'avons vu dans le chapitre 3, les contraintes réglementaires rencontrées dans les guides pratiques du CSTB ne peuvent pas toutes être formalisées avec des requêtes SPARQL. Ces contraintes, non formalisables, sont essentiellement des règles de mise en œuvre, comme celle décrite sur la **Figure 32**.

```
@prefix odt: <http://www.cstb.fr/ontologies/odt#> .
@prefix dt: http://www.cstb.fr/ontologies/Data#> .

dt:processus3 a kg:Pipeline ;
              kg:body (dt:rule8).
dt:rule8 odt:hasSBVRrule dt:sbvr12 ;
         odt:hasValue "Si les arêtiers sont scellés ou à sec alors
              les tuiles sont tranchées biaises à la ligne
              d'arêtier".
```

Figure 32 Extrait de la représentation RDF d'une contrainte de mise en œuvre

L'identification des contraintes qui restent à vérifier manuellement repose sur l'instanciation d'un premier patron de requête (**Figure 33**) permettant d'interroger le processus complexe de

vérification de conformité d'un dossier technique et de lister toutes les règles qui le composent.

```
SELECT ?rule WHERE {
     ?p rdf:type kg:Pipeline
     ?p kg:body ?x
     ?x rdf:rest*/rdf:first ?rule
     ?rule rdf:type kg:Test
}
```

Figure 33 Patron de requête SPARQL d'extraction des règles de conformité

L'instanciation d'un second patron de requête (**Figure 34**) permet d'extraire les représentations SBVR des contraintes qui n'ont pas de représentation SPARQL.

```
SELECT ?t WHERE {
     ?rule odt:hasSBVR ?srule
     ?srule odt:hasValue ?t
     ?rule rdf:type kg:Test
     OPTIONAL { ?rule kg:then ?y}
     FILTER ( !BOUND(?y) )
}
```

Figure 34 Patron de requête SPARQL d'extraction des représentations SBVR des règles

Ces représentations SBVR sont intégrées au rapport de conformité du dossier technique (**Figure 35**) et constituent pour les instructeurs du CSTB un résumé des vérifications restant à effectuer manuellement.

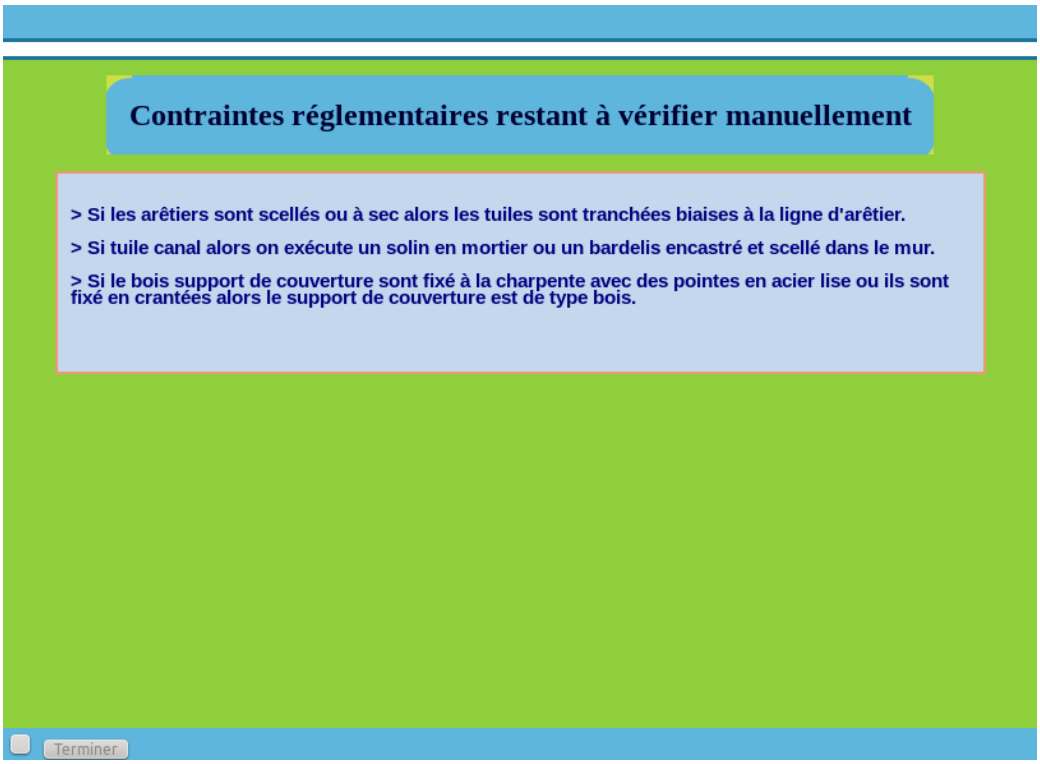


Figure 35 Exemple de rapport de conformité (suite)

4 Conclusion

Nous avons présenté dans ce chapitre un modèle de vérification de la conformité d'un dossier technique sur lequel un avis technique est demandé. Comme cela apparait sur la **Figure 36** suivante qui résume notre modèle, on distingue :

- une automatisation de certaines vérifications de la conformité du produit décrit dans le dossier technique aux réglementations formalisées ;
- la production automatique d'explications des résultats de non-conformité issus des vérifications automatiques ;
- la production des contraintes restant à vérifier par l'expert du CSTB, sous la forme de règles SBVR.

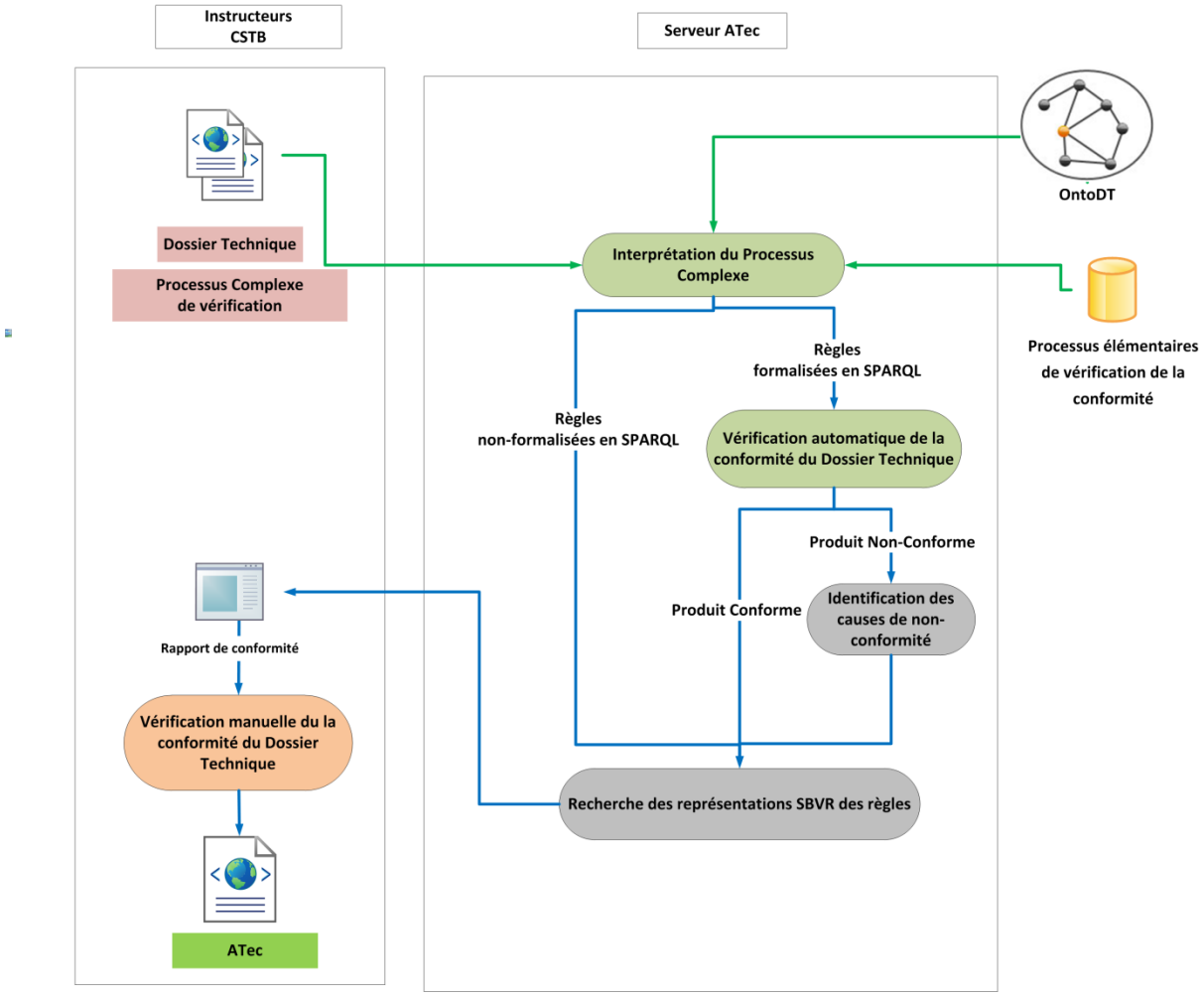


Figure 36 Aide à la rédaction d'un avis technique

Notre contribution consiste en un modèle unifié qui repose sur l'interrogation des représentations RDF du dossier technique, du processus de vérification du dossier technique et des contraintes réglementaires, à l'aide d'un moteur sémantique :

- Le processus de vérification de la conformité du dossier technique est interprété par un moteur de processus qui repose sur l'interrogation de la description RDF du processus grâce à des patrons de requêtes SPARQL prédéfinies.
- Le contrôle automatique (partiel) du dossier technique repose sur l'interrogation de la description RDF du dossier technique grâce aux requêtes SPARQL identifiées par le moteur de processus, qui représentent certaines contraintes réglementaires.
- L'explication des résultats de non-conformité produits par les contrôles automatiques repose sur l'interrogation de la représentation RDF des règles de conformité non vérifiées pour en extraire la représentation SBVR.

- La production des règles de conformité restant à vérifier repose également sur l'interrogation de la représentation RDF des règles de conformité non représentées en SPARQL pour en extraire la représentation SBVR.

Le rapport de conformité qui rassemble ces résultats est destiné aux instructeurs du CSTB pour les assister dans leur travail de vérification des dossiers techniques et de rédaction des avis technique. L'implémentation de notre modèle de vérification de la conformité d'un dossier technique au réglementations qui s'appliquent nous a permis de développer un système d'aide à la rédaction d'avis techniques à destination des instructeurs du CSTB que nous décrivons dans le chapitre suivant.

CHAPITRE 5
EVALUATION DU SYSTEME D'AIDE A LA REDACTION D'AVIS TECHNIQUE

1 Introduction 104
2 Spécifications 104
2.1 Spécifications fonctionnelles 104
2.2 Spécifications techniques 112
3 Conception 113
3.1 Modèle MVC global 113
3.2 Modèle MVC de nos deux outils 115
4 Validation et évaluation 118
4.1 Validation des représentations des contraintes réglementaires 118
4.2 Validation de l'ontologie du Document Technique (OntoDT) 120
4.3 Evaluation et validation du processus de vérification de conformité 122
4.4 Evaluation et validation des systèmes d'aide à la rédaction de dossiers techniques et d'avis techniques 123
5 Conclusion 124

1 Introduction

Nous avons présenté dans les trois chapitres précédents notre approche pour l'aide à la rédaction d'avis technique. Dans le chapitre 2, nous exposons la modélisation que nous proposons du dossier technique et des connaissances de domaine afin de supporter le processus de rédaction de Documents Techniques. Dans le chapitre 3, nous présentons la modélisation que nous proposons des contraintes réglementaires et des pratiques en matière de contrôle de la vérification de ces contraintes, à partir des guides pratiques du CSTB. Et dans le chapitre 4, nous avons montré comment nous utilisons cette modélisation pour aider à la rédaction d'avis techniques.

