Towards a Semantic Interoperability in an e-Government Application

Fathia Bettahar, Claude Moulin and Jean-Paul Barthès
University of Compiègne,France
fathia.bettahar@utc.fr
claude.moulin@utc.fr

Abstract. Research issues have emerged from the rapid introduction of new technologies in government services in order to deliver efficient and cost effective services, information and knowledge through information and communication technologies. However, the complexity of government services and the diversity of actors involved in the processes make the access to the right information difficult and pose several problems. Some problems are linked to the way of presenting and accessing information. Other problems are linked to interoperability among applications and processes of eGovernment services. The objective of the European TerreGov project is to find a solution to such problems. The project focuses on the semantic requirements of governments at local, intermediate and regional levels, needed to build flexible and interoperable tools to support the change towards eGovernment services. We propose, within this project, an ontology to present knowledge and to achieve the required level of semantic interoperability. We use the ontology to describe the domain knowledge of the organization and to index the resources from which civil servants may receive information. The key point of the system is a unique and multimodal ontology used simultaneously for describing domain knowledge, for adding semantics to agency services, for indexing various documents in knowledge bases used by civil servants and finally for supporting the interaction between the users and the system. We present in this paper the challenges of using ontology in eGovernment environments, such as the lack of expressivity of the formalism chosen for interoperability in the project and the risk of inconsistency when the ontology changes. We propose our solution to such challenges and we demonstrate the use of the ontology by the module in charge of managing complex tasks in the system.

Keywords: ontology, e-government, ontology formalism, semantic interoperability

1. Introduction and motivation

The private enterprises and the public administrations need to better communicate with their customers and their partners in order to improve the effectiveness and quality of their services. Considering the complexity of the exchanges between heterogeneous systems, it is necessary to use a new model of collaboration based on a better interoperability between the information systems, independently of the type of platforms where they are implemented (Pokraev, 2007). The civil servants, in particular, will be confronted with more and more complex procedures that use various information systems. The evolution of their profession also requires better collaboration among them through forums or shared knowledge bases. This collaboration will only be possible if they share a common vocabulary and the same semantics of like knowledge being used. This semantic "interoperability" is ensured by ontologies that allow a better description of a domain and provide definitions of its concepts. We need a strong interoperability at three interaction levels: application-application, user-user and application-user. Several research works aim at improving interoperability among information systems, in particular European research projects like HI-TOUCH (Legrand 2004) in the domain tourism, and QUALEG (QUALEG 2005) and GUIDE (GUIDE 2006) in the domain e-Government.

The TERREGOV research project aims at supporting civil servants and improving the service offered by a public administration. (TerreGov is a European integrated project with sixteen partners from eight different countries working on enhancing eGovernment services). The main goal of the TerreGov project is to support civil servants in order to improve the social care service The TERREGOV research project addresses the issue of bringing the best information to local European civil servants in the Social Care domain. Until now each European region has handled the issue separately, using different concepts and business processes. Within each region the problem is complex since it calls upon many different services in a highly dynamic environment. TERREGOV aims at integrating such local solutions, not to give a final answer to the problem of Social Care, but to provide building blocks and a method for evolution. Resolving the problem at the local level implies that the various services be organized into a consistent framework, the semantic glue being provided by regional ontologies.

Reference this paper as:
Bettahar, F, Moulin, C, and Barthès, J. “Towards a Semantic Interoperability in an e-Government Application.” *Electronic Journal of e-Government* Volume 7 Issue 3 2009, (pp209 - 226), available online at www.ejeg.com

If the framework is generic enough then it can be used at a higher level to share concepts and processes, thus leading to substantial savings. The TERREGOV project lets civil servants access all information sources that may be knowledge base, domain expert or others civil servant (**Figure 1**).

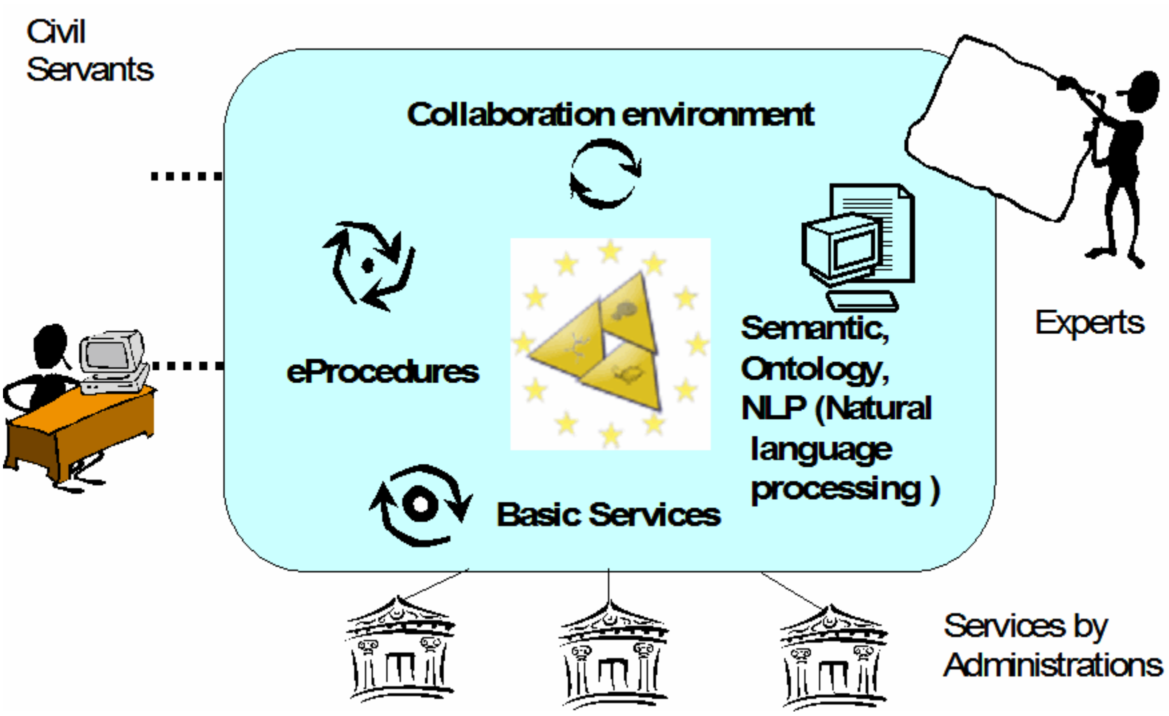

Figure 1: TERREGOV: why?

In the following we present a user case study in order to explain the real problem posed in the French public administration and gives an overview of the TERREGOV solution. The Table 1 gives the abbreviations used in this use case.

Table 1: Notations and meaning

Notation	Meaning
RMI	Minimum Integration Revenue
CAF	fund for family affaires
CCAS	Social Action Communal Centre
CMS	medico-social Centers
NLP	Natural language processing

The case of allowance request

In the case of allowance request, the principle of solidarity in France led to the RMI program (Minimum Integration Revenue), which is under the responsibility of French state but is managed by its decentralized services. The payment of the allowances is carried out by the fund for family affaires (CAF). The RMI process involves many actors (Figure 2). The processing of RMI starts by the request of a citizen who asks a social action communal centre (CCAS) for an RMI allowance. Many actors interfere to follow the citizen's case and to decide whether the citizen is eligible or not for this kind of assistance.

