
Extension d'ontologie favorisant la catégorisation et l'indexation sémantique

Claude Moulin* — **Fathia Bettahar*** — **Jean-Paul Barthès*** — **Marco Luca Sbodio****

* *Université de Technologie de Compiègne, Heudiasyc, CNRS, Centre de recherche de Royallieu, BP 20529, 60205 Compiègne, France*

claude.Moulin@utc.fr;fathia.bettahar@utc.fr;jean-paul.barthes@utc.fr

** *Italy Innovation Center, Hewlett Packard Italiana*

24, C.so Trapani 16, 10139 Torino, Italy

marco.sbodio@hp.com

RÉSUMÉ. Dans le domaine du e-Gouvernement, les applications gèrent souvent des connaissances provenant de plusieurs sources d'information. La complexité des services administratifs et la diversité des acteurs intervenant dans les processus rendent l'accès à l'information difficile et posent plusieurs problèmes. Certains sont liés à l'accès et à la représentation de l'information. D'autres à l'interopérabilité entre les applications et les processus des services administratifs en ligne.

Dans cet article, nous présentons une ontologie décrivant des connaissances du domaine de l'aide sociale et effectivement utilisée par plusieurs modules d'une plate-forme articulée autour d'une architecture de services.

Nous présentons également deux mécanismes d'extension d'ontologie que nous avons utilisés pour nos besoins mais suffisamment génériques pour être réutilisés dans d'autres ontologies. Le premier facilite la catégorisation des éléments d'une base de connaissances. Il s'appuie sur un ensemble de règles s'articulant autour des notions de classe virtuelle, de relation virtuelle et de relation virtuelle potentielle ajoutées à l'ontologie.

Le second a pour but de faciliter l'indexation sémantique des éléments d'une base de connaissances sur le contenu de l'ontologie. Pour cela, nous étendons l'ontologie en lui ajoutant une propriété d'annotation permettant de définir les attributs de l'ontologie qui doivent servir à créer des entrées d'index.

ABSTRACT. In the e-Gouvernement domain, the applications often manage knowledge built on several information sources. The complexity of public services and the diversity of the actors involved in the processes make the information difficult to be accessed and generate several

problems. Some are linked to the access and the representation of the information. Others depend on the interoperability between applications and public service processes.

In this paper, we present an ontology describing knowledge of the social care domain and actually used by different modules of a platform based on a service oriented architecture.

We also present two mechanisms allowing to extend the ontology but generic enough to be reused in other situations. The first mechanism simplifies the categorization of knowledge base elements. It involves a set of rules defining the notions of virtual class, virtual property and virtual potential property added to an ontology.

The second one aims at facilitating the semantic indexing of knowledge base elements on the ontology content. For this, we extend the ontology with an annotation property allowing to specify the ontology attributes useful to create the index entries.

MOTS-CLÉS : Ontologie, Indexation, Catégorisation, e-Gouvernement

KEYWORDS: Ontology, Indexing, Categorization, e-Government

1. Introduction

Les informations contenues dans cet article résultent d'un travail réalisé dans le cadre du projet européen intégré TERREGOV ¹ (IST 507749) concernant le développement de services administratifs disponibles sur le Web. La plate-forme TERREGOV, destinée aux fonctionnaires territoriaux plutôt qu'aux citoyens, doit permettre aux premiers de mieux servir les derniers. Plusieurs applications pilotes ont été définies dans le secteur des services sociaux en France, Italie, Pologne et au Royaume Uni. Le problème principal concerne l'interopérabilité des services qui ne peut être réalisée sans ajout de sémantique à l'intérieur de la plate-forme et à la description des services Web. La place des ontologies dans le projet TERREGOV est essentielle et intervient à différents niveaux :

- dans la description, le repérage et le choix d'un service Web particulier ;
- dans l'indexation sémantique des documents ;
- dans l'interaction entre l'utilisateur et la plate-forme.

Le choix du projet concernant les aspects sémantiques a été de mettre en place une ontologie centralisée multi-lingue, sous la responsabilité d'un spécialiste, intégrant les différentes demandes de modification, et contrôlant les différentes versions de cette ontologie centrale. À partir de cette ontologie l'idée était d'extraire les différents éléments pour les distribuer dans les modules de la plate-forme où ils sont utilisés, tout en permettant l'ajout de concepts et d'individus en fonction des besoins.

Dans le domaine de e-Gouvernement, les applications gèrent souvent des bases de connaissances inter-connectées dans lesquelles les informations concernent les citoyens. Il est donc intéressant de pouvoir interroger ces bases de connaissances afin de sélectionner les personnes ayant des caractéristiques similaires. Par exemple, il est utile de connaître les personnes qui peuvent bénéficier de certains services sociaux en regard de leur situation.

Dans cet article, nous définissons trois méthodes permettant de classifier les éléments d'une base de connaissances. Ces méthodes sont définies à partir de règles, reprenant les caractéristiques de l'ontologie et créant de nouvelles assertions.

La complexité des processus administratifs et des documents les régissant entraîne des difficultés supplémentaires pour les fonctionnaires. L'interaction avec les systèmes informatiques doit être simplifiée et l'accès aux diverses sources d'information rendu complètement transparent. De plus, d'un point de vue technique, il est indispensable de créer des index internes qui permettent à partir de mots-clés de retrouver les éléments pertinents d'une base de connaissances. Nous présentons également un mécanisme d'extension de l'ontologie orientant la création d'index appropriés. Ils facilitent par exemple l'indexation sémantique de documents mais aussi l'analyse des dialogues entre les systèmes et les utilisateurs. La base du mécanisme s'inscrit dans l'ontologie elle-même, c'est pourquoi il faut l'envisager dès la conception de celle-ci.