En reposant sur l'approche présentée dans les chapitres 2, 3 et 4, nous avons construit un outil d'aide à la rédaction de dossier technique à destination des industriels et un outil d'aide à la rédaction d'avis technique à destination des instructeurs du CSTB. Dans ce chapitre, nous présentons les spécifications et les modèles de conception des outils développés.

Ensuite, nous présentons les résultats de la mise en œuvre et l'évaluation de notre approche et de nos outils. Pour ce faire, nous présentons les différentes expérimentations effectuées, en soulignant leurs caractéristiques principales.

2 Spécifications

La **Figure 37** résumé le principe de la demande d'un avis technique émise par un industriel au CSTB. Les deux outils développés sont complémentaires pour assister l'industriel dans la rédaction d'un dossier technique sur son produit et pour assister les instructeurs du CSTB dans la rédaction d'un avis technique sur le dossier technique soumis.

2.1 Spécifications fonctionnelles

2.1.1 Outil d'aide à la rédaction de documents techniques

Cet outil a pour but d'assister les industriels dans leur élaboration du dossier technique. L'industriel débute son interaction avec l'outil sur une première page web où il doit choisir le

domaine d'application. Il lui est alors proposé une liste de composants de haut niveau pour initier les étapes de rédaction à suivre.

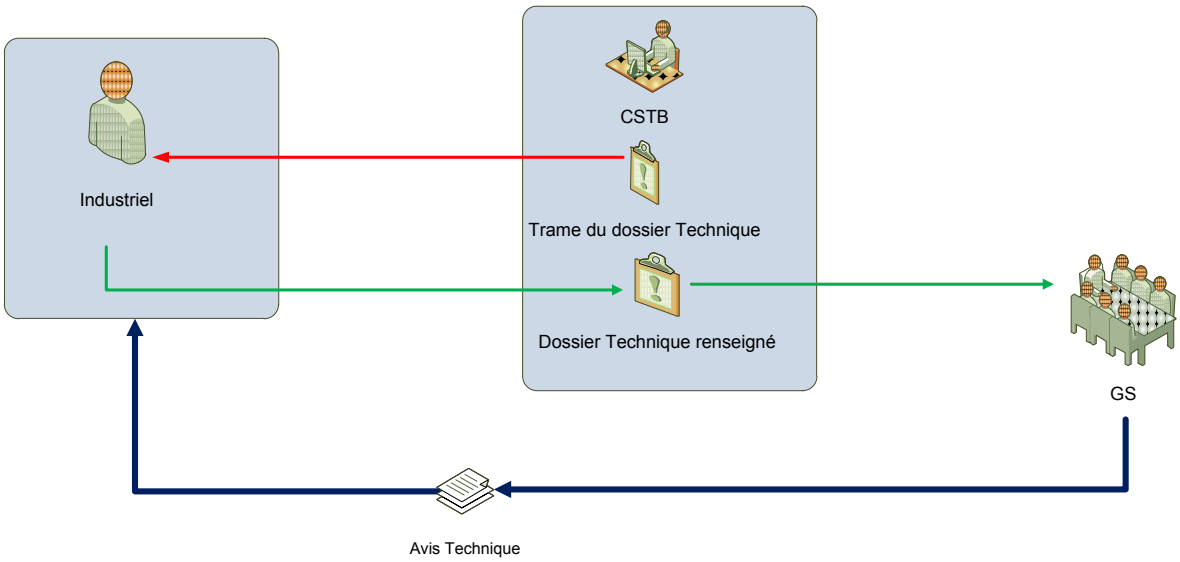


Figure 37 Principe de la demande d'avis technique sur un produit ou procédé

Plusieurs fonctions composent notre outil qui permettent à l'industriel de renseigner efficacement les informations relatives à son produit. En effet, à partir des choix effectués par l'industriel et des premières informations renseignées, les étapes à suivre sont générées à la volée en reposant sur l'ontologie du domaine OntoDT, sans que l'industriel ne se pose de questions sur les niveaux de détail à renseigner.

En fonction du choix effectué, l'outil interroge l'ontologie du domaine puis renvoie au fur et à mesure des renseignements plus précis. Ces informations permettent d'établir un Dossier Technique valide tant sur la forme (les différentes parties sont renseignées avec le niveau de granularité adéquat) que sur le fond (le renseignement des informations est guidé par l'ontologie OntoDT et la valeur sémantique des informations est ainsi assurée).

2.1.1.1 Création dynamique des formulaires du dossier technique

Cette fonction permet à l'outil de mettre en place toutes les informations nécessaires à l'élaboration d'un document technique pour un produit donné.

Entrées

Cette fonctionnalité repose sur l'interrogation de l'ontologie OntoDT.

Le chargement dynamique des formulaires du dossier technique est initié par une première « Entrée » faite par l'industriel. Cette entrée consiste à choisir dans une liste quel sera le composant principal sur lequel va porter le dossier technique **Figure 38.**



Figure 38 Interface d'accueil industriel de l'outil d'aide à la rédaction d'Avis Techniques

Cette liste contient les informations relatives aux composants entrant dans la définition du produit décrit et aux propriétés structurelles et dimensionnelles du produit **Figure 16** .

Traitement

L'outil met en place à partir des données choisies par l'industriel, l'ensemble des connaissances nécessaires au renseignement du dossier technique. Il détermine notamment :

- Les composants à décrire. Un patron de requête SPARQL est appliqué afin d'interroger récursivement les définitions des classes de l'ontologie OntoDT représentant les composants du produit à décrire.

- Les formulaires associés à chaque composant à décrire (**Figure 38).**

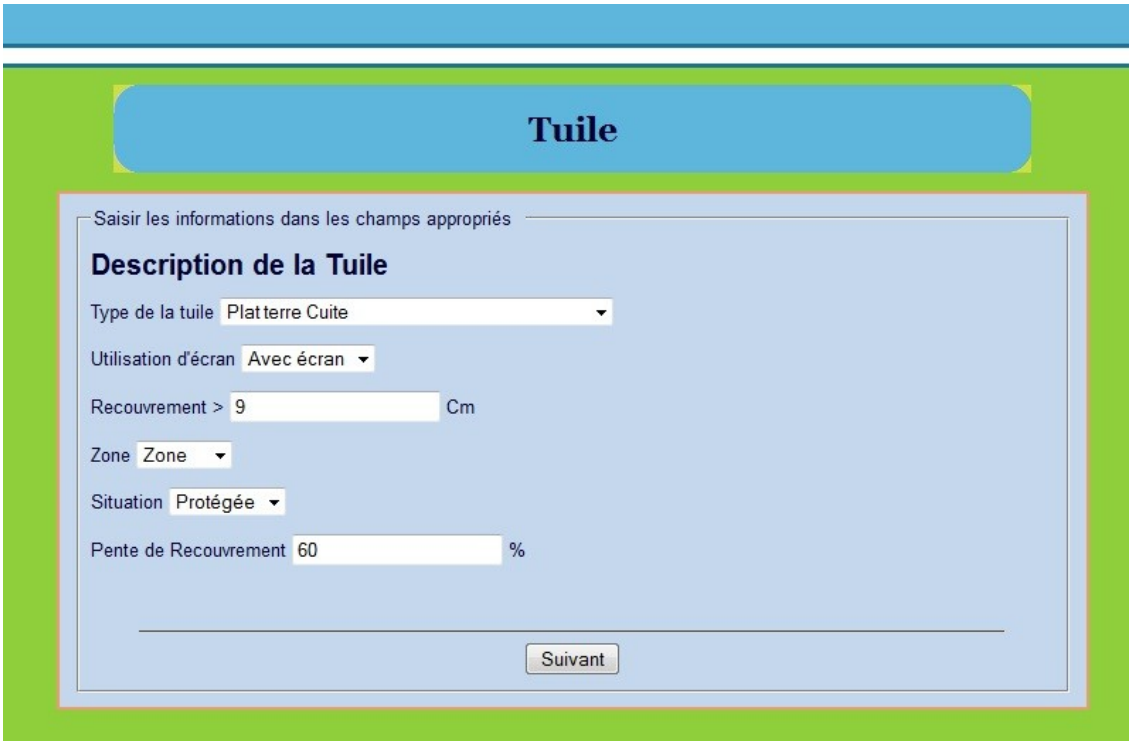


Figure 39 Exemple de formulaire

Sorties

L'outil produit des pages web contenant les formulaires correspondant à chaque composant. Ces formulaires sont interdépendants selon un lien explicite issu de la définition des concepts qui les représentent dans l'ontologie OntoDT.

L'enchainement des formulaires produits s'effectue de façon transparente pour l'industriel qui n'a pas à gérer la décomposition des éléments ni la granularité de la description à faire. Tout ceci étant formalisé dans l'ontologie, laquelle sert de source pour la génération des formulaires.

2.1.1.2 Création de la représentation RDF du Dossier Technique

Entrées

Les informations renseignées par l'industriel.

L'ontologie OntoDT

Traitement

Le système construit une description RDF du Dossier Technique en interrogeant les définitions de classes de l'ontologie OntoDT.

Sortie

Un fichier RDF contenant la représentation formelle du dossier technique renseigné par l'industriel (**Figure 40**).



Figure 40 Interface de fin de création d'un Dossier Technique

2.1.1.3 Création du processus de vérification de la conformité d'un dossier technique

Comme nous l'avons vu dans les chapitres 3 et 4, en même temps qu'il crée une représentation RDF d'un Dossier Technique (Chapitre 3 & 4), l'outil d'aide à la rédaction d'un dossier technique crée automatiquement la description RDF du processus complexe de vérification de la conformité du Dossier Technique à destination de l'outil d'aide à la rédaction d'avis techniques (**Figure 40**).

Entrées

- OntoDT
- Représentations RDF des processus élémentaires de contrôle de conformité

Traitement

- Identification des processus élémentaires de vérification de conformité pertinents pour le dossier technique en cours de rédaction.
- Construction du processus complexe de vérification de conformité.

Sortie

Un fichier RDF contenant la représentation formelle du processus complexe de vérification de la conformité du dossier technique.

2.1.2 Outil d'aide à la rédaction d'avis techniques

Cet outil vise à assister les instructeurs du CSTB dans la rédaction d'avis techniques. Il vérifie automatiquement la conformité du produit décrit dans le dossier technique aux contraintes réglementaires qui ont pu être formalisées et produit un rapport de conformité qui explique le cas échéant quelles contraintes ne sont pas vérifiées et quelles contraintes restent à être vérifiées manuellement.

2.1.2.1 Vérification automatique de la conformité d'un Dossier Technique

Entrées

- La représentation RDF d'un dossier technique
- La représentation RDF du processus de vérification de la conformité de ce dossier

Traitement

- Les représentations RDF du Dossier Technique et de son Processus de vérification de conformité sont chargées simultanément.
- Le moteur de processus interprète la représentation du processus complexe. Les processus élémentaires de conformité sont identifiés et les requêtes SPARQL relatives aux contraintes interprétables sont chargées.
- Le moteur KGRAM exécute ces requêtes sur la description RDF du Dossier Technique.

Sorties

Les résultats d'exécution des requêtes SPARQL.

A partir des résultats de ces requêtes, l'outil génère un rapport de conformité en langue naturelle, comme décrit ci-après.

2.1.2.2 Explication des résultats des vérifications automatiques

Entrée

La représentation RDF du processus de vérification de la conformité de ce dossier

Traitement

Cette fonctionnalité repose sur l'exploitation de la description RDF des contraintes réglementaires qui associe représentation formelle en SPARQL et représentation SBVR.