In this example, the diversity of actors (local government, state offices, non-profit organizations) is such that citizens have difficulty in finding a good entry point. Even the civil servants are not able to identify clearly actors' competencies and thus cannot give a global answer to the citizen's need. Moreover, the citizen must give the same information to different organizations because there is no exchange of information between them. The main goal of the TERREGOV project is to support civil servants during their daily job when delivering social care services to ordinary citizens. This is done by improving the interactions between administrations and building a knowledge base that helps civil servants to give correct and needed information in an acceptable time. To achieve this objective, we need to implement a common platform (TerreGov) that:

- Allows the implementation of government processes that invoke services from multiple administrations;
- Makes such government processes available to other administrations as eGovernment services;
- Supports civil servants involved in such eGovernment processes in getting a clear knowledge of the processes and of the services in order to act as knowledgeable front-end to citizens.

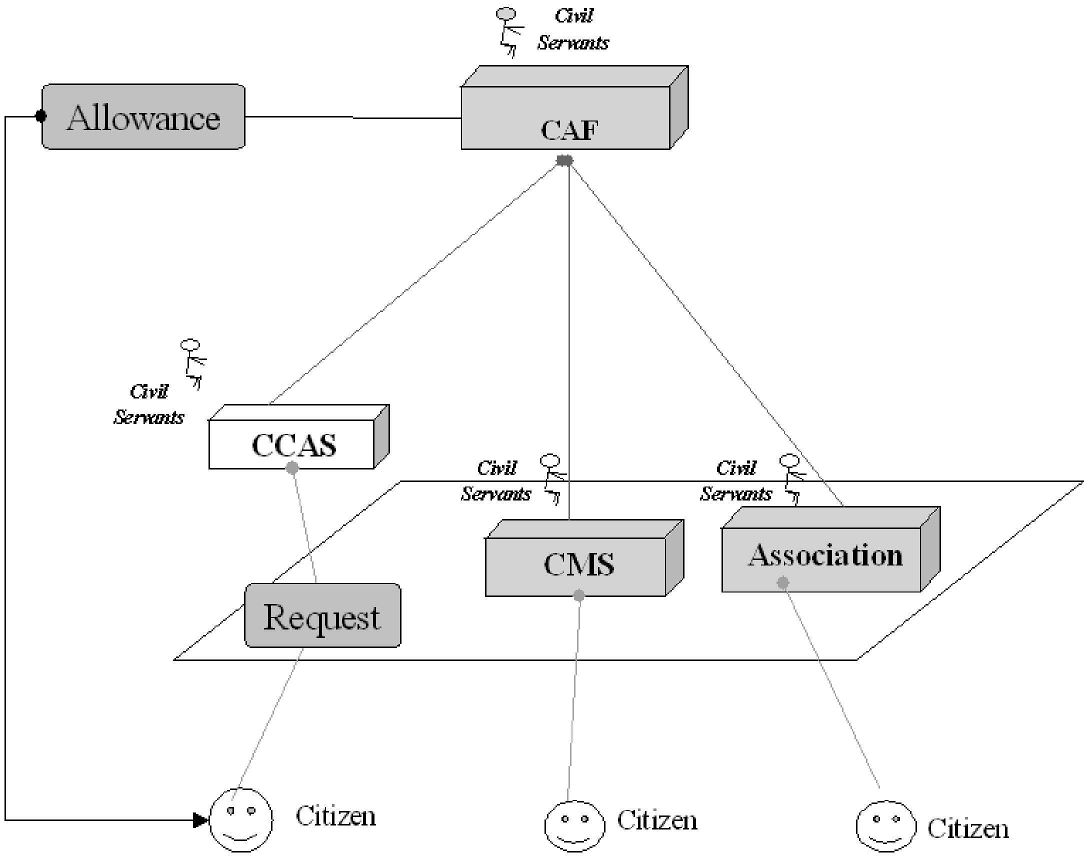

Figure 2: RMI process

The complexity of government services and the diversity of actors make the access to information complicated. In the TERREGOV project (TERREGOV 2008) we are using Web Services as an important element for application interoperability and integration (Fremantle 2002). The description of constraints and capabilities of web services should be unambiguous. This description is crucial to enable automating web service discovery, invocation, composition, monitoring, verification and simulation (Berardi 2004). To achieve this level of interoperability, it is necessary to use the vocabulary of shared ontologies (Guarino 1998). However, to efficiently use the ontology, we should first answer some challenges. The first challenge is the ontology representation. i.e. how to choose the best ontology formalism to allow a high expressive power while keeping an acceptable easiness of reasoning. The second challenge is how to build a multilingual ontology without losing the semantics specific to each language. Finally, the last challenge is how to manage the ontology in order to keep consistency after multiple changes and to ensure the access to information in the knowledge base.

2. Role of ontology

Ontology is the set of concepts, their attributes and the relationships among them that represent objects in the real world (Guarino 1995 and Gruber 1993). It provides a shared and common understanding of a domain that can be communicated across people and application systems. Ontologies are used by various communities such as knowledge engineering, natural language processing, information retrieval, etc. An ontology allows developing of automatic extraction systems, semantic and automatic natural language processes, reducing ambiguities in knowledge base interrogation. Ontologies are used by automated tools to power advanced services such as: (i) Knowledge acquisition: the ontologies allow the development of automatic extraction systems of knowledge; (ii) Automatic translation, natural language processing; (iii) Systems: Reduction of ambiguities in knowledge base interrogation, Reasoning: accessing information (information retrieval), Artificial intelligence, information organization (web, semantic web), etc. In the TERREGOV project, the area of social care that has been selected implies the collaboration of many actors, such as local government services, social workers and suppliers, which is why we can quote the following reasons for the need to develop/define a central social care ontology:

- Sharing common understanding of the structure of information among people or software agents or "intelligent" components.
- Facilitating the extraction of information and processing of documents.
- Enabling reuse of existing domain knowledge and its further extension.
- Providing a contextual framework enabling unambiguous communication of complex and detailed concepts.
- Providing a kind of semantic typing for the data distributed all over the web in order to facilitate their interrogation by users through search or query engines, and more generally their use as input or output of web services.
- Capturing a certain view of the world, supporting intentional queries regarding the content of a database and reflecting relevance of data by providing a declarative description of semantic information independent of data representation.
- Comparing objects that can be retrieved or integrated across heterogeneous repositories.