1. Site du projet TERREGOV : www.terregov.eupm.net

La section 2 présente les objectifs de l'ontologie, la méthodologie retenue pour sa création et la résolution des difficultés rencontrées. La section 3 présente l'extension de l'ontologie permettant la catégorisation des éléments d'une base de connaissance. La section 4 présente le mécanisme facilitant la création d'index dans ces mêmes bases.

2. Réalisation de l'ontologie

2.1. Objectifs de l'ontologie

La solution que nous proposons à l'intégration sémantique dans un projet tel que TERREGOV, est basée sur la définition et la construction d'une ontologie servant plusieurs objectifs. Celle-ci est construite en collaboration par plusieurs groupes d'experts et suit une méthodologie déjà éprouvée (Fernandez-Lopez, 1999; Aussenac-Gilles *et al.*, 2005; Bachimont, 2000).

Nous suivons une approche à la fois descendante et ascendante dans le but de répondre aux utilisations de l'ontologie. L'approche descendante permet de conceptualiser la partie supérieure de l'ontologie et participe à la classification générale des concepts. L'approche ascendante s'appuie davantage sur la pratique des utilisateurs finals de la plate-forme et permet d'ajouter les concepts voisins des feuilles de l'arborescence. Nous nous appuyons également sur des recherches dans le domaine du e-Gouvernement (Peristeras *et al.*, 2004). Cette ontologie volumineuse et complexe permet de servir trois objectifs.

Le premier objectif est de fournir une simple terminologie de termes dont le but est de permettre l'indexation automatiquement de documents textuels constituant la base de documents d'une agence locale.

Le second objectif de l'ontologie est de faciliter la découverte de services Web publiés généralement dans des annuaires de type UDDI². Ces services sont qualifiés de sémantiques dans la mesure où ils possèdent une description faisant référence aux éléments de l'ontologie. Dans le projet le formalisme standard OWL-S³ a été choisi pour écrire ces descriptions. Les annuaires de base sont adaptés pour recevoir les descriptions sémantique des services.

La recherche d'un service nécessite à la fois ce type d'informations et l'ontologie elle-même. Des moteurs d'inférence spécifiques sont nécessaires à cette recherche et utilisent les caractéristiques principales des services comme la nature des entrées et des sorties ainsi que leur relation intrinsèque ou des informations traitées par le service (Sycara *et al.*, 2004). Une véritable dénotation des actions réalisées par le service est de plus indispensable pour une recherche efficace (Sbodio *et al.*, 2006).

2. Universal Description Discovery and Integration, <http://www.uddi.org/>.

3. OWL-S, DAML Web Service Ontology, <http://www.daml.org/services/owl-s/>.

Le troisième objectif de la construction de l'ontologie est lié à la définition des interfaces graphiques dont les utilisateurs, fonctionnaires ou citoyens, devront se servir pour proposer leurs requêtes. Ces interfaces comportent des champs statiques, mais doivent également faciliter les interactions en laissant des champs textuels d'entrée libre ou même permettre des saisies vocales. Il faut que le système puisse interpréter ces entrées, à l'aide d'un processus de dialogue entre l'utilisateur et le système. Les compléments que le système est amené parfois à demander sont générés automatiquement à partir de l'analyse des entrées des utilisateurs et des éléments contenus dans l'ontologie (Moulin *et al.*, 2006).

Pour atteindre ces objectifs il faut répondre à deux défis : (i) La simplification d'accès à l'information par les différents modules utilisant l'ontologie. Cela est rendu possible par un mécanisme d'indexation sémantique basé sur l'ontologie ; (ii) La classification des éléments des bases de connaissances grâce à des informations pouvant être ajoutées à l'ontologie.

2.2. Construction de l'ontologie

2.2.1. Méthodologie

Le problème qui nous était posé était de construire une ontologie de domaine multi-langue contenant des concepts spécifiques liés à différentes sortes d'administration : régionale, nationale et même européenne. La construction d'une ontologie peut soulever plusieurs types de problèmes. Certains sont liés à la méthodologie, d'autres au contexte même du projet, d'autres encore à la communication entre experts de domaine et experts d'ontologie. De plus, les termes utilisés pour décrire les éléments doivent être accessibles par les futurs utilisateurs (Fikes *et al.*, 1999; Uschold *et al.*, 1995; Staab *et al.*, 2001).

Le choix d'une méthodologie de construction d'ontologie doit être pris en tenant compte des coûts et des risques tout au long du processus de construction tout en assurant la qualité du résultat prévu. Le choix de la méthode dépend aussi du contexte et de la spécificité de la future ontologie. Enfin la méthode de construction doit permettre un guidage efficace et une évaluation des décisions prises concernant les choix ontologiques (Guarino *et al.*, 2002).

Nous avons proposé une méthode pour la construction de l'ontologie dont la première phase est réalisée par extraction automatique de termes candidats par un système de traitement du langage naturel. Pour ceci, nous nous sommes référés principalement à la méthode de construction d'ontologie à partir d'un corpus de textes proposée par (Aussenac-Gilles *et al.*, 2000a) et qui définit l'étape de conceptualisation d'une méthode de construction d'ontologie.

Notre méthode utilise un corpus de textes défini comme "*Une expression des connaissances à travers une langue de spécialité, avec un vocabulaire, une terminologie et une sémantique spécifiques*" et des outils terminologiques pour modéliser

le domaine. Cette méthode nécessite une validation manuelle des résultats obtenus par les experts du domaine. La seconde phase est réalisée par des experts et enrichit la conceptualisation avec de nouveaux concepts et des relations les unissant (Bettahar *et al.*, 2005).

Après cette étape de validation nous nous sommes inspirés de la méthode proposée dans le projet TOVE (Grüniger *et al.*, 1995) dans laquelle sont définis des scénarios présentant des problèmes ou des exemples qui ne sont pas résolus avec l'ontologie existante. Un scénario présente un problème pour lequel une solution doit être apportée. Ces solutions fournissent une sémantique faisant intervenir concepts et relations qui devront être ajoutés à l'ontologie.