Sorties

Une page web contenant le rapport de conformité automatique en langage naturelle. Pour chaque contrainte testée, soit elle est soit déclarée vérifiée soit sa représentation SBVR est présentée pour expliquer ce qui n'est pas vérifié **Figure 41**.

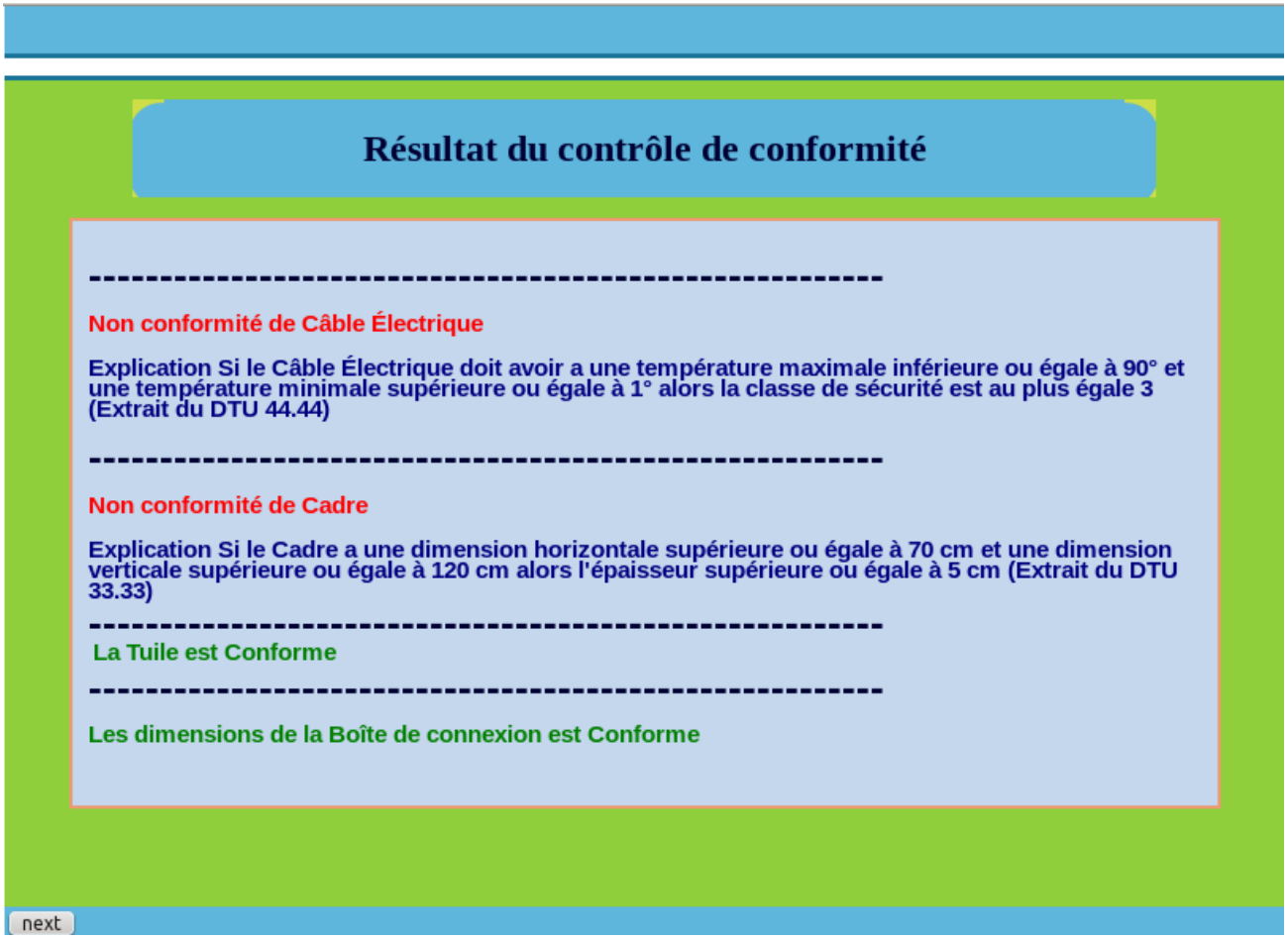


Figure 41 Rapport automatique de conformité

2.1.2.3 Aide à la vérification manuelle

Les contraintes qui n'ont pas de formalisation SPARQL ne sont pas vérifiées automatiquement. Le moteur de processus les repères lors de l'interprétation de la description RDF du processus complexe et elles sont listées à l'utilisateur.

Entrées

La représentation RDF du processus de vérification de la conformité de ce dossier

Traitement

Cette fonctionnalité repose sur l'exploitation de la description RDF des contraintes réglementaires qui associe représentation formelle en SPARQL et représentation SBVR.

Sorties

Une page web contenant le rapport de conformité automatique en langage naturelle. La représentation SBVR de chaque contrainte non testée automatiquement est présentée pour indiquer ce qui reste à vérifier (**Figure 42**).

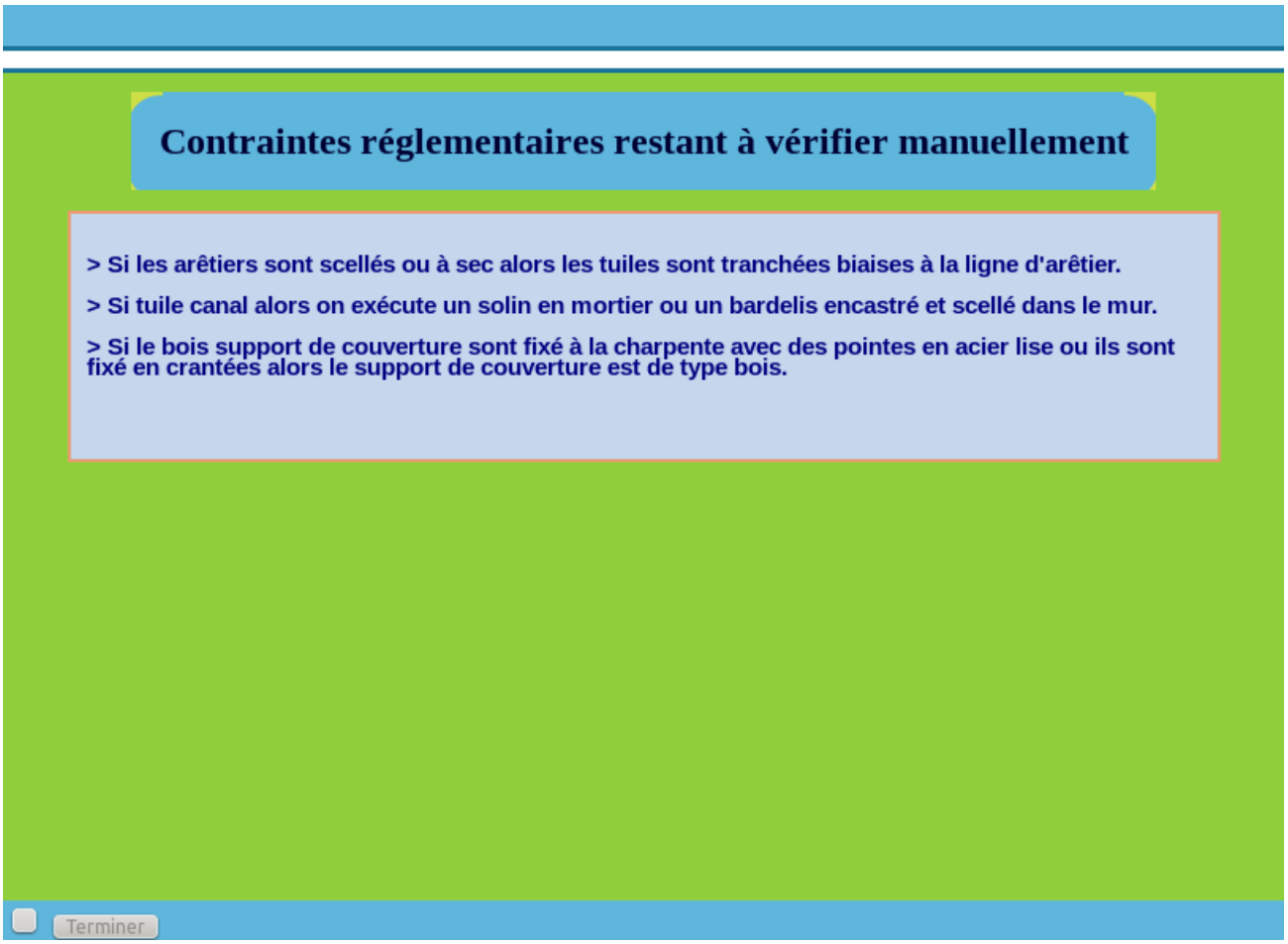


Figure 42 Rapport de conformité

2.2 Spécifications techniques

Les outils d'aide à la rédaction de Dossiers Techniques et d'Avis Technique doivent pouvoir être utilisés indifféremment sur une plateforme PC/MAC/LINUX connectée à internet. Pour cela, nous avons développé ces deux outils dans le langage Java sous plateforme J2EE (JSP/Servlet). L'outil est déployé en utilisant le serveur Web Tomcat d'Apache.

Ils doivent pouvoir être utilisés indépendamment de tout autre outil ou logiciel. Pour cela, nos outils utilisent uniquement le moteur sémantique Corese/KGRAM qui est Open Source.

Les formats de données utilisés doivent être suffisamment documentés pour permettre une interopérabilité avec d'autres systèmes. C'est bien le cas des données RDF, RDFS, OWL SPARQL et SBVR que manipulent nos deux outils.

3 Conception

En vue du déploiement sur le web des outils d'aide à la rédaction de dossiers techniques et d'avis techniques, nous avons choisi le modèle MVC (Model-View-Controller).

3.1 Modèle MVC global

L'architecture MVC (Modèle-Vue-Contrôleur)[27] est un patron de conception qui est très utilisé de nos jours dans de nombreux domaines et langages (Knight and Dai, 2002). Ce modèle est dédié au développement d'applications logicielles en séparant le modèle de données, l'interface utilisateur et la logique de contrôle. Un avantage du modèle MVC permet d'avoir non seulement des objets réutilisables pour d'autres applications, mais aussi de pouvoir faire évoluer aisément son programme (Luong, 2007). Ainsi, si l'on souhaite modifier sa base de données, il suffit de revoir son modèle et cela est aussi valable dans le cas où l'on souhaite changer d'interface. Les 3 parties du model MVC sont réellement autonomes.

Le modèle de conception impose donc une séparation en trois couches :

- Model (Modèle) : Le modèle représente le comportement de l'application : traitement des données, interactions avec la base de données, etc. Il décrit ou contient les données manipulées par l'application. Il assure la gestion de ces données et garantit leur intégrité.
- Controller (Contrôleur) : Le contrôleur prend en charge la gestion des événements de synchronisation pour mettre à jour la vue ou le modèle et les synchroniser. Il reçoit tous les événements de l'utilisateur et enclenche les actions à effectuer. Si une action nécessite un changement des données, le contrôleur demande la modification des données au modèle et ensuite avertit la vue que les données ont changé pour qu'elle se mette à jour. Certains événements de l'utilisateur ne concernent pas les données mais la vue. Dans ce cas, le contrôleur demande à la vue de se modifier. Le contrôleur

[27] http://fr.wikipedia.org/wiki/Mod%C3%A8le-Vue-Contr%C3%B4leur

n'effectue aucun traitement, ne modifie aucune donnée. Il analyse la requête du client et se contente d'appeler le modèle adéquat et de renvoyer la vue correspondant à la demande.

- View (Vue) : La vue correspond à l'interface avec laquelle l'utilisateur interagit. Sa première tâche est de présenter les résultats renvoyés par le modèle. Sa seconde tâche est de recevoir toutes les actions de l'utilisateur (clic de souris, sélection d'une entrée, boutons, …). Ces différents événements sont envoyés au contrôleur. La vue n'effectue aucun traitement, elle se contente d'afficher les résultats des traitements effectués par les couches inférieures.