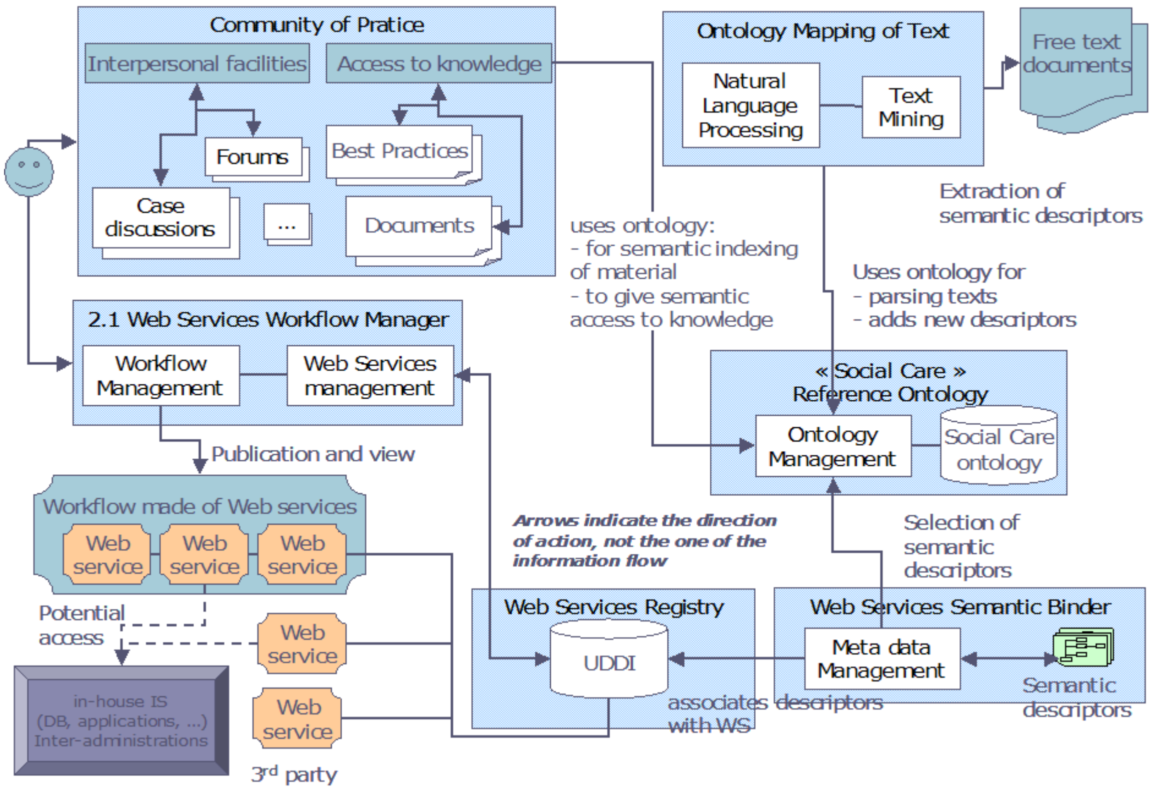

Figure 3: Role of ontology in TERREGOV project

The main objective of an ontology in the TERREGOV project is to improve the modelling of a semantic coherence for allowing the interoperability of different modules of environments dedicated to eGovernment (services; knowledge base, user interface). But there, some challenge is posed to obtain semantic interoperability. We use web services to provide interoperability among applications. The interoperability must be also in the communication between user and system. Therefore, the civil servant must have a direct access to services and must be aware of new services. The civil servant can ask about semantic aspects such as the definitions of concepts or the rules for people categorization.

For this reason we use an ontology to have words or expressions to refer to a global vocabulary coming from labels associated with ontology entities: concepts, relations, and individuals. The ontology offers also a semantic description of web services that allows the evolution of the local governments. On the other hand the ontology is used to index the Knowledge Base semantically. The civil servant in the TERREGOV platform is the point of contact between a citizen and the public administration. When he/she is asked about a problem, the civil servant can collaborate with other civil servants and experts to get information, he can access specific knowledge bases, he can discover the best web service and invoke it (Figure 3).

3. Ontology challenges in TERREGOV project

The first issue is how to build the ontology. We consider two phases to build an ontology: the first one is the research of conceptual definitions and links between concepts; the second one is the operationalization of the ontology, e.g. the way to represent the ontology using a formalism, in order to verify its consistency, to instantiate a population for this ontology, and to make inferences on concepts and instances. The great diversity of formalisms makes a good choice difficult for a given situation and the result is often only a stopgap between several solutions. We must choose a formalism adaptable to the TERREGOV context. Second, the ontology must define all the concepts and relationships used in the knowledge base from different European countries, so we must create a multilingual ontology. However, each country has specific concepts and specific regulations. Moreover, generic or common concepts may have different meanings in different countries. Thereby, building a multilingual ontology is not a simple translation of concepts from one language into another. Third, administrative processes can change according to new legal texts, to a particular situation encountered by a citizen, or according to some changes in the source of data, etc. The central ontology in TERREGOV must be updated accordingly. However, the changes in the ontology must be done without losing data and must maintain consistency. In addition, the ontology must simplify the access to knowledge base information. In the following paragraphs we detail each of three following challenges:

Challenge 1: the choice of the ontology formalism and its adequacy to the user needs.

Challenge 2: building a multilingual ontology: specifics and shared concepts

Challenge 3: managing the ontology: the responsibility of updating and using the ontology to access to information

In the next subsections, we present our solutions proposed for each of these challenges.

3.1 Ontology formalism

Operational ontologies need formalisms to be represented and storage systems to be implemented. The great diversity of formalisms makes difficult a good choice for a given situation and the result is often only a stopgap between several solutions.

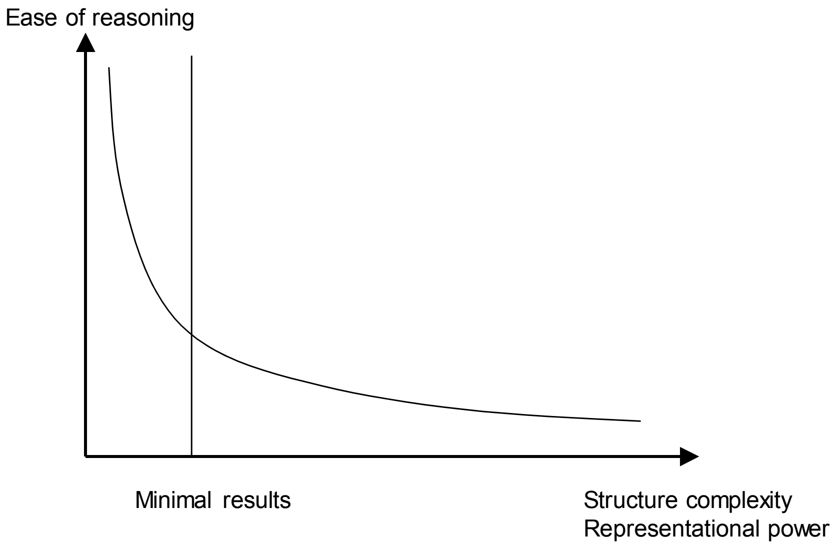

Figure 4: Ease of reasoning vs. expressivity power

Selecting a formalism for encoding an ontology is not an easy task. Indeed, the subsequent reasoning that will be done with the ontology depends on the type of selected representational structure. Some representational structures are easy to construct and to manage, but lack expressive power, thus limiting the range of possible conclusions that can be obtained, others have a high expressive power, but this leads sometimes to intractable reasoning. The correspondence between expressive power and easy of reasoning is shown Figure 4. One could be led to think that the best solution would be to choose a simple representation that allows an easy reasoning. Unfortunately, when the representation lacks expressive power, the kind of reasoning and conclusion that can be inferred is sometimes useless (see Doyle 1989). In practice, there is a minimal level of complexity that must be selected in order to be able to draw meaningful conclusions. Thus, it is necessary to examine the various possibilities as proposed by some of the formalisms for representing ontological information.

3.1.1 The choice of OWL formalism

The various formalisms available for representing an ontology derive from research in knowledge representation. There are two main approaches: (i) frame-based representation (Minsky 1975); and (ii) descriptions (Baader 2007). The resulting ontologies are deployed in two contexts: (i) locally; or (ii) in a distributed manner, i.e., web-based. Web-based formalisms (Patel-Schneider 2008) have been developed recently and are a translation of the traditional approaches to take into account the possibilities and constraints attached to distribution.