Les scénarios peuvent être considérés comme des conditions d'expressivité présentées sous forme de questions. L'ontologie doit intégrer les nouveaux éléments sémantiques qui permettront à des systèmes basés sur elle de répondre à ces questions.

Nous avons utilisé un corpus de questions posées par les utilisateurs finals de l'application pour évaluer l'expressivité de l'ontologie construite. Cette évaluation nous a permis d'ajouter, de modifier ou de supprimer des éléments de l'ontologie et même d'ajouter des règles pour parfaire sa sémantique.

La figure 1 représente les principales étapes de la construction de l'ontologie.

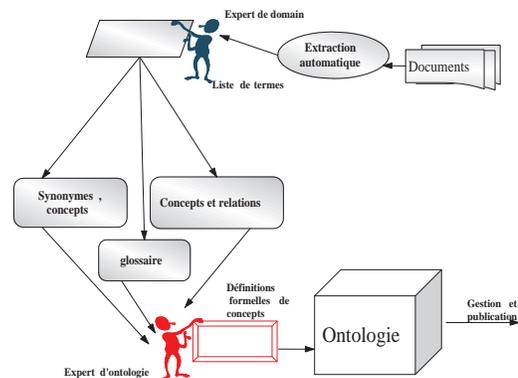


Figure 1. *Etapes de la construction de l'ontologie*

L'ontologie que nous avons construite couvre en partie le domaine de l'aide sociale. Le corpus utilisé contient des documents disponibles dans les administrations publiques d'où sont extraits les termes qui sont à l'origine de la constitution des concepts initiaux de notre ontologie (Aussenac-Gilles *et al.*, 2000b) .

Nous avons amélioré ce premier résultat par un corpus de questions qui peuvent être posées par un utilisateur confronté à l'exécution d'un service. L'ontologie a été

validée ensuite par des experts du domaine et est testée par les fonctionnaires territoriaux à travers les systèmes technologiques développés dans le projet.

2.2.2. Réalisation pratique

Corpus. Le corpus doit être constitué en fonction du domaine à traiter et les applications liées à l'ontologie afin d'atteindre les objectifs définis. Les documents ont été choisis parce qu'ils contenaient la terminologie considérée par les experts du domaine comme très pertinente pour l'ontologie. Prenons par exemple le cas du processus d'attribution du RMA (Revenu Minimal d'Activité) dans l'administration française. Pour extraire les termes pertinents associés à ce processus nous avons collecté les documents concernant chaque phase du processus : définition, scénarios, acteurs, réglementations, etc.

Nous avons ensuite utilisé l'analyseur linguistique Nomino⁴ (Dumas *et al.*, 1996) qui a permis l'extraction des termes. Cet analyseur est utilisé pour l'analyse linguistique de texte et pour détecter les unités nominales complexes afin de classer l'ensemble des termes. Il trouve pour la plupart des lemmes présents leur catégorie syntaxique (linguistique). Il donne de plus la fréquence de chaque terme dans le texte.

Le résultat de cette analyse est une liste des termes existant dans le corpus. Cette liste doit être modifiée par la suppression des termes inutiles ou l'addition de termes intéressants et à valider par des experts du domaine.

Amélioration manuelle des candidats termes. Le travail de catégorisation et de lemmatisation fourni par Nomino nous a permis d'obtenir une liste des termes pertinent existants dans le corpus. Cependant, certains termes peuvent être non significatifs pour les applications visées. C'est une discussion avec les experts du domaine qui nous a permis d'éditer manuellement la liste initiale. L'expert du domaine peut rejeter des termes erronés ou non significatifs. Il peut aussi reformuler une extraction erronée d'un terme ou ajouter des termes significatifs qui ne sont pas obtenus par l'extraction automatique.

Après cette discussion la liste résultante contient les candidats termes et les expressions considérés comme les concepts initiaux de la future ontologie. Un terme candidat doit être lié aux autres termes par des relations comme celle d'inclusion (part-of) et d'identité (is-a). La définition de ces relations permet aussi l'ajout de quelques concepts manquants.

Exemple : Certaines entités extraites représentant des noms propres (exemple : France) peuvent aider à déterminer les concepts correspondants (pays). Dans ce cas, ces entités sont considérées comme des instances de ces concepts.

Parallèlement à l'élaboration des concepts initiaux de l'ontologie, nous formulons les relations initiales. Cela nécessite l'intervention à la fois de l'expert d'ontologie

4. développé par Pierre Plante, Lucie Dumas et André Plante du laboratoire ATO de la faculté des sciences humaines de l'Université du Québec à Montréal

et l'expert du domaine. Ceci est favorisé par la liste des verbes extraits du corpus. Chaque phrase verbale peut indiquer une relation entre deux concepts de l'ontologie.

L'expert d'ontologie fournit la liste des relations génériques rencontrées habituellement dans une ontologie (comme type-de, partie-de, etc.), mais ces relations doivent être évaluées et validées. L'expert du domaine valide également les relations spécifiques (comme a-pour-fournisseur-de-service, a-pour-organisation, a-pour-municipalité, dans le domaine de l'aide sociale). Par exemple, partant de la phrase "M. François a un dossier", nous pouvons déterminer deux concepts, celui de Personne et celui de Dossier et une relation, possède, entre ces concepts ainsi que son inverse.

Questions aux utilisateurs. Les utilisateurs de notre ontologie à travers la plateforme sont des fonctionnaires territoriaux. L'ontologie doit donc répondre à leurs besoins au travers des outils manipulés. Afin de réaliser cet objectif nous avons construit un corpus de questions qui sont généralement posées aux systèmes par les fonctionnaires selon les applications sur lesquelles ils travaillent. Ces questions sont posées au travers d'une interface graphique. Pour plus de détails sur l'analyse de ces questions voir (Moulin *et al.*, 2006).