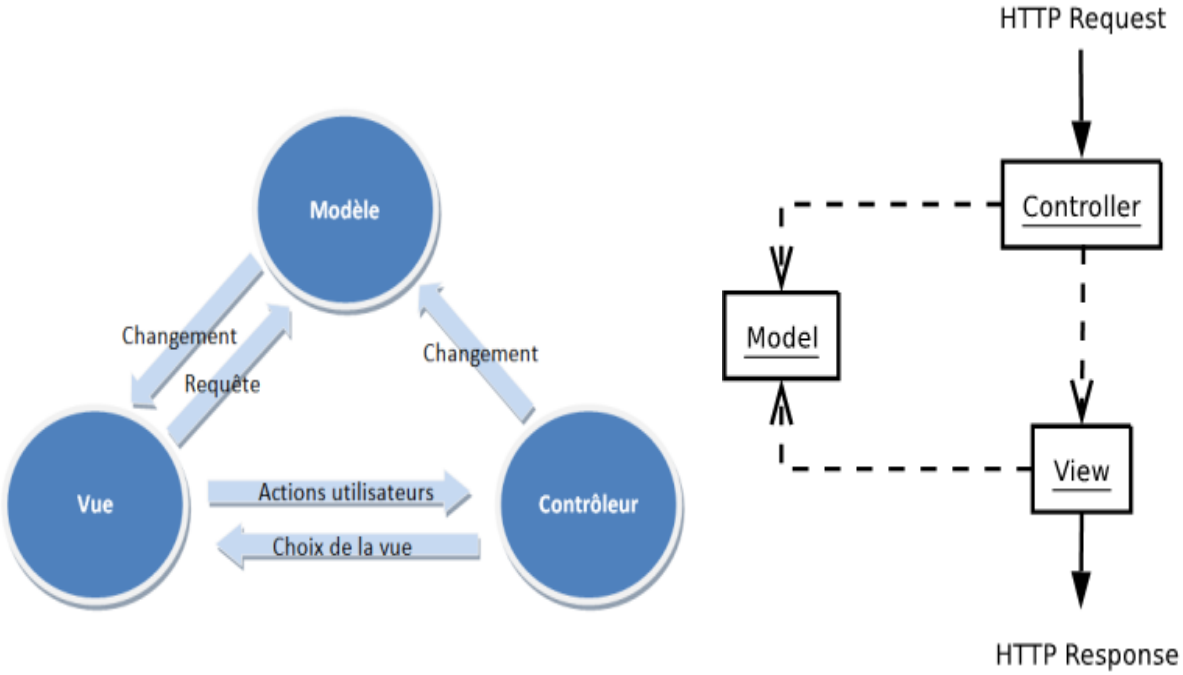


Figure 43 Modèle global MVC

La **Figure 43** représente le modèle MVC global et le diagramme de séquence d'un modèle MVC simple. La couche « contrôleur » analyse la requête, et éventuellement extrait les informations dans l'URL de la requête. Elle fait ensuite une mise à jour du modèle si nécessaire, en lui passant des paramètres, détermine la vue à afficher et demande son affichage.

3.2 Modèle MVC de nos deux outils

Nous avons appliqué le modèle MVC en trois couches à la conception de nos outils. Les couches *Contrôleur* et *Vue* ont les mêmes fonctionnalités dans l'outil d'aide à la rédaction de dossier technique et l'outil d'aide à la rédaction d'avis techniques. Le contrôleur (servlet) a pour but de recevoir des requêtes de l'utilisateur, d'analyser les requêtes et ensuite de transférer des requêtes avec les paramètres nécessaires aux fonctions correspondantes à la couche modèle. Elle est aussi dédiée à la synchronisation du modèle et de la vue. Elle reçoit tous les événements de l'utilisateur et enclenche les actions correspondantes à effectuer.

La couche *Vue* contient une interface entièrement implémentée sous forme de pages. Ces pages permettent, côté industriels, l'affichage des formulaires relatifs à chaque composant et, côté instructeurs CSTB, l'affichage des rapports de conformité en langage naturel.

Dans l'outil d'aide à la rédaction de dossiers techniques, la couche *Modèle* contient les classes qui permettent d'interroger des données RDF à l'aide de requêtes SPARQL, via KGRAM, en prenant en compte des connaissances ontologiques représentées en OWL Lite. Egalement, elle comprend l'ontologie du dossier technique ainsi que les informations renseignées par l'industriel stockées sous forme de descriptions RDF. En effet, elle définit des méthodes d'accès dédiées aux tâches spécifiques telles que la gestion des choix de produit à renseigner. En fonction des choix effectués par l'industriel dans ce premier niveau, l'outil s'adapte et parcourt l'ontologie pour demander au fur et à mesure des renseignements plus précis. D'un point de vue technique, ces enchainements de formulaires sont orchestrés grâce à un ensemble requêtes SPARQL qui interroge l'ontologie OntoDT. Cette couche se base essentiellement sur l'API KGRAM qui permet la manipulation et l'exécution des requêtes SPARQL sur l'ontologie OntoDT et la génération des représentations RDF du Dossier Technique ou *et* de son processus de contrôle de conformité.

Dans l'outil d'aide à la rédaction d'avis techniques, la couche *Modèle* contient des classes qui permettent de charger les descriptions RDF d'un Dossier Technique et de son processus complexe de vérification de sa conformité et d'interpréter la description du processus pour contrôler le dossier technique. Le moteur de processus utilise l'API KGRAM, pour charger la description RDF du Dossier Technique et celle du de son processus complexe et pour exécuter différents patrons de requêtes. En effet, il permet d'exécuter, à la demande, une ou plusieurs requêtes séquentiellement grâce à l'interprétation du modèle de processus. Il est appelé récursivement à chaque invocation d'un nouveau processus.

Concrètement, ce moteur consiste en une classe Java Pipe, dont les méthodes appellent certaines méthodes de classes du moteur KGRAM pour permettre : (1) le chargement d'une annotation ou d'une ontologie, (2) l'exécution d'une requête SPARQL, (3) l'exécution d'une ou plusieurs requêtes quand une condition spécifique est satisfaite. Le moteur de processus interprète alors la représentation RDF du processus pour identifier les différentes actions à effectuer.

En effet, dès le chargement d'un processus (élémentaire ou complexe) un premier patron de requête est appliqué afin de lister les différents composants de ce processus. Chaque composant du processus est traité selon son type par l'appel à une fonction de KGRAM, et le traitement de chaque processus peut déclencher récursivement le traitement d'autres processus y compris lui-même. L'algorithme général du moteur de processus est décrit dans la **Figure 44** suivante.

```
Fonction run(liste des composants d'un processus p)
      Pour chaque composant c du processus p
      Si (type(c) = PIPE)
      Alors pipe (c) // récursion : la méthode pipe appelle la méthode run
      Sinon      Si (type(c) = LOAD)
                 Alors load(c)
                  Sinon  Si (type(c) = QUERY)
                         Alors query (c)
                         Sinon  Si (type(c) = Test)
                                Alors test(c)
```

Figure 44 Algorithme du moteur de processus

La méthode `load` permet de charger la description RDF du document technique à vérifier. Son paramètre peut être une `URI` ou un chemin vers un fichier.

La méthode `query` permet de charger et d'exécuter une ou plusieurs requêtes SPARQL.

La méthode `pipe` permet d'invoquer un ou plusieurs autres processus.

La fonction `test` permet quant à elle d'exécuter une ou plusieurs requêtes si une condition spécifique est vérifiée. Pour cela, un patron de requête est exécuté afin d'identifier les actions à effectuer et leurs types, si une condition particulière est vérifiée.

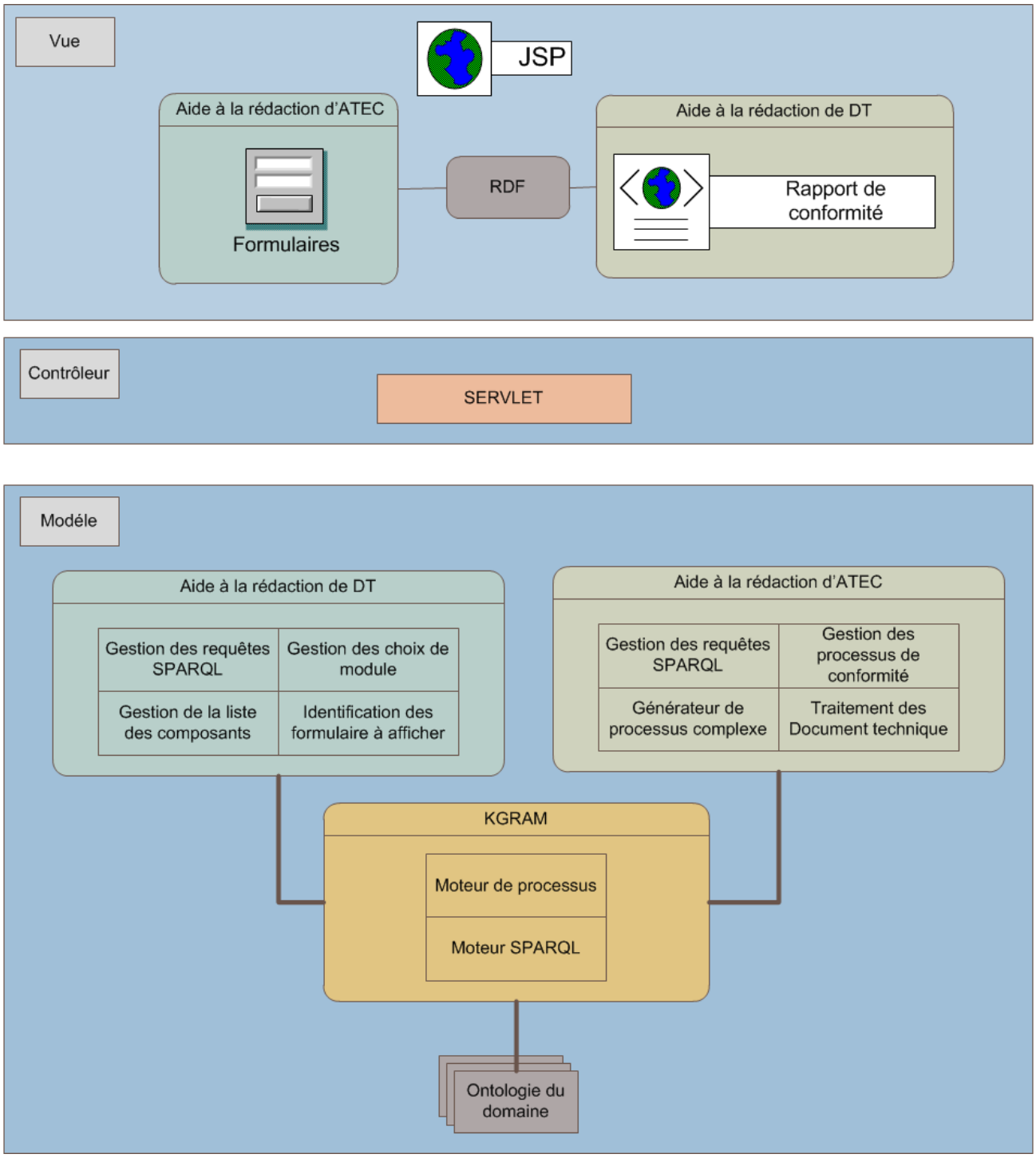


Figure 45 Modèle MVC appliqué pour nos outils

4 Validation et évaluation

Nous avons identifié quatre principaux aspects à considérer pour valider nos travaux :

1. La validation des représentations des contraintes réglementaires
2. La validation de l'ontologie du Document Technique (OntoDT)
3. La validation du processus de vérification de la conformité d'un dossier technique
4. L'évaluation et la validation des outils d'aide à la rédaction de documents techniques et d'aide à la rédaction d'avis technique

Nous avons évalué les outils implémentés dans des contextes et scénarios d'utilisation réels, afin de mesurer leur *utilisabilité* et le gain apporté aux utilisateurs. Ces outils visant à simplifier la rédaction de Dossiers Techniques pour les industriels et la vérification par les experts du CSTB de la conformité de ces documents par rapport aux normes en vigueur, l'évaluation de nos outils par les experts du domaine est une tâche essentielle pour déterminer l'efficacité de notre contribution.