Furthermore, one should distinguish between the representation structure (usually built through object editors) and the language for accessing or transferring information. Since the TERREGOV project is dependent on the web technology it is safe to assume that the target formalism is the Ontology Web Language (OWL) (W3C 2004). However, OWL is not a unique solution and comprises three levels of representation from very simple to highly complex. Thus, the simple choice of OWL does not solve the selection problem. We need to examine what mechanisms will be useful in order to define what level of OWL must be selected, what restrictions we will introduce and what mechanisms we would like to add to OWL for our purposes. In the TERREGOV project, the formalism chosen for representing ontologies is OWL because it is the formalism proposed by the W3C and thus is a standard. However, the first modelling of ontology shows that the natural way for representing knowledge uses the OWL FULL level, even for basic structures (e.g. covering mechanism uses the union operator). In some cases, the description must be simplified. In other cases, it is possible to modify the modelling in order to use the OWL LITE level instead of the OWL FULL level. In particular, when the transitivity of the subclass relation is used, it is possible to substitute it by a transitive property on individuals . Another problem occurs when using OWL; it is often necessary to use reasoning mechanisms like "transfer through" which are not integrated in the OWL formalism. To solving these problems, it is possible to add specific rules to the inference engine to create missing facts in the knowledge base. It seems also necessary to implement procedural attachments in some way. In some cases, it is a nonsense to strictly apply the transitivity of a property. The inferred facts may become not expressive at all after two deductions. A solution is to create the property category called "attenuated transitivity". Then, it is necessary to classify the assertions in true, false, true by inference, false by inference and unknown and not apply the transitivity to the category true by inference (Lenat 1990 and Curtis 2006).

OWL limitations: Visualisation of an OWL ontology

The TERREGOV ontology contains more than a hundred concepts. It defines concepts extracted from resources and relationships amongst them. When we use OWL formalism to present this large ontology, the OWL syntax is not easy to manage it is necessary to insert features that no OWL editor (used such as Protégé, SWOOP or TopBraid Composer) currently support. Moreover the structure of concepts and properties is not presented clearly in the OWL editor. Therefore, the multilingual aspect is ill used. The following figure defines the concept (territory) with one name and some comments in four languages: English, French, Italian and Polish:

The ontology editors do not give the different labels of a concept nor its synonyms easily and the use of a OWL file directly is very difficult. Since the ontology elements are defined in a distributed manner and anywhere in this file. Moreover, some standard tools do not accept the UTF-8 format that is necessary for multilingual aspects.

```
<owl:Class rdf:ID="Z-Territory">
<rdfs:label xml:lang="en">Territory</rdfs:label>
<rdfs:label xml:lang="fr">Territoire</rdfs:label>
<rdfs:label xml:lang="it">territorio</rdfs:label>
<rdfs:label xml:lang="pl">terytorium</rdfs:label>
<rdfs:comment xml:lang="en">
A territory is a part of  a geographical entity under the rule
of nation or a part of it.</rdfs:comment>
<rdfs:comment xml:lang="fr">
Un territoire est une étendue de terre qu'offre un État,
une province, une ville, une juridiction, etc.</rdfs:comment>
<rdfs:comment xml:lang="pl">
Terytorium to czescjednostki geograficznej znajdujaca sie pod
rzadami jednego narodu lub jego czesci.</rdfs:comment>
<rdfs:comment xml:lang="it">
Un territorio è una parte di un'entità geografica governata da uno
stato o da una sua parte (Regione, Provincia ecc.)</rdfs:comment>
</owl:Class>
<owl:DatatypeProperty rdf:ID="hasName">
<rdfs:label xml:lang="en">Name</rdfs:label>
<rdfs:label xml:lang="fr">Nom</rdfs:label>
<rdfs:label xml:lang="pl">Nazwa</rdfs:label>
<rdfs:label xml:lang="it">Nome</rdfs:label>
<rdfs:domain>
<owl:Class>
<owl:unionOf rdf:parseType="Collection">
<owl:Class rdf:about="#Z-Territory"/>
<owl:Class rdf:about="#Z-Conurbation"/>
<owl:Class rdf:about="#Z-Country"/>
.......
</owl:unionOf>
</owl:Class>
</rdfs:domain>
<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#Name"/>
</owl:DatatypeProperty>
<owl:Class rdf:about="#Z-Territory">
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#hasName"/>
<owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">
1</owl:cardinality>
</owl:Restriction>
</rdfs:subClassOf> </owl:Class>
```

OWL limitations: Lack of expressivity

An ontology is used particularly to classify concepts. However, the division of a class into many subclasses may cause the definition of an instance as a subclass, which makes the reasoning more complex. In addition several regulations can be translated into rules, which do not necessarily require the creation of a subclass. For example, a person can benefit from the housing service if he fulfils some conditions. Do you have to create the subclass "PersonBenefitFromHousingService"? It is clear that the reasoning stage will use some rules related to ontology. Where can we put such rules?

In addition, there are some roles of concepts that cannot be defined by the OWL formalism. For example " adult" is a person that is aged >18 - we cannot represent this constraint with OWL. So it is necessary to use rules to define some adequate conditions of concepts

3.1.2 Simple Ontology Language (SOL)

In order to solve the problems related to using the OWL formalism, we propose some solutions that are based on a simple ontology representation. TERREGOV ontology is a central ontology managed by an ontology expert. We propose a global view of the ontology in the form of a textual file, based on a frame approach and organized in chapters and sections. The domain expert can always use the classic viewing tools for local tests and for the ontology change tests. We adopt a simple format SOL (Simple Ontology Language) that allows the management of multilingual aspects without scattering properties in the central file text. The ontology format follows in the tradition of approach frame-based and it is derived from the formalism PDM/MOSS (Barthès 1994). the following description gives the equivalent SOL format of the territory concept presented in OWL format.

```
 (defconcept
(:name :en "Territory" :fr "Territoire"
:it "territorio" :pl "terytorium")
(:att (:en "name" :fr "nom" :pl "nazwa") (:type :name)(:unique))
(:doc :en "A territory is a part of a geographical entity under
the rule of nation or a part of it."
:fr "Un territoire est une étendue de terre qu'offre un État,
une province, une ville, une juridiction, etc."
:pl "Terytorium to czesc jednostki geograficznej
znajdujaca sie pod rzadami jednego narodu lub jego czesci."
:it "Un territorio è una parte di
un'entità geografica governata da uno stato
o da una sua parte (Regione, Provincia ecc.)"))
```

The following example shows the readability of SOL format. The attribute "name" has as type: name (default: string) and each territory has a unique name (option: unique). Using SOL, the relationships are presented in the same manner. For example, the relationships of concept Authority is defined as follows:

```
 (defconcept (:name :en "Authority" :fr "Autorité" :it "Ente
Autoritario" :pl "Wladza" )
(:att (:en "name" :fr "nom" :pl "nazwa") (:type :name) (:unique))
(:rel (:en "management responsibility" :pl "odpowiedzialnosc
zarzadcza") (:to "territory") (:unique))
(:doc :en "An Authority is a legal body operating at the level of
Europe, of a country, a department, a city."
:fr "L'autorité comprend les magistrats, les hauts fonctionnaires
chargés d'une partie quelconque de l'administration publique."
:pl "Wladza to powolane zgodnie z prawem cialo funkcjonujace na
poziomie Europy, kraju, miasta czy wydzialu."))
```

In our approach, the properties attributes and relations are defined at the class level. However, a global property is composed from all local definitions of a given property, involving the merge of synonyms in different languages. Inverse properties are created automatically in SOL.