Le fonctionnaire peut poser une question en langue naturelle pour connaître la définition d'un terme administratif, ou rechercher un document ou un service. Ces questions nous permettent de connaître des scénarios d'utilisation possible et d'améliorer l'ontologie pour qu'elle puisse favoriser les réponses à ces questions.

Prenons par exemple la question suivante posée par un fonctionnaire "*Qu'est ce que le RMA ?*" Pour répondre à cette question l'ontologie doit contenir un élément *RMA* et sa définition. Le terme *RMA* apparaît dans notre modélisation comme l'acronyme de l'élément <Revenu Minimal d'Activité>, instance du concept <Programme Social>. Ainsi à partir d'une simple question nous pouvons ajouter des éléments à l'ontologie (individu, concept, propriété, littéral). Dans ce cas précis nous avons défini le concept <Programme Social> ayant comme propriété <acronyme> et ayant parmi ses instances <Revenu Minimal d'Activité>. On retient que c'est le terme *RMA* qui permet de remonter jusqu'au concept <Programme Social>. Cette remarque est aussi à l'origine du mécanisme que nous présentons dans la section 4.

2.2.3. Formalisation

Le résultat de la conceptualisation doit être présenté d'une manière formelle. Nous avons développé un formalisme suffisant pour représenter les différentes constructions de concepts dont nous avons besoin. Cependant OWL (W3C, 2004) reste le formalisme incontournable pour assurer l'interopérabilité entre les modules utilisant l'ontologie. Un parseur a également été développé qui permet de valider l'ontologie et d'automatiser un certain nombre de mécanismes lors de la transformation en OWL, comme par exemple la construction systématique des relations inverses. L'implémentation du mécanisme de création d'index décrit en section 4 est également très largement facilité par la génération de la syntaxe OWL par le parseur.

3. Catégorisation basée sur l'ontologie

Une structure de représentation d'ontologie peut avoir un haut degré d'expressivité, mais cela conduit parfois à des conclusions incontrôlables et, inversement, une restriction trop importante de la représentation permettant de mieux contrôler les raisonnements peut aboutir à l'impossibilité de représenter certains concepts (Doyle *et al.*, 1989). Il est donc nécessaire d'avoir une méthode qui permet d'améliorer l'expressivité de l'ontologie tout en assurant le contrôle du raisonnement. Parmi les solutions proposées nous avons choisi d'étendre l'ontologie en ajoutant des règles (Golbreich *et al.*, 2005) destinées à créer des assertions supplémentaires dans les bases de connaissances.

La création de règles permet de mettre à jour les catégories auxquelles peuvent appartenir les instances de la base de connaissances sans changer la modélisation de l'ontologie. Ceci est le cas lors de changement de réglementation par exemple (Ae Chun *et al.*, 2003; Baumeister *et al.*, 2007).

Nous considérons dans notre application que les bases de connaissances sont de type RDF et respectent un schéma fourni par une ontologie qui peut être étendue localement. Les ontologies et les bases de connaissances sont représentées en OWL. Nous définissons ici les notions qui permettent de classer les personnes selon certaines conditions. Ces notions permettent en particulier la catégorisation de personnes mais sont suffisamment générales pour permettre la catégorisation d'instances de tout type de concepts d'une ontologie (documents, services, etc.).

La manière d'exprimer la catégorisation d'individus dépend fortement de la modélisation de l'ontologie. Cependant, nous prenons en considération différentes manières de définir dans l'ontologie certains éléments pertinents liés à la catégorisation. Nous définissons trois méthodes permettant de catégoriser les éléments de la base de connaissances. Elles s'articulent autour des notions de classe virtuelle, de relation virtuelle et de relation virtuelle potentielle. Ces éléments sont qualifiés de virtuels parce qu'ils indiquent une manière d'organiser des instances existantes et simultanément d'interdire la création directe de nouvelles instances pour les classes considérées. Ainsi une créera des instances de personnes auxquelles sont attachées leurs date de naissance mais on ne créera jamais un adulte.

La première notion indique la création d'une nouvelle classe de l'ontologie ou de son extension ; les deux autres indiquent la création de nouvelles relations entre les concepts de l'ontologie. Toutes trois sont définies à partir de règles.

3.1. Classe virtuelle

Nous avons introduit la notion de classe virtuelle. Une telle classe est définie dans l'ontologie, mais il est impossible de créer directement une instance de cette classe. Seule l'application d'une règle permet à un moteur de l'instancier en analysant les conditions requises. Par exemple, supposons que l'ontologie contienne le

concept de <Personne>, domaine de l'attribut <age> (datatype property dans le jargon OWL). Notre objectif est de retrouver dans une base de connaissances les instances de <Personne> dont l'âge est supérieur ou égal à 18 ans. Nous définissons ainsi la catégorie <Adulte>. Les classes virtuelles seront reconnues comme telles parce qu'elles sont instances d'une classe les définissant. La classe des classes virtuelles sera définie de la façon suivante :

```
<rdfs:Class rdf:about="VirtualClass">
  <rdfs:subClassOf rdf:resource =
    "http://www.w3.org/2002/07/owl#Class"/>
  <rdfs:label xml:lang="fr">Classe Virtuelle</rdfs:label>
  <rdfs:comment xml:lang="fr">
    la class des classes virtuelles</rdfs:comment>
</rdfs:Class>
```

La classe virtuelle <Adulte> sera définie de la façon suivante :

```
<owl:Class rdf:about="Z-Adult">
  <rdfs:subClassOf rdf:resource =
    "http://www.terregov.eupm.net/...terregov.owl#Z-Person"/>
  <rdfs:label xml:lang="fr">Adulte</rdfs:label>
  <rdf:type rdf:resource="#VirtualClass"/>
  ...
</owl:Class>
```

En revanche, comme n'importe quel concept cette classe doit être définie clairement et sans ambiguïté. Elle doit posséder en OWL au moins un label et un commentaire. Elle peut être introduite dans l'ontologie dans laquelle le concept <Personne> est défini ou dans une ontologie qui l'étend. La condition d'appartenance à cette classe est produite par une règle définissant le seul moyen pour une instance de <Personne> d'appartenir à la classe <Adulte> : si une instance de <Personne> a un âge supérieur ou égal à 18 ans alors c'est une instance de <Adulte>.