4.1 Validation des représentations des contraintes réglementaires

Afin d'évaluer le niveau de formalisation des exigences réglementaires, nous avons effectué une analyse comparative des travaux effectués en amont dans la thèse de Yurchyshyna (Yurchyshyna, 2009) et dans le projet ISTforCE (Turk et al., 2002) mentionnés dans le chapitre 1.

Dans le projet ISTforCE, six documents juridiques relatifs à l'accessibilité en France ont été étudiés par des experts qui en ont extrait 163 règles de conformité. La plupart de ces exigences (72%) ont été classées comme non-vérifiables ; 6% des exigences signalées étaient partiellement vérifiables et 22% étaient complètement vérifiable.

Dans les travaux de (Yurchyshyna, Faron-Zucker, Thanh, et al., 2008), neuf textes réglementaires relatifs l'accessibilité ont été choisis et ont permis d'extraire 350 contraintes réglementaires de conformité. Ces exigences ont été classées en trois catégories : 5% de ces exigences étaient complètement interprétables, 25% nécessitaient une reformulation du modèle IFC à contrôler, 45% étaient partiellement interprétables et 25% non interprétables.

En prenant ces résultats comme point de départ, nous avons interviewé trois experts du GS 21 spécialisés dans les Avis Techniques Photovoltaïques au CSTB pour analyser la base documentaire des réglementations qui valident ces ATec.

Tout d'abord pour vérifier la conformité des Dossiers Techniques, les experts du CSTB se réfèrent à des normes, DTU et guides pratiques. Nous avons choisi le guide pratique des couvertures en tuiles, ouvrage de 104 pages, qui résume les exigences réglementaires de 7 DTU et exprime environ 90% de l'ensemble des exigences relatives aux différents types de tuiles et leurs caractéristiques. Il définit leurs conditions de mise en œuvre ainsi que leurs critères de vérification. Nos entretiens avec les experts du GS 21nous ont permis d'arriver aux résultats suivants :

- Nous avons construit une base de 164 règles SBVR et de 177 requêtes SPARQL
- 100% des contraintes réglementaires issues des guides pratiques pourraient être modélisables en SBVR mais seulement 30% formalisables en SPARQL
- 70% des règles SBVR portant sur la mise en œuvre des tuiles sont non-transformables en SPARQL.

Afin de valider nos modélisations nous avons établi un questionnaire d'évaluation que nous avons soumis aux experts du CSTB. Les questions posées sont les suivantes :

- Acceptez-vous qu'un texte réglementaire soit décomposé en plusieurs exigences et récrit en SBVR en utilisant un vocabulaire contrôlé ?
- Approuvez-vous la transformation des exigences réglementaires en règles d'obligation SBVR ?
- Approuvez-vous le résultat de la transformation des règles SBVR en requête SPARQL ?

Plusieurs réunions ont eu lieu avec trois instructeurs CSTB afin de leur présenter nos transformations de contraintes réglementaires premièrement en SBVR et ensuite en SPARQL. Lors de ces rencontres nous avons présenté nos résultats de modélisation en SBVR et SPARQL, expliqué brièvement le standard SBVR et le langage SPARQL pour une meilleure compréhension. Nous avons aussi mis le point sur la structure des modélisations afin de nous assurer que nous ne perdions pas le sens des réglementation lors de leur représentation en SBVR puis surtout en SPARQL et pour lever tout problème d'ambiguïté réglementaire.

Nous avons demandé aux trois instructeurs CSTB de répondre à ces questions pour l'évaluation de notre modélisation des contraintes réglementaires. Nous avons réalisé

quelques modifications de modélisation suite à ces entretiens, concernant essentiellement les règles non-transformables en SPARQL. Nous avions en effet mal interprété certains tableaux du guide pratique des couvertures en tuile et nous avons corrigé notre modélisation avec l'aide des experts. Par exemple, notre première modélisation SBVR d'une contrainte réglementaire était la suivante :

> **Si la mise en œuvre doit être en Zone 1 en situation Protégée avec un recouvrement supérieur à 8 cm alors la pente de couverture *est* égale à 70 %**

Les instructeurs CSTB nous ont fortement suggéré de formuler cela autrement. En effet, les instructeurs vérifient la pente de couverture en premier lors de la validation d'un document technique. Une fois la pente de couverture identifiée, ils procèdent alors à la vérification de la zone et de la situation de mise en œuvre ainsi qu'à la valeur de recouvrement. Ceci nous a amené à reformuler notre règle SBVR de la manière suivante :

> **Si la pente de couverture *est* égale à 70 % alors la mise en œuvre doit être en Zone 1 en situation Protégée avec un recouvrement supérieur à 8 cm.**

Pour résumer, nous estimons que toutes les réglementations pourraient être modélisables en SBVR. Cependant, seulement 30% des contraintes réglementaires issues des guides ont été formalisées en SPARQL pour contrôler automatique la conformité des Dossiers Techniques. Les 70% restantes ne sont pas transformables en SPARQL, néanmoins, elles sont exploitées semi-automatiquement, par notre outil, pour aider l'expert à évaluer différentes situations de mise en œuvre.

4.2 Validation de l'ontologie du Document Technique (OntoDT)

L'ontologie du Document Technique OntoDT que nous avons créée, qui définit les composants utilisés dans le domaine photovoltaïque ainsi les connaissances expertes qui y sont relatives, a été validée par des experts du CSTB.

OntoDT est basée sur la trame du Dossier Technique (Bouzidi, Faron-Zucker, Fies, Corby, and Le Thanh, 2012b), elle contient l'ensemble de classes et propriétés qui définissent les composants du domaine photovoltaïque ainsi que leurs caractéristiques structurelles et dimensionnelles. Les experts du CSBTont participé dès le début à la création de l'ontologie OntoDT. En premier lieu, nous avons organisé des réunions afin d'identifier les concepts pertinents issus de la trame du dossier technique. En second lieu, les experts ont répondu à un

questionnaire d'évaluation qui leur a été proposé en fin de processus d'élaboration afin de valider notre ontologie. Les questions posées ont été les suivantes :

- Etes-vous d'accord avec la définition des étapes obligatoires de la description d'un document technique ? Dans notre cas d'utilisation, la description d'un Dossier Technique commence par l'identification de l'industriel demandeur, la caractérisation de son domaine d'exécution, une description dite succincte et sa mise en œuvre.
- Etes-vous d'accord avec les définitions des composants faites dans l'ontologie OntoDT qui seront utilisés dans les descriptions des dossiers techniques ? Par exemple, un module de type verre polymère est composé de verre polymère, d'une cellule photovoltaïque, d'un film polymère, etc **Figure 11**.
- Etes-vous d'accord avec la hiérarchie des classes que nous avons mise en place ? Par exemple, le concept de « *Tuile* » est une généralisation de plusieurs sous concept « *Tuiles à canal* », « *Tuile plate en terre cuite* », etc **Figure 19**.

Finalement, OntoDT a été développée avec l'aide des experts du domaine dans le but de supporter le processus de vérification de la conformité des dossiers techniques. Toutes les classes et les relations de l'ontologie ont été définies ou validées par des experts du domaine comme suit : Nous avons proposé plusieurs définitions aux expert CSTB qui ont servi de base de discussion lors de nos réunions. Cela nous a permis non seulement de mieux comprendre les différentes nomenclatures des produits du domaine photovoltaïque et de modéliser ces produits correctement, mais aussi de remettre en question certains aspects de notre modélisation et d'en valider d'autres. C'est grâce à ce travail collaboratif avec les experts du CSTB que nous avons pu aboutir à une modélisation exploitable des documents techniques.

Pour valider OntoDT, nous avons suivi la méthode de validation proposée par (Karoui et al., 2007)

Qui repose sur 3 critères que nous avons adaptés au domaine de la vérification de la conformité :

1. Tous les concepts et les propriétés utilisées dans le contrôle de conformité sont définis par des experts de la construction et leurs définitions sont pertinentes, exactes et crédibles.
2. L'ontologie de domaine initiale est homogène et cohérente : les experts ont identifié plusieurs concepts généralisés qui peuvent être désignés par plusieurs sous-concepts (par exemple la classe *Tuile*).

3. L'ajout à OntoDT de nouvelles classes ou relations, définies par des experts, est admissible, s'ils sont de nouveaux concepts du domaine ou s'ils sont utilisés dans les *processus de vérification* de conformité.

4.3 Evaluation et validation du processus de vérification de conformité

Nous avons élaboré un modèle de processus de vérification de la conformité d'un dossier technique aux règlementations qui s'appliquent qui synthétise le savoir-faire des experts du domaine, et représente les procédures adoptées par ces derniers pour vérifier la conformité des Dossiers Techniques.

De manière indirecte notre modèle de processus peut être validé en demandant à un expert du CSTB de valider d'une part les résultats des vérifications automatiques de la conformité des composants vis-à-vis des réglementations issues des guides pratiques et d'autre part les explications fournies par le système en cas de non-conformité et la liste fournie par le système des vérifications manuelles restant à réaliser par les instructeurs.

Nous avons conduit des réunions avec des experts chargés des Avis Techniques Photovoltaïques au CSTB, qui n'étaient pas impliqués dans la modélisation des processus de vérification de la conformité et nous leur avons posé la question suivante :

Les processus de vérification illustrent-ils votre façon d'agir pour vérifier la conformité des documents techniques?

Les experts interrogés ont confirmé que ces processus correspondent aux contrôles qu'ils effectuent, et qu'ils reflètent leurs savoir-faire. Les experts du domaine ont également formulé de nouveaux concepts, comme par exemple celui de « Processus élementaire » (qui se distingue du concept de « Processus complexe »), et validé la base de connaissances de l'ensemble du processus de vérification de la conformité.

D'autre part, nous avons vérifié que notre modèle de processus appliqué à des représentations de dossiers techniques, permettait effectivement la validation des caractéristiques structurelles et dimensionnelles des composants rentrant dans la fabrication des modules photovoltaïques.

Cependant, une des limitations de notre modélisation est que les processus de vérification formalisés ne permettent pas de vérifier automatiquement les aspects liés à l'installation, la

mise en oeuvre des composants, et plus généralement toutes pratiques nécessitant des visu expertes du domaine. Il reste donc un défi intéressant, consistant à étudier la possibilité de modéliser de telles connaissances expertes, acquises auprès des instructeurs CSTB spécialistes du domaine photovoltaïque. Dans le cadre de notre thèse, nous n'avons pas abordé cette problématique, qui peut encore être développée comme l'une des perspectives de nos travaux.

4.4 Evaluation et validation des systèmes d'aide à la rédaction de dossiers techniques et d'avis techniques

Nous avons soumis nos deux outils d'aide à la rédaction de Dossiers Techniques, et d'aide à la rédaction d'Avis Techniques à une évaluation par les experts du domaine. Ces derniers ont répondu à un premier questionnaire qui nous a permis par la suite de valider les exigences et fonctionnalités principales des outils.