Identifiers: from SOL to OWL

We describe hereafter how to build a OWL identifier from SOL identifier. Each SOL string used to define objects (concept, attribute, relation, individual) may contain more than one name. For example for "bachelor; spinster" The OWL class identifier corresponding to the name of a SOL concept is built according to the following algorithm:

```
Search for the first English SOL name.
If this name does not exist, then
search for the first available name.
If there is no name, then an error occurs.
Otherwise, the OWL identifier is composed of:
"Z-" followed by the SOL name.
```

The OWL property identifier corresponding to a SOL attribute or relation name must be a noun and is built according to the following algorithm:

```
Search for the first English SOL name.
If this name does not exist,
then search for the first available name.
If there is no name, then an error occurs.
Otherwise, the OWL identifier is composed of:
"has" followed by the SOL name.
```

For example, the SOL English name "Town" gives the OWL property identifier "hasTown". The inverse property will be identified in OWL by "isTownOf". The OWL individual identifier corresponding to a SOL individual name is built using the same algorithm as for a class identifier.

Virtual classes and virtual properties

In the ontology paradigm, a difference is made between concepts and roles. Concepts are defining permanent properties of individuals. Roles are defining temporary properties of individuals. Person is a concept, adult is a role. Person is generally designed as a primitive concept with some characteristics like name, address or age. Every person owns such features. They define necessary conditions for an instance (individual of an ontology) to be a person. Roles are generally defined by sufficient or necessary and sufficient conditions. Main formalisms do not allow the ontological differentiation between concepts and roles and then, once implemented, roles become concepts. We propose to represent roles as virtual classes (class is the OWL word for concepts), classes which generally require built-in operators to be defined. OWL does not allow the definition of such classes yet and we have to use the rule formalism of the implementation to express these conditions. For example, if we have the concept of Person that has the property age (range is integer), we can say that a person whose age is more than 18 is an adult. This can be written as:

```
 (defvirtualclass
(:name :en "Adult")
(:is-a "person")
(:def
(?* "age" :ge 18)))
```

The tests we did with the TERREGOV ontology and some instances of Person, show that it is not necessary to define, in the TERREGOV ontology, primitive empty concepts like Z-Adult in the example. From an ontological point of view, it seems better not to introduce virtual concept definitions in the ontology itself, but to add rules with a specific identifier as Adult-rule, together with its definition as a comment in the TERREGOV rule file.

```
 [Adult-rule:
(?person rdf:type tgkb:Z-Adult)
<-
(?person rdf:type tg:Z-Person)
(?person tgkb:hasAge ?age)
ge(?age, 18) ]
```

3.2 Multilingual ontology

A multilingual ontology that represents a specific domain needs to account for the difference in language and in culture. The multilingualism and multicultural aspect is studied in the field of terminology in order to represent a multilingual and culture-specific knowledge. In (Kerremans 2003) the authors detailed the problems. They explained when they used an ontology to represent the European Value Added Tax (VAT) regulatory domain. They said the signification of a legislative term can be changed when it is translated to another language. Another complexity arises when the same language is used in different cultural settings. For example, the term "taxable event" can be implemented differently in the legislations of the different member states (Italian VAT legislation, French VAT legislation or UK VAT legislation). However, these problems depend on the domain and on the difference between terminologies in each language to represent this domain. In the TERREGOV project, we build a multilingual ontology representing the social care domain in four European countries (Italy, France, Poland, UK). In order to solve the above problems, we classified our ontology concepts in two levels: (i) the "generic level" which contains the domain concepts shared by the four languages and cultures. Examples of generic level concepts are: Person, Organization, Social Service, etc. (ii) The "specific level" which contains the domain concepts specific to each country. Examples of specific level concepts are: "French department", "Fiscal Italian Code", etc. This classification allows the representation of all concepts without losing the specificity of each culture of each country.

Currently three methods for ontology acquisition are applied in order to create a multilingual domain ontology. The first is to create a small domain specific core ontology from scratch and then improve it with interesting domain terms. The second acquisition approach takes a well-established thesaurus as a basic vocabulary reference set and converts it to an ontology representation. This approach is adopted by Simonet (Simonet 2006) they used the MeSH thesaurus as the basis of the conceptual structural of a multilingual ontology. The third approach is a combination of the first and the second methods. The first step in this approach is the creation of the core ontology. The second step is deriving the domain ontology from a thesaurus. The core ontology and the derived ontology have to be assembled into a single ontology. In (Lauser 2002) the authors used this approach to build a multilingual Biosecurity ontology. When a set of monolinguals ontologies exists to describe a specific domain, a multilingual upper-level ontology is developed for describing and identifying the relationships between specific applications and the ontologies used to describe them. For example, the MULECO ontology is designed to provide a mechanism that will allow existing ontologies in the domain of electronic commerce to identify their inter-relationships by identifying the relationships among themselves and a set of terms defined in the multilingual ontology that has been designed specially to allow finding terms using their native language.

Ontologies are the key point for semantic interoperability. From this point of view, they must be shared between several entities and different kinds of people have to acquire a good understanding of ontology structures, features, components, etc. An ontology can present different levels of complexity but is always multidimensional. In these conditions, even for experts, discovering an ontology is a hard task. Some methodology is generally required. The user of the ontology in the TERREGOV project may be no specialist and want to understand the problems related to large multilingual ontologies. The user of an ontology must scan all the ontology to access a single ontology element. Moreover, a user from a French administration does not need to consult the definition concept in English, Polish or Italian. So it is necessary that access to the ontology correspond to the level of interest of the user.

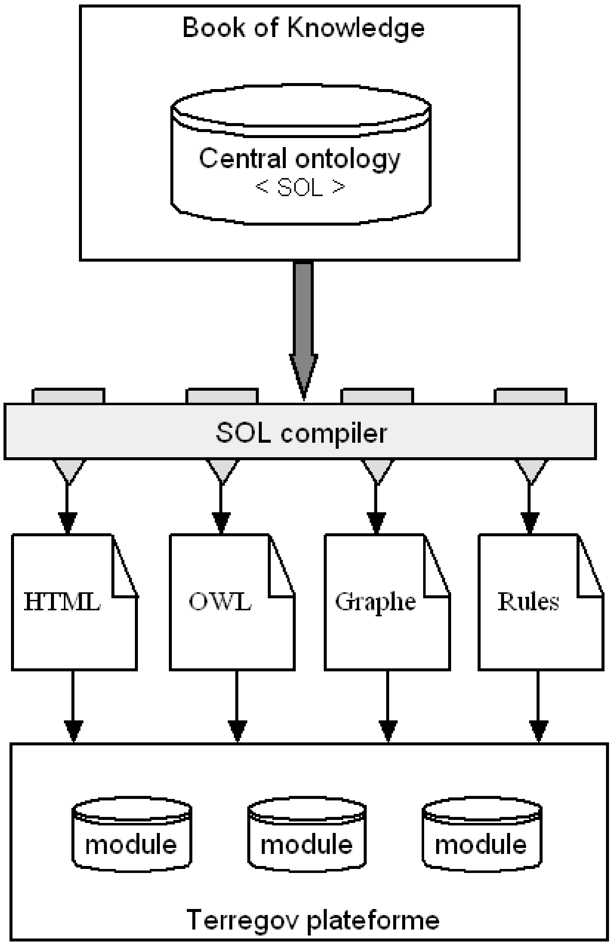

Figure 5: Building and Using TERREGOV Ontology

We proposed a solution based on the use of three kinds of tools: book of knowledge, editors, graphers.