Les classes virtuelles peuvent exiger la définition d'attributs désignant une valeur calculée à partir d'autres informations présentes dans la base des connaissances. En fait, c'est le cas de la classe <Adulte> parce que l'âge d'une personne n'est pas un attribut de la classe <Personne> (il est amené à changer). L'attribut réel est <date de naissance> qui est fixe. Nous appelons attribut virtuel, une propriété qui dépend d'autres propriétés et dont la valeur ne peut être calculée qu'à partir d'autres informations.

3.2. Relation virtuelle

La deuxième méthode de catégorisation consiste à utiliser une relation (Object-Property dans le jargon OWL). Supposons que l'ontologie contienne le concept

de <types d'aide> dont les citoyens peuvent bénéficier. Supposons que certains types d'aide aient été modélisés comme instances de cette classe comme <aide au logement> ou <aide financière>.

Une personne peut bénéficier d'un certain type d'aide sous certaines conditions, concernant par exemple l'âge, le salaire, le travail, etc. Les conditions d'attribution de ces aides dépendent de règles utilisant des informations présentes dans la base de connaissances relatifs aux concepts et aux propriétés de l'ontologie de référence. Il est alors possible de définir une relation virtuelle dont le domaine est la classe <Personne> et dont le co-domaine (range en anglais) est <types d'aide>.

Nous qualifions cette propriété de virtuelle pour les mêmes raisons. Elle ne peut être instanciée que par un moteur d'inférence. Les conditions de mise en relation d'une personne avec un type d'aide sont définies par une ou plusieurs règles. Chaque règle définit une façon de bénéficier d'une aide. Il est bien sûr possible de catégoriser des éléments autres que des personnes en appliquant le même processus.

Une classe particulière de l'ontologie permet de regrouper et ainsi de connaître les relations virtuelles d'une ontologie. La syntaxe en OWL est similaire à celle présentée dans la section précédente.

3.3. Relation virtuelle potentielle

La troisième méthode de catégorisation consiste à définir une relation virtuelle potentielle. Prenons l'exemple des types de services dont une personne peut bénéficier. Ces services comme <Service d'Assistance> sont modélisés comme des concepts, sous-concepts du concept <Service Social>.

Une instance d'un service peut être vue comme un contrat entre une administration et une personne. Une personne peut bénéficier d'un service si elle satisfait certaines conditions. Lorsque le cas d'une personne est étudié, le contrat n'est pas encore créé et donc aucune instance de ce service n'existe dans la base de connaissances pour représenter cette potentialité.

Une propriété <peut bénéficier> peut être définie ayant <Personne> comme domaine et <Service d'Assistance> comme co-domaine. Nous qualifions une telle propriété de relation virtuelle potentielle ; virtuelle pour les mêmes raisons que celles évoquées dans les sections précédentes et potentielle parce que une instance du co-domaine de cette propriété n'existe pas au moment de sa définition. Les conditions requises pour bénéficier d'un service sont définies par une ou plusieurs règles. Une instance anonyme du service est créée le cas échéant quand la règle permettant une telle catégorisation est examinée.

Une relation virtuelle potentielle doit être déclarée sans ambiguïté (étiquette, commentaire, domaine, co-domaine) pour qu'une application puisse l'exploiter pleinement. Elle doit être aussi déclarée comme une instance de la classe des relations virtuelles potentielles. .

```

<rdfs:Class rdf:ID="VirtualPotentialObjectProperty">
  <rdfs:comment xml:lang="fr">
    La classe des relations virtuelles potentielles
  </rdfs:comment>
  <rdfs:label xml:lang="fr">
    Relation Potentielle Virtuelle</rdfs:label>
  <rdfs:subClassOf rdf:resource =
    "http://www.w3.org/2002/07/owl#ObjectProperty"/>
</rdfs:Class>

```

3.4. Technologie

Nous avons développé une bibliothèque java qui permet d'analyser une base de connaissances en regard de ces trois types de catégorisation. Elle reçoit en entrée différents éléments. Ils consistent en une ou plusieurs ontologies, une base de connaissances et la description des règles. Chaque ontologie et la base de connaissances correspondent à un modèle OWL.

L'une des ontologies doit définir les quatre méta-classes des classes virtuelles, des attributs virtuels, des relations virtuelles et des relations virtuelles potentielles. Chaque concept ou propriété virtuel doit être déclaré comme une instance de l'une de ces quatre classes. Les conditions de l'instanciation de ces classes sont traduites par des règles au format requis par le raisonneur. Dans notre cas, les bases de connaissances utilisées par les divers modules de la plate-forme TERREGOV sont des bases de type RDF et utilisent la technologie Jena⁵.