Il en a résulté que les outils devaient fournir des interfaces (ou des menus) qui permettent aux industriels et instructeurs CSTB d'effectuer les opérations générales suivantes :

Industriel :

- Création de documents techniques en mettant en place un assistant/wizard.
- Génération automatique de la version formelle du document technique
- Construction automatique du processus de vérification du dossier technique

Instructeur CSTB :

- Application automatique du processus de vérification de conformité.
- Génération automatique d'un rapport de conformité.

A l'issue de nos développements, les experts ont également répondu au questionnaire suivant :

1. Est-ce que le prototype répond aux problématiques actuelles (une trame de Dossier Technique mal comprise, trop d'échanges entre le CSTB et les industriels) ?
2. Croyez-vous que l'outil facilitera la tâche des industriels dans leur renseignement de Dossiers Techniques ?

3. Selon votre expérience est ce que les industriels seront prêts à utiliser un tel outil ?
4. Est-ce que l'utilisation d'un logiciel ayant les mêmes fonctionnalités pourra être envisagée dans le futur ?
5. Est-ce que vous serez prêt à consacrer du temps pour formaliser les sources réglementaires que vous utilisez en SBVR et SPARQL (après une formation adéquate)?
6. Est-ce que ce prototype pourra vous faire gagner du temps dans votre travail ?
7. Est-ce-que l'utilisation d'un outil de ce genre pourra diminuer votre charge de travail ?
8. Une fois que les dossiers techniques seront vérifiés, ils seront stockés dans une base sémantiquement riche, est-ce que vous serez prêt à réutiliser cette base dans vos recherches réglementaires ?

Les experts ont certifié que les outils répondaient à l'exigence liée à la rédaction de document technique. Qu'il facilitera le renseignement du document technique en proposant aux industriels des formulaires adéquats aux produits à certifier. Ils ont aussi admis que l'utilisation de tels outils sera un atout, permettant de réduire la charge de travail qui pèse sur les instructeurs. Cependant, ils estiment que ces outils ne permettront pas aux instructeurs CSTB de gagner plus de temps. Ceci est dû au faite que les instructeurs CSTB sont obligés de contrôler chaque partie, même si celle-ci a été vérifiée automatiquement. Une des perspectives identifiées par les instructeurs CSTB est d'identifier les Dossiers Techniques susceptibles de correspondre/ressembler à ceux déjà certifiés.

5 Conclusion

Dans ce chapitre, nous avons présenté les spécifications fonctionnelles et techniques des outils d'aide à la rédaction de Dossiers Techniques et des ATEC. Les deux outils ont été développés en Java, JSP/Servlet ; ils sont déployés dans l'environnement du Web en utilisant le serveur Web Tomcat Apache. Ces outils ont été implémentés en se basant sur un modèle MVC permettant de séparer le modèle de données, l'interface utilisateur et la logique de contrôle.

Ces outils sont à l'état de prototypes. Nous avons pu les mettre en œuvre expérimentalement au CSTB. Nous présentons dans le chapitre suivant le résultat de leur évaluation.

Nous avons également présenté les différentes validations effectuées par les experts du CSTB vis-à-vis de notre approche. L'intervention d'experts du domaine est une tâche essentielle pour l'analyse de la faisabilité et la validation de notre contribution. En premier lieu, nous avons évalué le niveau de formalisation des exigences réglementaires, et effectué une analyse comparative avec les travaux de Yurchyshna (Yurchyshyna, 2009) et ceux du projet ISTforCE (Turk et al., 2002).

En deuxième lieu, les experts du domaine ont évalué notre ontologie du domaine OntoDT et validé les classes et propriétés utilisées pour définir les composants des Dossiers Techniques, ainsi que ceux utilisés dans les processus de vérification de conformité.

En troisième lieu, les expert ont validé notre modélisation des processus de vérification de conformité et conclu qu'ils modélisent correctement les différentes étapes qui permettent de valider un Dossier Technique.

En dernier lieu, les experts ont évalué les deux outils d'aide à la rédaction de dossiers techniques et d'aide à la rédaction d'avis techniques. Ils ont conclu que ces outils répondaient à une problématique réelle d'aide à la création de Dossiers Techniques tout en diminuant la charge de travail des instructeurs du CSTB.

CONCLUSION GENERALE ET PERSPECTIVES

L'objectif général de notre travail de thèse a été de développer des modèles et outils pour aider à la rédaction de documents technico-réglementaires en appliquant les méthodes et outils de l'ingénierie des connaissances. Nous proposons, en guise de conclusion, une relecture de notre travail de thèse selon les trois axes (modélisation du processus de rédaction des dossiers techniques, modélisation des textes réglementaires, contrôle de conformité et aide à la prise de décision). Nous mettrons en évidence nos contributions ainsi que les perspectives de notre travail.

Nous avons dans un premier axe développé une approche pour guider la rédaction de documents techniques. Nous avons adopté une méthode d'ingénierie des connaissances où nous reposons sur une modélisation par une ontologie du domaine abordé par un dossier technique. Nous avons expérimenté sur le domaine particulier du photovoltaïque et nous avons construit manuellement une ontologie de ce domaine à partir d'une part de la trame du dossier technique papier utilisé au CSTB et des guides pratiques du bâtiment édités par le CSTB et d'autre part du thésaurus du REEF développé au CSTB. Nous avons identifié manuellement les termes qui apparaissent dans les documents papiers utilisés pour décrire un dossier technique, des contraintes réglementaires et les termes du thesaurus REEF. Ces termes REEF correspondent à ceux rencontrés dans la trame d'un dossier technique et dans les guides pratiques du CSTB. La phase d'extraction et d'analyse de ces termes dits candidats a été manuelle ; cependant, un traitement automatique ou semi-automatique pourrait être envisagée (Despres and Szulman, 2006). A partir de ces termes, nous avons défini des concepts et construit une ontologie du dossier technique, OntoDT.

Dans un deuxième axe, nous avons construit un modèle du dossier technique et un système d'aide à la rédaction de dossiers techniques qui repose sur l'ontologie OntoDT. Pour décrire un produit et construire ainsi son dossier technique, l'enchaînement des formulaires à renseigner par un industriel est construit dynamiquement, en fonction des inputs de l'utilisateur et en exploitant les définitions de concepts d'OntoDT. Cela garantit que (1) seuls les formulaires nécessaires pour un produit donné sont présentés à l'utilisateur (contrairement

à la trame du dossier technique papier générique où l'utilisateur ne sait pas quelles parties doivent être renseignées pour un produit particulier) et que (2), inversement, celui-ci renseigne bien toutes les caractéristiques relatives au produit. Notre modèle et notre système reposent sur les langages et techniques du web sémantique et assurent ainsi l'interopérabilité des connaissances manipulées et des résultats obtenus. Notamment, le système produit une représentation en RDF du dossier technique.

Dans le troisième axe, nous proposons une modélisation des contraintes réglementaires basée sur les standards SBVR et SPARQL pour reformuler à la fois dans un langage contrôlé et dans un langage formel les contraintes réglementaires décrites dans les Guides Pratiques du CSTB. Nous avons construit une base de 164 règles SBVR représentant des contraintes extraites du guide pratique GP et 177 requêtes SPARQL. L'utilisation de ces deux standards nous a permis de modéliser 100% des contraintes réglementaires représentées en SBVR, extraites du et 30% en SPARQL. Il reste des limitations ou difficultés notamment pour la modélisation des règles de mise en œuvre.

Les contraintes réglementaires sont organisées en différents processus de contrôle de la conformité à la réglementation en vigueur ; à chaque composant intervenant dans un dossier technique correspond un processus soit élémentaire soit composé des processus nécessaires au contrôle des sous-composants qui interviennent dans la définition du composant. Le processus général de vérification de la conformité du dossier technique est ainsi une composition organisée des processus de contrôle des différents composants du produit sur lequel porte le dossier technique. Nous représentons les processus de contrôle de conformité en RDF, chaque processus élémentaire intègre à la fois les représentations SPARQL des contraintes élémentaires à vérifier et leurs représentations SBVR ; le processus complexe de vérification d'un dossier technique complet est construit automatiquement à partir de sa représentation en RDF : il est composé des seules contraintes réglementaires qui s'appliquent au dossier.

Nous avons développé un moteur de processus capable d'interpréter ces représentations déclaratives de processus pour ordonner et déclencher l'ensemble des requêtes SPARQL représentant les contraintes réglementaires qui doivent être vérifiées pour contrôler un dossier donné. En reposant sur ce moteur de processus, nous avons développé un outil permettant de vérifier automatiquement parmi les contraintes qui intéressent un dossier technique celles formalisées en SPARQL, en les appariant à la description RDF du dossier technique, tandis que celles représentées en SBVR uniquement sont produites à l'utilisateur comme restant à vérifier manuellement. Dans le cas où certaines contraintes réglementaires ne sont pas

vérifiées par le dossier technique, nous produisons à l'utilisateur une explication de cette non-conformité en lui présentant les caractéristiques du dossier technique et les représentations SBVR des contraintes concernées. L'outil produit un rapport de conformité du dossier technique par rapport à la réglementation qui s'applique, qui constitue une aide à la prise de décision à destination des instructeurs du CSTB en charge de produire des avis techniques sur les dossiers techniques déposés par les industriels du domaine du bâtiment.

Une validation fonctionnelle et expérimentale par les experts du CSTB nous a permis de vérifier la flexibilité de notre modèle et son aptitude à répondre de façon satisfaisante aux exigences des industriels et des instructeurs du CSTB. Nous avons alors évalué en premier lieu le niveau de formalisation des exigences réglementaires. En second lieu, les experts ont évalué nos processus de vérification, les deux prototypes d'aide à la rédaction de dossiers techniques et de vérification de la conformité des dossiers techniques à la réglementation. Ils ont conclu que ces outils répondaient à une problématique réelle d'aide à la création de documents techniques tout en diminuant la charge de travail des instructeurs du CSTB.

Dans la continuité de nos travaux, nous avons identifié quatre différentes perspectives de recherche. La première concerne la construction automatique d'ontologie et d'annotations sémantiques à partir de textes. L'inconvénient de la construction manuelle d'une base connaissance est la lenteur du processus et l'intervention de plusieurs acteurs experts. À l'heure actuelle, plusieurs approches d'extraction automatique de contraintes réglementaires ont vu le jour. Les résultats obtenus par de telles approches sont prometteurs. L'utilisation de telles méthodologies permettrait d'envisager l'automatisation (partielle) de la construction d'un vocabulaire contrôlé à partir des trames des Dossiers Techniques et de diminuer ainsi le temps et les ressources nécessaires pour la modélisation réglementaire.

Une seconde perspective est relative à la génération automatique de requêtes SPARQL à partir de règles SBVR. Nous avons transformé manuellement les antécédent et conséquent des règles opératives SBVR en requêtes SPARQL de la forme ASK, en utilisant le même vocabulaire contrôlé. L'automatisation de cette transformation pourrait être envisagée en identifiant des patrons pour transformer des fragments SBVR en requêtes SPARQL. Une fois les requêtes SPARQL créées.

Une troisième perspective est relative à la modélisation des règles de mise en œuvre pour une vérification automatique de leur respect. Les règles de mise en œuvre représentent des conditions spécifiques d'installation pour un produit. En effet, en plus des caractéristiques

structurelles et variables dimensionnelles relatives à la validation d'un produit, les guides pratiques recueillent des détails d'exécution présentant des situations possibles de mise en œuvre. Ces contraintes règlementaires permettent de vérifier des caractéristiques d'implémentation d'un produit du bâtiment (ex. « Les bois support de couverture sont fixés à la charpente, généralement sur les chevrons à l'aide de pointes en acier lisses torsadées ou crantées »). Nous n'avons pas formalisées ces règles dans le cadre de notre thèse et elles ne sont donc pas exploitables automatiquement par notre outil d'aide à la rédaction d'avis techniques. Une solution peut être envisagée combinant à la fois une modélisation des produits du bâtiment en IFC et une modélisation des contraintes réglementaires en SBVR et SPARQL. En effet, le modèle en IFC permet de représenter les caractéristiques relatives à la forme et la mise en œuvre d'un produit du bâtiment et des travaux précédents ont montré qu'il était possible de transformer le modèle IFC en RDFS (Yurchyshyna, Faron-Zucker, Mirbel, et al., 2008). Le langage contrôlé résultant pourrait être utilisé pour modéliser certaines réglementations SBVR. Toutefois, l'exploitation automatique des contraintes réglementaires de mise en œuvre nécessiterait l'aide des experts de domaine et un processus de désambiguïsation évolué.