The method consists of three phases: (i) a global approach that allows reading all the concepts, relations, and individuals of the ontology and gives a first understanding of the content; (ii) a structural approach that gives the tree of concepts and a vision of the structure of the ontology from super concepts to sub concepts, it allows the understanding of concept specialization; (iii) a local approach where the ontology can be represented by a graph. Starting from a concept, this view allows a local representation of an interesting part of the ontology. It also allows a horizontal navigation inside the ontology from one concept to other close concepts. These three levels are achieved using three categories of tools:

Book of knowledge: it is a concept developed inside the TERREGOV project. It is composed of the original SOL file and the HTML files generated by the SOL parser. The SOL file contains the whole ontology organized by chapters and sections. It can be read by any text editor. The HTML files (one for each language) are extracted from the SOL file and contain the description of all ontology elements. Links between elements allow an easy navigation

- OWL Ontology editors: they generally present a hierarchical view of ontology concepts. Main tools are Protégé, SWOOP and TopBraid Composer.
- Graphers: they present a graphical view (2d or 3D) of an ontology. They can be integrated in editors or be independent tools.

3.3 Managing ontology

Used for semantic interoperability, ontologies must be updated according to modifications proposed by domain experts. In the context of public administration, such experts may be civil servants making propositions for updating ontologies at local, regional or national level. In our approach domain experts only make requests to correct the defaults or lacks of the ontologies. Their requests must then be examined by ontology experts in charge of adapting the ontology to the specific application domain. An ontology may have different goals, and careful studies must be done before any adaptation. The third actor is the entity in charge of the ontology lifecycle that has to decide when new versions must be distributed. The main questions to answer when updating the ontology are: how is the ontology to be used and locally modified; how is it upgraded; how is a new version distributed and how does it replace the previous one. The main issue raised by such questions is how to manage the ontology changes and who is responsible for the ontology changes. In order to update the ontology, a domain expert fills in a Request For Change (RFC) form proposing modifications to the current ontology. The updating of ontology is done by the collaboration of two actors: the ontology expert who is in charge of modifying the ontology implementation and the domain expert who asks for the modification of ontology elements. Once the RFC forms are filled, the ontology expert studies the requests and tries to find an implementation compatible with all the objectives of the ontology. The ontology consistency must also be insured. Civil servants in different public administrations can send requests for change. But they do not have the rights to modify the ontology directly even if they know how to do it. The modification of the ontology asked by domain experts may result in: (i) Extending the ontology by adding concepts or relations. In fact, the modification or the creation of an administrative process requires in some cases the evolution of the semantics. In the RFC forms the civil servant gives all the characteristics of new elements, which are necessary to run the administrative process. (ii) Modifying of some concepts or relations, for example modifying the domain and range of a relationship. The civil servant can also ask the ontology expert to delete some features that are unused by the administrative process.(ii) Cancelling of a concept or a relation, this action must be avoided. Since deleting a concept may have many consequences, in particular eliminating all its relations with other concepts. This type of action changes the conceptual model of the ontology. The following figure shows the possible actions concerning a concept.

The ontology expert receives the request from the domain expert. In some cases, the ontology expert needs to clarify the request or ask for missing information and then a dialog is launched. We call RFM a request for missing information. For example, in the first version of the ontology we have defined the concepts and relations corresponding to the RMI process. After an analysis of the ontology elements based on documents given by the domain expert in the French pilot and discussions with the domain expert we propose some changes in the ontology. The changes allow improving of the execution of the RMI process. The changes consist of:

- Adding the concept "civil servant": the domain expert explains that a civil servant has some specific properties such as the properties rights, links services, etc.
- Adding the attribute "Title" to the concept of person: the domain expert explains that the information about title (Mr., Mme., Mle.) must be announced for each citizen.
- Adding the relation "medical doctor": according to French regulations each person must have a medical doctor.

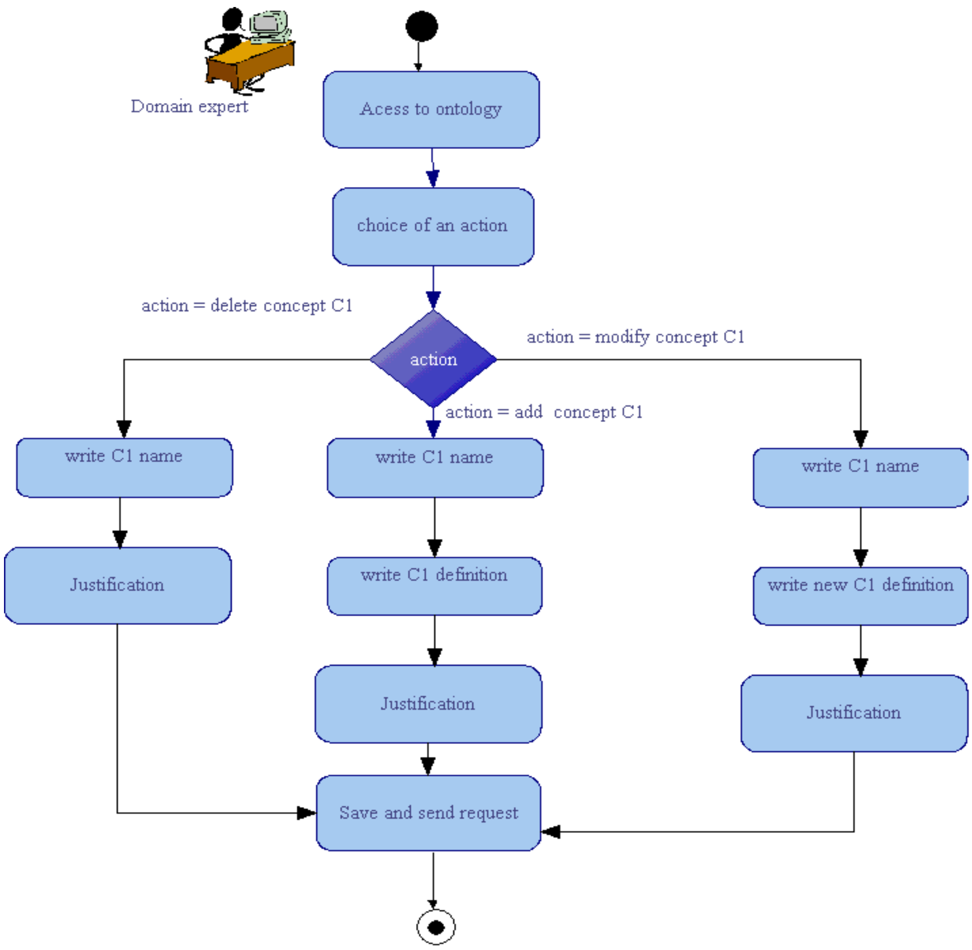

Figure 6: Request for change: actions concerning a concept

4. Improve ontology by an index mechanism

The complexity of eGovernment services and administrative processes requires handling knowledge from multiple information sources. Keyword indexing is often not a satisfactory option because it lacks precision and does not take into account the denotation of information. Moreover, it does not allow any reasoning on knowledge structures. Information retrieval and indexing based on ontology is proposed in academic and industrial research environments (Guarino 1999, Woods 1997) as a new approach to improve information discovery by adding semantic support (Haav 2001). This approach is adopted by several authors and particularly for information retrieval from the Web. Internet search engines like Google or Altavista use a central database to index information and a simple keyword based requester to reach information. To improve the semantics of search, two major approaches are proposed by researchers. The first approach concerns an annotation technique based on the use of ontologies (Guarino 1999, Sycara 2004). The annotations are used to retrieve documents. This approach is dedicated to request/answer systems like KAON (Bozsak 2002). The second approach is an information retrieval technique based on the use of domain ontologies like the work of Desmontils in (Desmontils 2003) or the Work of Rossitza (Rossitza 2007) that presents a concept indexing algorithm, which adds ontology information to words and phrases and allows full text to be searched, browsed and analyzed at different levels of abstraction. This algorithm uses a general purpose ontology and an ontologically tagged corpus.