La définition des règles Jena permet d'utiliser les opérateurs habituels mais également de définir et d'insérer d'autres opérateurs. Éventuellement, un fichier complémentaire d'opérateurs propres à l'application devra être produit. Par exemple la règle correspondant à la définition de la classe virtuelle <Adulte> est la suivante (ge pour greater or equal) :

```

[Role_Z-Adult:
  (?p rdf:type tgkb:Z-Adult)
  <-
  (?p rdf:type tg:Z-Person)
  (?p tgkb:hasAge ?y)
  ge(?y, 18) ]

```

Dans la règle précédente haAge n'est pas un attribut natif de l'ontologie mais est en fait un attribut virtuel. Il est défini à partir de la date de naissance, attribut de l'ontologie, et de la date courante grâce à l'opérateur getYears que nous avons créé à cet effet :

5. <http://jena.sourceforge.net>

```
[ Role_hasAge:
  (?p tgkb:hasAge ?y)
  <-
  (?p tg:hasBirthDate ?d)
  getYears(?d, ?y)
]
```

Dans le cas de la catégorisation à partir de propriétés virtuelles potentielles, il est à noter que l'écriture des règles Jena permet la création d'instances anonymes lorsqu'elles sont évaluées positivement. Ainsi dans la règle suivante signifiant qu'une personne handicapée peut bénéficier d'un service d'aide à domicile, `makeTemp(?x)` crée dans la base l'instance du concept qui permet de reconnaître cette potentialité.

```
[ Rule_hasHousingService:
  (?p, tgkb:hasHousingService, ?x)
  <-
  (?p rdf:type tg:Z-DisabledPerson)
  makeTemp(?x)
]
```

4. Ajout d'un mécanisme d'index à une ontologie

L'exemple du RMA (section 2.2.2) montre que toute base de connaissance a besoin d'être munie d'index permettant de retrouver des individus ou des concepts à partir de mot-clés fournis par les utilisateurs. La complexité des services et des processus administratifs dans le domaine du e-Gouvernement nécessite l'utilisation et le traitement des connaissances provenant de plusieurs sources d'information. L'indexation à base de mots-clés n'est souvent pas satisfaisante car elle ne prend pas en compte la signification des concepts auxquels pourraient se rattacher ces mot-clés. En plus il ne permet aucun raisonnement sur les structures de connaissances sous-jacentes.

La recherche d'information et l'indexation basées sur les ontologies est proposée dans la recherche académique et industrielle (Guarino *et al.*, 1999; Woods, 1997; Isaac, 2005) comme une nouvelle approche pour améliorer la découverte d'information grâce à un appui sémantique (Haav *et al.*, 2001). Cette approche est adoptée par plusieurs auteurs et en particulier dans la recherche d'information sur le Web (Song *et al.*, 2005).

Nous proposons d'ajouter des informations dans les ontologies afin d'aider à la création des index requis par certains outils. Notre approche consiste à annoter dans les ontologies les propriétés dont les valeurs doivent servir à créer des entrées d'index. Ces propriétés permettent d'associer des mots ou expressions avec des éléments de bases de connaissances. Cette approche dérive du mécanisme de point d'entrée établi dans MOSS, un environnement de connaissances développé à l'université de Compiègne (Barthès, 1994).

La propriété principale considérée pour l'indexation est l'étiquette (`rdfs:label`). Cette propriété d'annotation qui est attachée à tout élément d'une ontologie (concept, relation et individu), instance de (`owl:Thing`) doit fournir en premier les entrées de l'index.

Cependant, indexer sur cette seule propriété n'est pas suffisant. L'exemple du RMA le prouve. Plus généralement, dans le domaine de e-Gouvernement, beaucoup d'éléments sont accessibles à partir de leurs acronymes. La valeur de l'acronyme, un attribut ici de type chaîne de caractères, défini dans le concept <Programme>, doit être utilisé pour indexer les instances de ce concept. D'une manière générale, nous considérons que tout attribut est une propriété potentiellement utilisable pour l'indexation des instances des concepts où cette propriété est définie.

4.1. Principe

Tout attribut peut être utilisé pour l'indexation. Évidemment les attributs de type chaîne de caractères sont les meilleurs candidats mais les autres types de propriétés ne doivent pas être exclus.

Si une propriété a comme domaine une union des concepts, elle peut être utilisée pour indexer seulement certains de ces concepts. Le principe de notre approche est le suivant : lorsqu'un couple concept (C) et attribut (p) est considéré pour l'indexation, chaque instance *i* du concept C peut être indexé à partir de la valeur *x* de *p* pour *i*. La valeur *x* est alors l'objet du triplet (*i*, *p*, *x*) dénotant que *i* a pour attribut *x*.

4.2. Avantages

Notre proposition d'indexation permet une indépendance totale des modules compatibles avec l'ontologie. En fait chaque module est ainsi responsable de la création de l'index dont il a besoin et il trouve dans l'ontologie les éléments utiles à cette création. Il peut ajouter d'autres éléments dont il a besoin localement et appliquer toute transformation utile pour la construction de l'index.

4.3. Réalisation en OWL

Nous créons tout d'abord un propriété d'annotation (`indexing`). C'est une propriété de cardinal 1 à valeurs booléennes et de domaine `owl:Restriction` :

```
<owl:FunctionalProperty rdf:ID="indexing">
  <rdf:type rdf:resource =
    "http://www.w3.org/2002/07/owl#AnnotationProperty"/>
  <rdf:type rdf:resource =
    "http://www.w3.org/2002/07/owl#DatatypeProperty"/>
  <rdfs:range rdf:resource =
```

```

    "http://www.w3.org/2001/XMLSchema#boolean"/>
<rdfs:domain rdf:resource =
    "http://www.w3.org/2002/07/owl#Restriction"/>
<rdfs:comment xml:lang="en">propriété d'indexation</rdfs:comment>
</owl:FunctionalProperty>

```

Un concept ayant un attribut destiné à être utilisé pour l'indexation est déclaré comme une sous-classe de owl:Restriction. Cette restriction est annotée par la propriété indexing ayant vrai comme valeur booléenne associée. Par exemple, le code OWL suivant indique que chaque instance de la classe <Programme> (Z-Program est l'identifiant du concept) doit être indexé par la valeur de la propriété hasAcronym.

```

<owl:Class rdf:about="#Z-Program">
<rdfs:subClassOf>
  <owl:Restriction>
    <owl:onProperty rdf:resource="#hasAcronym"/>
    <owl:cardinality rdf:datatype =
      "&xsd;nonNegativeInteger">1</owl:cardinality>
    <indexing rdf:datatype =
      "http://www.w3.../XMLSchema#boolean">true</indexing>
  </owl:Restriction>
</rdfs:subClassOf>
</owl:Class>

```

4.4. Algorithme

Nous proposons un algorithme pour la création de l'index d'une base de connaissances à partir des propriétés annotées de l'ontologie. Nous considérons un index simple, i.e. une structure table de hachage dont les clés sont les termes littéraux extraits soit des étiquettes des éléments, soit des attributs annotés, et dont les valeurs associées sont les listes des identificateurs des éléments concernés par ces termes.