Une quatrième perspective à envisager est relative à la généralisation de l'approche et des outils développés dans le cadre de cette thèse afin d'impacter toute la chaine de production d'information technico règlementaire. En effet, il pourrait être envisagé un outil d'aide à la rédaction de texte qui s'appuie sur des ontologies, un vocabulaire contrôlé et des règles d'écriture afin que l'expert technique soit en mesure de rédiger un texte formalisable directement dès la saisie et dont les assertions seraient alors confrontées au corpus de contraintes règlementaires existant afin d'identifier (et de corriger) les contradictions qui pourraient survenir entre les nouvelles exigences et celles déjà définies.

BIBLIOGRAPHIE

Aimé, X., 2011. Gradients de prototypicalité, mesures de similarité et de proximité sémantique : une contribution à l'Ingénierie des Ontologies.

Aussenac-Gilles, N., Biebow, B., Szulman, S., 2000. Revisiting Ontology Design: A Methodology Based on Corpus Analysis, in: Proceedings of the 12th European Workshop on Knowledge Acquisition, Modeling and Management, EKAW '00. Springer-Verlag, London, UK, UK, pp. 172–188.

Bachimont, B., 2004. Arts et sciences du numérique : ingénierie des connaissances et critique de la raison computationnelle.

Badra, F., Bendaoud, R., Bentebibel, R., Champin, P.-A., Cojan, J., Cordier, A., Desprès, S., Jean-Daubias, S., Lieber, J., Meilender, T., Mille, A., Nauer, E., Napoli, A., Toussaint, Y., 2008. TAAABLE: Text Mining, Ontology Engineering, and Hierarchical Classification for Textual Case-Based Cooking, in: ECCBR Workshops. pp. 219–228.

Bajwa, I.S., Lee, mark G., Bordbar, B., 2011. SBVR Business Rules Generation from Natural Language Specification, in: AAAI (Ed.), AAAI 2011 Spring Symposium - Artificial Intelligence for Business Intelligence (AI4BA), Spring Symposium. AAAI, pp. 2–8.

Ben Mustapha, N., A., A.M., H., B.-Z., 2006. Vers une approche de construction de composants ontologiques pour le Web sémantique, in: Actes Du Troisième Atelier Sur La Fouille De Données Complexes Dans Un Processus D'extraction Des Connaissances Des 6èmes Journées Francophones Extraction Et Gestion Des Connaissances - Troisième Atelier Sur La Fouille De Données Complexes Dans Un Processus D'extraction Des Connaissances, 6èmes Journées Francophones Extraction Et Gestion Des Connaissances. (EGC 2006. France.

Berio, G., 2003. Deliverable D3.1; Requirements analysis: initial core constructs and architecture. (UEML v. 1.0). Thematic network.

Berners-Lee, T., Connolly, D., Kagal, L., Scharf, Y., Hendler, J., 2007. N3Logic: A Logical Framework For the World Wide Web. CoRR abs/0711.1533.

Berners-Lee, T., Fielding, R., Masinter, L., 2005. Uniform Resource Identifier (URI): Generic Syntax. Request for Comments.

Boer, A., Hoekstra, R., Winkels, R., 2002. METALex: Legislation in XML, in: In Proceedings of Jurix 2002: 15th Annual International Conference on Legal Knowledge and Information Systems.

Bolioli, A., Dini, L., Mercatali, P., Romano, F., 2002. For the automated mark-up of italian legislative texts in xml, in: Legal Knowledge and Information Systems (Jurix 2002). IOS Press, pp. 21–30.

Borgo, S., Leitão, P., 2004. The Role of Foundational Ontologies in Manufacturing Domain Applications, : On the Move to Meaningful Internet Systems 2004: CoopIS, DOA, and ODBASE.

Bouaud, J., Bachimont, B., Charlet, J., Zweigenbaum, P., 1995. Methodological Principles for Structuring an "Ontology, in: Proc. of IJCAI 1995 Workshops. Morgan Kaufmann.

Bourigault, D., Aussenac-Gilles, N., 2003. Construction d'ontologies à partir de textes, in: Actes De La 10e Conférence Annuelle Sur Le Traitement Automatique Des Langues Naturelles - TALN 2003. Batz-sur-Mer, pp. 27–50.

Bouzidi, K.R., Faron-Zucker, C., Fies, B., Bourdeau, M., Le-Thanh, N., 2011. An Ontology For Modelling And Supporting The Process Of Authoring Technical Assessments, in: Submitted at Proceedings of the International Council for Building Conference (CIB). Sophia-Antipolis, France.

Bouzidi, K.R., Faron-Zucker, C., Fies, B., Corby, O., Le Thanh, N., 2012a. Modelisation de documents reglementaires dans le domaine du batiment, in: EGC. pp. 557–558.

Bouzidi, K.R., Faron-Zucker, C., Fies, B., Corby, O., Le Thanh, N., 2012b. Towards a Semantic-based Approach for Modeling Regulatory Documents in Building Industry, in: 9th European Conference on Product & Process Modelling, ECPPM 2012. Reykjavik, Islande.

Bouzidi, K.R., Faron-Zucker, C., Fies, B., Corby, O., Le-Thanh, N., 2012. Aide à la rédaction de documents réglementaires dans le domaine du bâiment, in: 23èmes Journées Francophones d'Ingénierie Des Connaissances (IC 2012), IC 2012. Paris, France, pp. 235–250.

Bouzidi, K.R., Faron-Zucker, C., Fies, B., Le-Thanh, nhan, 2011. An Ontological Approach for Modeling Technical Standards for Compliance Checking, in: RR. pp. 244–249.

Bouzidi, K.R., Faron-Zucker, C., Fies, B., Zarli, A., Le Thanh, N., 2012. Semantic Web Approach to Ease Regulation Compliance Checking in Construction Industry. Future Internet 4, 830–851.

Brickley, D., Guha, R.V. (Eds.), 2004. RDF Vocabulary Description Language 1.0: RDF Schema, W3C Recommendation. World Wide Web Consortium.

Brodie, C.A., Karat, C.-M., Karat, J., 2006. An empirical study of natural language parsing of privacy policy rules using the SPARCLE policy workbench, in: SOUPS '06: Proceedings of the Second Symposium on Usable Privacy and Security. ACM, New York, NY, USA, pp. 8–19.

Bus, N., Bruno, F., Bourdeau, M., Zarli, A., 2009. Reef sémantique, Diffusion et application des textes technico-réglementaires, Accompagnement des pouvoirs publics dans la rédaction des textes officiels. CSTB.

Camara, G., Despres, S., Djedidi, R., Lo, M., 2012. Modélisation ontologique de processus dans le domaine de la veille épidémiologique, in: Actes De La Conférence RFIA 2012. Lyon, France, pp. 978–2–9539515–2–3.

Cheng, C.P., Lau, G.T., Law, K.H., 2007. Mapping regulations to industry-specific taxonomies, in: Proceedings of the 11th International Conference on Artificial Intelligence and Law, ICAIL '07. ACM, Stanford, California, pp. 59–63.

Condamines, A., 2005. Sémantique et corpus, IC2: Série Cognition et traitement de l'information. Hermès science publ.

Corcho, O., Fernández-López, M., Gómez-Pérez, A., 2003. Methodologies, tools and languages for building ontologies: where is their meeting point? Data Knowl. Eng. 46, 41–64.

Despres, S., Szulman, S., 2006. TERMINAE method and integration process for legal ontology building, in: Proceedings of the 19th International Conference on Advances in Applied Artificial Intelligence: Industrial, Engineering and Other Applications of Applied Intelligent Systems, IEA/AIE'06. Springer-Verlag, Annecy, France, pp. 1014–1023.

Dimitrov, M., Simov, A., Momtchev, V., Konstantinov, M., 2007. WSMO Studio — A Semantic Web Services Modelling Environment for WSMO, in: Proceedings of the

4th European Conference on The Semantic Web: Research and Applications, ESWC '07. Springer-Verlag, Innsbruck, Austria, pp. 749–758.

Dubauskaitė, R., Vasilecas, O., 2009. AN OPEN ISSUES IN BUSINESS RULES – BASED INFORMATION SYSTEM DEVELOPMENT. In Innovative Infotechnologies for Science, Business and Education.

Eastman, C., Lee, J., Jeong, Y., Lee, J., 2009. Automatic rule-based checking of building designs. Automation in Construction 18, 1011–1033.

Ernst Eberg, N., Heieraas, T., Olsen, J., Eidissen, S.-H., Eriksen, S., Kristensen, K.H., Christoffersen, Ø., Lê, M.A.T., Mohus, F., 2006. Experiences in development and use of a digital Building Information Model (BIM) according to IFC standards from the building project of Tromsø University College(HITOS) after completed Full Conceptual Design Phase. The Norwegian Agency of Public Construction and Property.

Fayyad, U.M., 2007. From mining the web to inventing the new sciences underlying the internet, in: Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '07. ACM, San Jose, California, USA, pp. 2–3.

Fensel, D., Horrocks, I., Harmelen, F. van, Decker, S., Erdmann, M., Klein, M.C.A., 2000. OIL in a Nutshell, in: Proceedings of the 12th European Workshop on Knowledge Acquisition, Modeling and Management, EKAW '00. Springer-Verlag, London, UK, UK, pp. 1–16.

Fenves, S.J., 1966. Tabular decision logic for structural design. Journal of Structural Engineering 92 (ST6) 473–490.

Fenves, S.J., Garrett, J.H., Reed, K.A., 1995. Computer representations of design standards and building codes: U.S. perspective. The International Journal of Construction Information Technology 473–490.

Fenves, S.J., Wright, R.N., Stahl, F.I., Reed, K.A., 1987. Introduction to SASE: Standards Analysis, Synthesis, and Expression. Report NBSIR 87-3513 U.S, Department of Commerce, National Bureau of Standards 473–490.

Fernández-López, M., Gómez-Pérez, A., 2002. Overview and analysis of methodologies for building ontologies. Knowl. Eng. Rev. 17, 129–156.

Gandon, F., Corby, O., Faron-Zucker, C., 2012. Le web sémantique: Comment lier les données et les schémas sur le web ?, Management des systèmes d'information. Dunod.

Gómez-Pérez, A., Fernández-López, M., DE VINCENTE, A.., 1996. Towards a Method to Conceptualize Domain Ontologies, in: ECAI-96 Workshop on Ontological Engineering, ECAI-96 Workshop Proceedings.

Grosof, B.N., 2009. SILK: Higher Level Rules with Defaults and Semantic Scalability., in: Polleres, A., Swift, T. (Eds.), RR, Lecture Notes in Computer Science. Springer, pp. 24–25.

Gruber, T.R., 1993. A translation approach to portable ontology specifications. Knowl. Acquis. 5, 199–220.

Gruber, T.R., 1995. Toward principles for the design of ontologies used for knowledge sharing. Int. J. Hum.-Comput. Stud. 43, 907–928.

Grüninger, M., Atefi, K., Fox, M.S., 2000. Ontologies to Support Process Integration in Enterprise Engineering. Comput. Math. Organ. Theory 6, 381–394.

Grüninger, M., Menzel, C., 2003. The process specification language (PSL) theory and applications. AI Mag. 24, 63–74.