These are dedicated for retrieving raw documents. In both cases, tools are required to create indexes based on the vocabulary occurring in ontologies. Annotation properties can be used for that. We propose to add information inside the ontology in order to support the creation of indexes for such tools. Our approach consists in annotating the properties whose values can be used to create index entries. This approach is derived from the entry point mechanism found in MOSS. In this section we describe the features we included in an ontology, the semantics of which can be used to build the index of a knowledge base. We first give the characteristics in the OWL formalism then we present the equivalent features in a simpler formalism that we have developed. The parser generating the OWL file is in charge of the creation of the required OWL structures.

In this section, all considerations concern the OWL formalism. The name of a concept is not defined in OWL; if it were defined, it would correspond to the identifier. The main property to consider for indexing is the label (introduced by rdfs:label). This property is attached to any owl:Thing and thus we consider that each instance of owl:Thing (concept, relation and individual) can be indexed from the value of this annotation property. Using only the label property is not generally sufficient. For example, in the domain of e-government, many elements are accessed from their acronym. In French, the acronym "RMA" can be used to access the social program the label of which is "Revenu Minimal d'Activité". The values of an acronym, a string data type property defined on the concept of Program, would be useful to index the instances of Program. More generally, we consider that each data type property may be a candidate property for indexing individual concepts.

Principle

Any data type property, can be used for indexing. Obviously string properties are the best candidates but other types of properties can also be used. If a property has for its domain a union of concepts, we also consider that this property can be used for indexing for only some of these concepts. The principle is the following: when a couple (concept (C), data type Property (p))is considered for indexing, each instance I of the concept C can be indexed from the value of p for i.

Advantages

Our indexing proposal allows a total independence of modules compliant with an ontology. Indeed, each module is responsible for the index creation it needs, and finds in the ontology all the elements useful for this creation. It can also add any element that it requires locally. It can apply any transformation for building the strings or other elements required for the entry keys, from the labels and values found in the ontology or knowledge base.

OWL Implementation

In the OWL syntax, an annotation property is first created. It is also a functional datatype property range of which is xsd:boolean. It is considered as a restriction on a datatype property for some concepts and its domain is the class 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>
```

A concept having a datatype property, intended to be used for indexing, is declared as a subclass of an owl:Restriction. This restriction is annotated by the indexing property with true as the Boolean value. For example, for the following OWL code indicates that each instance of the class <Program> must be indexed by the value of the property 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>
```

Index in SOL

In SOL indexing, a datatype property is simply declared with the (:index) feature. In the following example, the attribute "acronym" is intended to be used for indexing the instances of the Program concept. If the attribute "acronym" were defined in another concept it could be used for indexing or not, according to the presence of (:index) in its declaration.

```
 (defconcept (:name :en "Program" :fr "Programme") (:att (:en
"acronym" :fr "acronyme") (:unique) (:index)) (:att (:en
"documentation" :fr "documentation")) ...)
```

Algorithm

We propose an algorithm for creating a knowledge base index from the ontology features. We consider a simple index, i.e. a hash table structure where keys are words or expressions found in element labels and as values of some data type properties and where each associated values is a list of element identifiers. We consider a knowledge base manager (KBM) for a system that keeps track of all concepts, relations and individuals of the knowledge base and supports SPARQL (SPARQL 2005) queries. The top level of the algorithm is:

```
Indexing All Concepts, All Relations, All Individuals
Indexing elements from "indexing" Annotation property
```

The indexing of concepts, relations and individuals from their labels are similar. The algorithm is the following:

```
1: get the elements (concepts, relations, individuals) from KBM
for each element el
2: get the element labels from KBM
for each label
get the canonical form of the label
3: get the list of element ids already indexed on this label
4: add the el id to the list
put the entry (canonic_label,list) into the index;
endfor
endfor
```

The previous algorithm gives the steps to creating an index from the labels of the ontology elements. In Step 1 we get the elements present in the knowledge base. In step 2, we get the labels attached to an element. More than one label may exist, e.g. in different languages. In step 3, the list of ids may be empty. In step 4 we add the element id to the list. This algorithm yields a table containing labels as keys and list of ids as values. The following algorithm shows how to index concepts and individuals from the annotation property "indexing":

```
1: set the first query string
(for searching concepts and relations concerned by the indexing)
2: get the query results from KBM 3: for each result (couple:
concept id ; relation id)
set the second query from concept and relation ids
(for searching the instances of the concepts and the values
associated through the relation)
get the query results from KBM
4: for each result (couple: element id, string value)
get the string value
get the canonical value as a string
get the list of element ids already indexed on this label
5: add the el id to the list
put the entry (canonic_label,list) into the index;
endfor
endfor
```

The first request (written here in SPARQL) allows searching the concepts and attributes concerned by the indexing. In this case, each concept appears as a subclass of a restriction on the relations also concerned by indexing:

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

For each concept and attribute the second query used in step 3 searches instances of this concept and values associated to the attribute.

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

Application

We consider a platform dedicated to e-Government services where a global ontology ensures the homogeneity of the semantics. In this platform several modules use the ontology: the semantic registry of web services contains the semantic descriptions (written in OWL-S) of services written thanks to the central ontology, the document base where documents are indexed on the concepts of the ontology. We plugged a dialog system helping a user to find concept definition, documents or services from free text queries. See (Moulin 2006) for more details about the dialog system. Our dialog system receives natural language queries from civil servants, analyzes them and extracts the keywords or expressions that it contains. According to this analysis the module that can answer the question is selected. The dialog system creates an index based on the process described above. The words extracted from queries, after a normalization process, allow finding the identifiers of the ontological elements related to the question (Figure 7). For example, from the identifiers, the document base can deliver the documents indexed on the corresponding concepts.

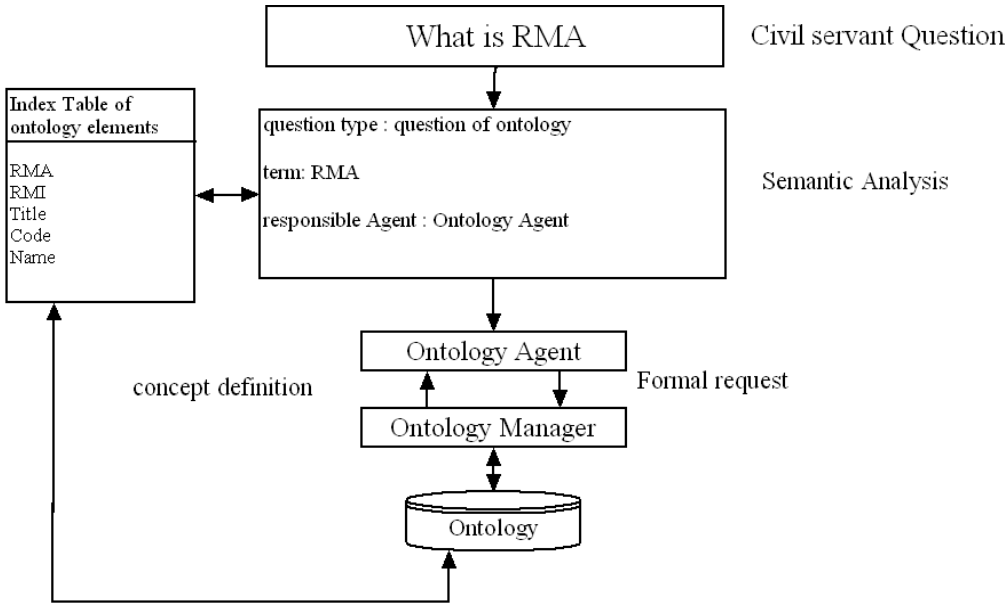

Figure 7: Using index to search a term

5. Conclusion

In this paper we present a multilingual ontology to improve the modelling of a semantic coherence allowing the interoperability of different modules of environments dedicated to eGovernment platforms. To improve the expressivity of our ontology, we propose to distinguish among important and secondary elements in ontology, and we introduce a simplified syntax, SOL, allowing a developing multilingual centralized ontology.