Considérons une base de connaissances et un outil (KBM) qui gère les concepts, les relations et les individus de cette base. Nous supposons qu'il accepte les requêtes SPARQL⁶. Le premier niveau de l'algorithme est :

```

Indexer les Concepts de l'ontologie
Indexer les Relations de l'ontologie
Indexer les Individus
Indexer les éléments à partir de
  la propriété d'annotation "indexing"

```

Les indexations des concepts, des relations et des individus à partir de leurs étiquettes sont similaires. L'algorithme est le suivant :

1: Obtenir les éléments (concepts, relations,

6. SPARQL Query Language for RDF, <http://www.w3.org/TR/rdf-sparql-query/>.

```

    individus) de KBM
  Pour chaque élément el
2:  Obtenir les étiquettes de el à partir de KBM
    Pour chaque étiquette
      Obtenir sa forme canonique
3:  Obtenir la liste des éléments déjà indexés sur cette étiquette
4:  Ajouter el à cette liste
      Ajouter la liste dans l'index
    fin
  fin

```

L'algorithme ci-dessus décrit les étapes de la création d'un index à partir des étiquettes des éléments. A l'étape 1, nous obtenons les éléments présents dans la base de connaissances. A l'étape 2, nous obtenons les étiquettes associées à un élément (on peut avoir plusieurs étiquettes associées à différentes langues). L'obtention de la forme canonique consiste à retirer la marque du langage qui suit généralement la chaîne. Dans l'étape 3, la liste des identificateurs peut être vide. Dans l'étape 4, nous ajoutons l'identificateur de l'élément à la liste. Cet algorithme fournit une table contenant les étiquettes comme clés et la liste des identificateurs comme valeurs. L'algorithme suivant présente la méthode d'indexation des instances à partir de la propriété d'annotation "indexing". Il requiert la construction de deux requêtes SPARQL que nous détaillons tout d'abord.

La première requête permet de rechercher les concepts et les attributs concernés par l'indexation. Dans ce cas, chaque concept apparaît comme une sous classe de `owl:Restriction`, restriction sur l'attribut concerné par l'indexation (les préfixes des espaces de noms sont omis) :

```

SELECT ?cpt ?att
WHERE {
  ?x rdf:type owl:Restriction .
  ?x owl:onProperty ?att .
  ?cpt rdfs:subClassOf ?x .
  ?x tg:indexing true }

```

Pour chaque concept et attribut, la seconde requête utilisée dans l'étape 4 de l'algorithme suivant recherche les instances de ce concept et les valeurs de l'attribut associé (cpt et att dans la requête suivante sont substitués par leurs valeurs).

```

SELECT ?ind ?v
WHERE
  {?ind rdf:type <cpt> .
  ?ind <att> ?v }

```

L'algorithme est donc :

1: Obtenir la requête 1

```

2: Obtenir les résultats de KBM
3: Pour chaque résultat (couple: concept id ; relation id)
    Obtenir la requête 2
    Obtenir les résultats de KBM
4: Pour chaque résultat (couple: id element , string value)
    Obtenir la chaîne sans le xsd datatype
    Obtenir la forme canonique
    Obtenir la liste des éléments déjà indexés sur cette chaîne
5: ajouter id element à la liste
    Ajouter la liste dans l'index
    fin
fin

```

5. Conclusion

Dans cet article nous avons présenté la méthodologie avec laquelle nous avons construit l'ontologie utilisée dans le projet TERREGOV. Cette méthodologie est directement l'expression des objectifs que cette ontologie devait satisfaire. Nous ne prétendons pas qu'elle soit complète mais elle prouve cependant le bien fondé de notre approche. Les différents modules de la plate-forme auxquels elle était destinée la charge et l'exploite : indexation sémantique de documents, description de services Web et analyse de dialogues avec les utilisateurs.

Nous avons présenté également deux mécanismes qui permettent d'utiliser encore plus largement l'ontologie. Celle-ci sert à la catégorisation des personnes dans des bases de connaissances, procédés présents dans beaucoup d'applications concernant l'aide sociale. La recherche dans le projet a fait apparaître deux approches possibles pour cette catégorisation, différant notamment dans la technologie employée mais qui s'inspirent de la méthode présentée ici.

Nous avons également détaillé un mécanisme qui permet d'indiquer à partir de l'ontologie les éléments pertinents pour l'indexation des bases de connaissances compatibles avec celle-ci. Nous avons donné l'exemple des acronymes, précisant que les valeurs correspondantes doivent permettre de remonter aux éléments réels qu'ils représentent. En fait, c'est l'expert du domaine qui indique à partir de ce mécanisme où doivent être trouvés les éléments utiles pour l'indexation écartant ainsi ce qui perturberait inutilement les investigations dans les bases de connaissances.

Ces deux mécanismes ont pu être ajoutés après le début de la création et de l'utilisation de l'ontologie sans que les modules déjà opérationnels en aient été affectés et ceci grâce à la méthodologie qui nous a conduits à ce résultat.

6. Remerciements

Le projet TERREGOV est un projet intégré co-financé par la Commission Européenne⁷ dans le Programme IST (Information Society Technologies), eGovernment unit, sous la référence IST-2002-507749. Le Consortium TERREGOV regroupe 16 partenaires, de 8 pays différents.