Guarino, N., 1995. Formal ontology, conceptual analysis and knowledge representation. Int. J. Hum.-Comput. Stud. 43, 625–640.

Guarino, N., 1997. Understanding, building and using ontologies. Int. J. Hum.-Comput. Stud. 46, 293–310.

Guissé, A., Lévy, F., Nazarenko, A., 2009. Annotation sémantique pour l'indexation de règles métiers, in: In Actes De La Conférence Internationale Sur La Terminologie Et l'Intelligence Artificielle (TIA 2009).

Guissé, A., Lévy, F., Nazarenko, A., 2011. Un moteur sémantique pour explorer des textes réglementaires, in: In Actes Des Journées Francophones d'Ingénierie Des Connaissances (IC 2011).

Guissé, A., Lévy, F., Nazarenko, A., 2012. From Regulatory Texts to BRMS: How to Guide the Acquisition of Business Rules?, in: RuleML. pp. 77–91.

Han, C.S., Kunz, J.C., Law, K.H., 1998. A client/server framework for on-line building code checking. J. Comp. in Civ. Engrg., ASCE 181–194.

Heijst, G. van, Schreiber, A.T., Wielinga, B.J., 1997. Using explicit ontologies in KBS development. Int. J. Hum.-Comput. Stud. 46, 183–292.

Hernandez, N., Mothe, J., 2006. TtoO: une méthodologie de construction d'ontologie de domaine à partir d'un thésaurus et d'un corpus de référence. IRIT, Université Paul Sabatier, Toulouse.

Horrocks, I., 2002. DAML+OIL: a Description Logic for the Semantic Web. IEEE Data Eng. Bull. 25, 4–9.

Kam, C., 2005. Dynamic Decision Breakdown Structure: Ontology, Methodology, & Framework for Information Management in Support of Decision-Enabling Tasks in the Building Industry.

Karoui, L., Aufaure, M.-A., Bennacer, N., 2007. Contextual Concept Discovery Algorithm, in: FLAIRS Conference. pp. 460–465.

Kerrigan, S., Heenan, C., Wang, H., Law, K.H., Wiederhold, G., 2003. Regulatory information management and compliance assistance, in: Proceedings of the 2003 Annual National Conference on Digital Government Research, Dg.o '03. Digital Government Society of North America, Boston, MA, pp. 1–6.

Kerrigan, S., Law, K.H., 2003. Logic-based regulation compliance-assistance, in: Proceedings of the 9th International Conference on Artificial Intelligence and Law, ICAIL '03. ACM, Scotland, United Kingdom, pp. 126–135.

Kerrigan, S., Law, K.H., 2005. Regulation-Centric, Logic-Based Conformance Assistance Framework. Journal of computing in civil engineering, ASCE 1–15.

Klyne, G., Carroll, J.J., 2004. Resource Description Framework (RDF) : Concepts and abstract syntax. W3C.

Knight, A., Dai, N., 2002. Objects and the Web. IEEE Softw. 19, 51–59.

Kopecký, J., Vitvar, T., Bournez, C., Farrell, J., 2007. SAWSDL: Semantic Annotations for WSDL and XML Schema. IEEE Internet Computing 11, 60–67.

Lau, G.T., Kerrigan, S., Law, K.H., Wiederhold, G., 2004. An e-government information architecture for regulation analysis and compliance assistance, in: Proceedings of the 6th International Conference on Electronic Commerce, ICEC '04. ACM, Delft, The Netherlands, pp. 461–470.

Lau, G.T., Law, K.H., Wiederhold, G., 2005. Legal information retrieval and application to e-rulemaking, in: Proceedings of the 10th International Conference on Artificial Intelligence and Law, ICAIL '05. ACM, Bologna, Italy, pp. 146–154.

Lau, G.T., Law, K.H., Wiederhold, G., 2006. A relatedness analysis of government regulations using domain knowledge and structural organization. Inf. Retr. 9, 657–680.

Lee, J., Gruninger, M., Jin, Y., Malone, T., Tate, A., Yost, G., Group, O.M.O.T.P.W., 1998. The Process Interchange Format and Framework. Knowl. Eng. Rev. 13, 91–120.

Lin, Y., 2008. Semantic Annotation for Process Models: Facilitating Process Knowledge Management Via Semantic Interoperability. Doktoravhandlinger ved NTNU.

Luong, P.H., 2007. Gestion de l'évolution d'un Web sémantique d'entreprise.

Malone, T.W., Crowston, K.G., Herman, G.G.A., 2003. Organizing business knowledge: the MIT process handbook, Process handbook. Massachusetts Institute of Technology.

Martínez-Fernández, J.L., González, J.C., Villena, J., Martínez, P., 2008. A Preliminary Approach to the Automatic Extraction of Business Rules from Unrestricted Text in the Banking Industry, in: NLDB. pp. 299–310.

McGuinness, D.L., Fikes, R., Rice, J., Wilder, S., 2000. The Chimaera Ontology Environment, in: Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence. AAAI Press, pp. 1123–1124.

McGuinness, D.L., Fikes, R., Stein, L.A., Hendler, J.A., 2003. DAML-ONT: An Ontology Language for the Semantic Web, in: Spinning the Semantic Web. pp. 65–93.

Mechouche, A., 2009. Représentation et partage de connaissances en neuroanatomie, et annotation sémantique des images IRM du cerveau humain.

Mika, P., Oberle, D., Gangemi, A., Sabou, M., 2004. Foundations for service ontologies: aligning OWL-S to dolce, in: WWW. pp. 563–572.

Nawari, N., 2011a. Automating Codes Conformance in Structural Domain, in: Proc. of Computing in Civil Engineering. American Society of Civil Engineers "ASCE Library.

Nawari, N.O., 2011b. A framework for automating codes conformance in structural domain. Journal of Computer and Information Technology.

Neff, M.S., Byrd, R.J., Boguraev, B.K., 2004. The Talent system: TEXTRACT architecture and data model. Nat. Lang. Eng. 10, 307–326.

Noy, N.F., Mcguinness, D.L., 2001. Ontology Development 101: A Guide to Creating Your First Ontology.

Noy, N.F., Mcguinness, D.L., 2008. Développement d'une ontologie 101 : Guide pour la création de votre première ontologie. Université de Stanford.

Noy, N.F., Musen, M.A., 2000. PROMPT: Algorithm and Tool for Automated Ontology Merging and Alignment, in: Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence. AAAI Press, pp. 450–455.

Nyman, D.J., Fenves, S.J., Wright, R.N., 1973. Restructuring study of the AISC specification, Department of Civil Engineering, University of Illinois at Urbana-Champaign. Civil Engineering Standards, Structural Research Series 393, University of Illinois at Urbana-Champaign, Urbana-Champaign, IL, USA.

Oberle, D., Lamparter, S., Grimm, S., Vrandecic, D., Staab, S., Gangemi, A., 2006. Towards ontologies for formalizing modularization and communication in large software systems. Applied Ontology 1, 163–202.

Omrane, N., Nazarenko, A., Rosina, P., Szulman, S., Westphal, C., 2011. Lexicalized ontology for a business rules management platform: an automotive use case, in: Proceedings of the 5th International Conference on Rule-based Modeling and Computing on the Semantic Web, RuleML'11. Springer-Verlag, Ft. Lauderdale, FL, USA, pp. 179–192.

Opdahl, A.L., Henderson-Sellers, B., 2002. Ontological Evaluation of the UML Using the Bunge-Wand-Weber Model. Software and System Modeling 1, 43–67.

Pan, J., Cheng, C.P., Lau, G.T., Law, K.H., 2008. Utilizing Statistical Semantic Similarity Techniques for Ontology Mapping - with Applications to AEC Standard Models, in: Proceedings of the 12th International Conference on Computing in Civil and Building Engineering (ICCCBE XII). Beijing, China.

Panetto, H., Berio, G., Benali, K., Boudjlida, N., Petit, M., 2004. A Unified Enterprise Modelling Language for enhanced interoperability of Enterprise Models, in: IFAC (Ed.), 11th IFAC INCOM2004 Symposium, April 5th-7th, Bahia, Brazil, ISBN : 0-08-044654-X. Elsevier, Bahia, Brésil, pp. 1–12.

Passant, A., 2009. Technologies du Web Sémantique pour l'Entreprise 2.0.

Pauwels, P., Deursen, D.V., Verstraeten, R., Roo, J.D., Meyer, R.D., Walle, R.V. de, Campenhout, J.V., 2011. A semantic rule checking environment for building performance checking. Automation in Construction 20, 506–518.

Reeder, R.W., Karat, C.-M., Karat, J., Brodie, C., 2007. Usability challenges in security and privacy policy-authoring interfaces, in: Proceedings of the 11th IFIP TC 13 International Conference on Human-computer Interaction - Volume Part II, INTERACT'07. Springer-Verlag, Rio de Janeiro, Brazil, pp. 141–155.

Rosemann, M., Green, P., 2002. Developing a meta model for the Bunge—Wand—Weber ontological constructs. Inf. Syst. 27, 75–91.

Satti, H.M., 2006. Integrating building code analysis within the architectural design process into the building information model.

Spreeuwenberg, S., Healy, K.A., 2009. SBVR's Approach to Controlled Natural Language, in: CNL. pp. 155–169.

Staab, S., Mädche, A., 2000. Axioms are Objects, too - Ontology Engineering beyond the Modeling of Concepts and Relations, in: Benjamins, V.R., Gomez-Perez, A., Guarino, N. (Eds.), Proceedings of the Workshop on Applications of Ontologies and Problem-solving Methods, 14th European Conference on Artificial Intelligence ECAI 2000, Berlin, Germany.

Staab, S., Studer, R., Schnurr, H.-P., Sure, Y., 2001. Knowledge Processes and Ontologies. IEEE Intelligent Systems 16, 26–34.

Studer, R., Benjamins, R.R., Fensel, D., 1998. Knowledge Engineering: Principles and Methods. Data Knowledge Engineering 25, 161–197.

Toch, E., Gal, A., Dori, D., 2005. Automatically grounding semantically-enriched conceptual models to concrete web services, in: Proceedings of the 24th International Conference on Conceptual Modeling, ER'05. Springer-Verlag, Klagenfurt, Austria, pp. 304–319.

Turk, Z., Katranuschkov, P., Scherer, R.J., Cerovs, T., 2002. The Tools and Services Integration Platform of the ISTforCE Project, in: In I. Tommelein (Ed.), Concurrent Engineering Conference. Berkeley.

Uschold, M., Gruninger, M., Uschold, M., Gruninger, M., 1996. Ontologies: Principles, methods and applications. Knowledge Engineering Review 11, 93–136.

Von Halle, B., Halle, L.G.G.C.J.Z. Barbara von, Goldberg, L., Zachman, J.A., 2006. Business Rule Revolution (ebook): Running Business the Right Way. Happy About.

Wang, N., Xu., X., 2000. A method to build ontology., in: Society., I.C. (Ed.), In HPC '00 : Proceedings of the The Fourth International Conference on High-Performance Computing in the Asia-Pacific Region. Washington, DC, USA, p. 672.

Yurchyshyna, A., 2009. (Thése) Modélisation du contrôle de conformité en construction : une approche ontologique.

Yurchyshyna, A., Faron-Zucker, C., Mirbel, I., Sall, B., Le Thanh, N., Zarli, A., 2008. Une approche ontologique pour formaliser la connaissance experte dans le modèle du contrôle de conformité en construction., in: Actes Des 19es Journées Francophones d'Ingénierie Des Connaissances (IC 2008). Nancy, France, pp. 49–60.

Yurchyshyna, A., Faron-Zucker, C., Thanh, N.L., Zarli, A., 2008. Towards an Ontology-enabled Approach for Modeling the Process of Conformity Checking in Construction, in: CAiSE Forum. pp. 21–24.