In this paper we also study the managing of ontology according responsibility in the project. The ontology expert can update an ontology according to modifications proposed by domain experts. Moreover in this paper, we present an indexing process enrichment, which makes it possible to take into account elements which do not belong to the ontology. Our approach consists of a creation of an index using ontology in a dialog system. Each module of this system is responsible for creation of its own index.

6. Acknowledgements

The TerreGov project is an integrated project cofunded by the European Commission[1] under the IST (Information Society Technologies) Programme, eGovernment unit, under the reference IST-2002-507749.

The TerreGov Consortium includes: AIRIAL Conseil (F), Université de Technologie de Compiègne (F), GFI Informatica (E), Hewlett Packard Italiana (I), Ratio Consulta (I), Oxford Computer Consultants (UK), Technion – Israel Institute of Technology (IL), HEC School of Management (F), Associazione Impresa Politecnico (I), Business Flow Consulting (F), European Regional Information Society Association (B), Stowarzyszenie Miasta w Internecie (PL), Aquitaine Europe Communication (F), Regione del Veneto (I), R&S Info (I), Epektasis (G).

References

Baader, F. Ganter, B. Sattler, U. Sertkaya, B. (2007), Completing Description Logic Knowledge Bases using Formal Concept Analysis, in: Proceedings of the Twentieth International Joint Conference on Artificial Intelligence (IJCAI-07), AAAI Press.

Barthès, J-P. (1994), Developing integrated object environments for building large knowledge-based systems, Int. J. Human-Computer Studies, 41:33–58, 1994.

Berardi, D Gruninger, M. Hull, R. and McIlraith, S. (2004), Towards a first-order ontology for semantic web services. In the W3C International Workshop on Constraints and Capabilities for Web Services (CCWS04).

Bozsak, E. Ehrig, M. Handschuh, S. Hotho, A. (2002), Towards a large scale semantic web. In DEXA 2002, Aix en Provence, France.

Curtis, J. Cabral J., Baxter, D. (2006), On the Application of the Cyc Ontology to Word Sense Disambiguation, *Proceedings of the Nineteenth International Florida Artificial Intelligence Research Society Conference*, Melbourne Beach, Florida, Published by The AAAI Press.

Desmontils, E Jacquin, . C. and Simon L. (2003), Ontology enrichment and indexing process, ingénierie des connaissances. Technical report, Institut de Recherche en Informatique de Nantes 2.

Doyle, J. and Patil, R. (1989), Two dogmas of knowledge representation: language restrictions, taxonomic classification, and the utility of representation services, rapport mit, MIT.

Fremantle Weerawarana, S. and Khalaf, R. (2002), Enterprise services. Communicationsof the ACM, 45(10):77–82.

Guarino, N. (1998), Formal ontology and information systems. Formal Ontology in Information Systems, IOS Press, Amsterdam, pp. 3–15.

Guarino, N. Giaretta, and P. (1995), Ontologies and knowledge bases: Towards a terminological clarification. In Towards Very Large Knowledge Bases: Knowledge Building and Knowledge Sharing, pages 25–32. Amsterdam, in n. mars (ed.) ios press edition.

Guarino, N. Masolo, C. and Vetere, G. (1999), Ontoseek: Content-based access to the web. IEEE Intelligent Systems, 14(3):70–80,

Haav, H-M. and Lubi, T.-L. (2001), A survey of concept-based information retrieval tools on the web. In A. Caplinkas (Eds) and J. Eder, editors, Advances in Databases and Information Systems, 5th East-European Conference ADBIS*2001, volume 2, pages 29–41, Vilnius "Technika".

Kerremans, K. Temmerman R., and Tummers, J. (2003), Respresenting multilingual and culture-specific knowledge in a vat regulatory ontology: support from the termontography approach. In Robert Meersman Tari and Zahir, editors, OTM Workshops, pages 662–674. Tübingen: Springer Verlag.

Lauser, B. Wildemann, T. Poulos, A. Fisseha, F. Keizer, J.and Katz, S. (2002), A comprehensive framework for building multilingual domain ontologies: creating a prototype biosecurity ontology. In Dublin Core and Metadata for e-Communities, pages 113–123.

Legrand, B. (2004), Semantic web methodologies and tools for intra-european sustainable tourism, In JITT, 2004.

Lenat, D. and Guha, R. V. (1990), Building large knowledge-based Systems: Representation and Inference in the Cyc project. Addison-wesley edition.

Minsky M. (1975), A Framework for Representing Knowledge. The psychology of computer vision, p. winston (ed.), mcgraw-hill edition, 1975.

Moulin, C., Bettahar, F. Sbodio, S. and Barthès, J-P. (2006), Semantic support for user interaction in an egovernment environment. In eGovInterop'06 "Interoperability of eGovernment Services", Bordeaux, France, 2006.

Patel-Schneider P. and Horrocks I. (2008), Owl 1.1 web ontology language overview, [http ://www.webont.org/owl/1.1/overview.html]

[1] The content of this paper is the sole responsibility of the authors and in no way represents the views of the European Commission or its services.

Pokraev, S. Quatel, D. Steen M. and Reichert, M. (2007), Semantic Service Modeling: Enabling System Interoperability, Enterprise Interoprability: New challenges and approaches, Book establishes the proceeding of the I-ESA-06. Springer-Verlag London Limited.

Rossitza M. Setchi, and Tang, Q.(2007), Concept Indexing using Ontology and Supervised Machine Learning, International Journal of Intelligent Systems and Technologies Volume 2.

Simonet, M. Patriarche, R. Bernhard, D. Diallo, G. Ferriol, S. and Palmer, P. (2006), Multilingual ontology enrichment for semantic annotation and retrieval of medical information. In MEDNET2006, Toronto – Canada.

Sycara, K. Martin, D. McGuinness, D.L. McIlraith, S.and Paolucci, M. (2004), Owl-s technology for representing constraints and capabilities of web services. In W3C Workshop on Constraints and Capabilities for Web Services, Oracle Conference Center, Redwood Shores, CA, USA.

Thomas R. Gruber (1993), Toward principles for the design of ontologies used for knowledge sharing, International Journal Human-Computer Studies Vol. 43, Issues 5-6, p.907-928

W3C (2004). OWL web ontology language semantics and abstract syntax w3c recommendation 10 feb 2004. patel-schneider, hayes, horrocks, eds.

Woods, W.A. (1997), Conceptual indexing: a better way to organize knowledge. Technical report, Sun Microsystems Laboratories.

GUIDE (2006), “Web site of GUIDE project”, [istrg.som.surrey.ac.uk/projects/guide/]

QUALEG (2005) , “Web site of QUALEG project”, [www.qualeg.eupm.net]

SPARQL (2005). Sparql, w3c recommendation, 23 November 2005, [http://www.w3.org/tr/2005/wd-rdf-sparql-query-20051123/].

TERREGOV (2008), Impact of e-Government on Territorial Government Service, IST-2002-507749. [http://www.terregov.eupm.net]