7. Bibliographie

- Ae Chun S., Atluri V., « Ontology-based Workflow Change Management for Flexible eGovernment Service Delivery », *The National Conference on Digital Government Research*, Boston, USA, 2003.
- Aussenac-Gilles N., Biébow B., Szulman S., « Corpus Analysis For Conceptual Modelling », *Workshop on Ontologies and Text, Knowledge Engineering and Knowledge Management, 12th International Conference EKAW*, Juan-les-pins, France, 2000a.
- Aussenac-Gilles N., Biébow B., Szulman S., « Revisiting Ontology Design : A Methodology Based on Corpus Analysis », *In 12th International Conference in Knowledge Engineering and Knowledge Management (EKAW)*, Juan-Les-Pins, France, 2000b.
- Aussenac-Gilles N., Biébow B., Szulman S., « Modélisation du domaine par une méthode fondée sur l'analyse de corpus », in l'Harmattan (ed.), *Ingénierie des connaissances*, 2005.
- Bachimont B., « Engagement sémantique et engagement ontologique. », in P. Eyrolles (ed.), *Ingénierie des connaissances : évolutions récentes et nouveaux défis*, p. chapitre 19, 2000.
- Barthès J., « Developing integrated object environments for building large knowledge-based systems, », *Int. J. Human-Computer Studies*, vol. 41, p. 33-58, 1994.
- Baumeister J., Kleemann T., Seipel D., « Towards the Verification of Ontologies with Rules », in A. Press (ed.), *the 20th International Florida Artificial Intelligence Research Society Conference (FLAIRS-2007)*, Florida, USA, p. 524-529, 2007.
- Bettahar F., Moulin C., Barthès J.-P., « Ontologies supporting eGovernment Services », *12th Portuguese Conference on Artificial Intelligence, EPIA 2005*, Covilhã, Portugal, 2005.
- Doyle J., Patil R., Two dogmas of knowledge representation : language restrictions, taxonomic classification, and the utility of representation services, rapport mit, MIT, 1989.
- Dumas L., Plante A., Plante P., Nomino 1.0, Technical report, Document centre D'ATO, UQAM, 1996.
- Fernandez-Lopez M., « Overview of Methodologies For Building Ontologies », *IJCAI'99 Workshop on Ontologies and Problem-Solving Methods*, Stockholm, p. 4/1,4/13, 1999.
- Fikes R., Farquhar A., « Distributed Repositories of Highly Expressive Reusable Ontologies », *IEEE Intelligent Systems*, 73-79, 1999.
- Golbreich C., Bierlaire O., Dameron O., Gibaud B., « Use Case : Ontology with Rules for identifying brain anatomical structures », *W3C Workshop on Rule Languages for Interoperability*, Washington, USA., 2005.

7. Le contenu de ce papier est de la seule responsabilité de ses auteurs et ne représente en aucun cas la vision de la commission européenne ou de ses services.

- Grüniger M., Fox M. S., « Methodology for the Design and Evaluation of Ontologies », *IJ-CAI'95, Workshop on Basic Ontological Issues in Knowledge Sharing*, Canada, 1995.
- Guarino N., Masolo C., Vetere G., « OntoSeek : Content-Based Access to the Web », *IEEE Intelligent Systems*, vol. 14, n° 3, p. 70-80, 1999.
- Guarino N., Welty C., « Evaluating Ontological Decisions With Ontoclean », *Communication of the ACM*, vol. 45, n° 2, p. 61-65, 2002.
- Haav H.-M., Lubi T.-L., « A Survey of Concept-based Information Retrieval Tools on the Web », in A. C. (Eds), J. Eder (eds), *Advances in Databases and Information Systems, 5th East-European Conference ADBIS*2001*, vol. 2, Vilnius "Technika", p. 29-41, 2001.
- Isaac A., Conception et utilisation d'ontologies pour l'indexation de documents, PhD thesis, Université Paris IV - Sorbonne, 2005.
- Moulin C., Bettahar F., Sbdio M., Barthès J.-P., « Semantic Support for User Interaction in an Egovernment Environment », *eGovInterop'06 "Interoperability of eGovernment Services"*, Bordeaux, France, 2006.
- Peristeras V., Tarabanis K., « Advancing the Government Enterprise Architecture -GEA : The Service Execution Object Model », *Database and Expert Systems Applications*, DEXA, Zaragoza, Spain, 2004.
- Sbdio M., Moulin C., « Denotation of Semantic Web Services Operations through OWL-S », *Workshop on Semantics for Web Services (SemWS'06) in conjunction with 4th European Conference on Web Services (ECOWS'06)*, Zurich, Switzerland, p. 17-24, 2006.
- Song J.-f., Zhang W.-m., Xiao W.-d., Li G.-h., Xu Z.-n., « Ontology-based information retrieval model for the semantic Web », *Proceedings. The 2005 IEEE International Conference e-Technology, e-Commerce and e-Service, 2005. EEE*, Hong Kong, p. 152 - 155, 2005.
- Staab S., Studer R., Schnurr H.-P., Sure Y., « Knowledge Process and Ontologies », *IEEE Intelligent Systems*, p. 26-34, 2001.
- Sycara K., Martin D., McGuinness D. L., McIlraith S., Paolucci M., « OWL-S Technology for Representing Constraints and Capabilities of Web Services », *W3C Workshop on Constraints and Capabilities for Web Services*, Oracle Conference Center, Redwood Shores, CA, USA, 2004.
- Uschold M., King M., « Towards a Methodology for Building Ontologies », *Workshop on Basic Ontological Issues in Knowledge Sharing*, 1995.
- W3C, « OWL Web Ontology Language Reference W3C Recommendation 10 Feb 2004, Dean, Schreiber, eds. », 2004.
- Woods W. A., Conceptual Indexing : a better way to organize knowledge, Technical report, Sun Microsystems Laboratories, 1997.