Ontology-Based Retrieval of Spatially Related Objects for Location Based Services

Hele-Mai Haav[1], Aivi Kaljuvee[4], Martin Luts[2,3], and Toivo Vajakas[4]

[1] Institute of Cybernetics at Tallinn University of Technology,
Akadeemia 21, 12618 Tallinn, Estonia
helemai@cs.ioc.ee
[2] ELIKO Competence Centre in Electronics-, Info- and Communication Technologies,
Akadeemia 23A, 12618 Tallinn, Estonia
[3] Department of Informatics, Tallinn University of Technology,
Ehitajate tee 5, 19086 Tallinn, Estonia
martin.luts@eesti.ee
[4] Regio Ltd, Riia 24, 51010 Tartu, Estonia
aivi.kaljuvee@reach-u.com, toivo@regio.ee

Abstract. Advanced Location Based Service (LBS) applications have to integrate information stored in GIS, information about users' preferences (profile) as well as contextual information and information about application itself. Ontology engineering provides methods to semantically integrate several data sources. We propose an ontology-driven LBS development framework: the paper describes the architecture of ontologies and their usage for retrieval of spatially related objects relevant to the user. Our main contribution is to enable personalised ontology driven LBS by providing a novel approach for defining personalised semantic spatial relationships by means of ontologies. The approach is illustrated by an industrial case study.

Keywords: Ontology-based information retrieval, domain ontology modelling, mobile IS, personalised LBS, semantic LBS.

1 Introduction

The International Open Geospatial Consortium [1] defines Location Based Service (LBS) as a wireless-IP service that uses geographic information to serve a mobile user or any application service that exploits the position of a mobile terminal. There is a broad spectrum of LBSs like friend-finder, weather information and address geo-coding and reverse geo-coding services that are widely provided (see for example [2, 3]). Business applications like people/vehicle tracking, traffic monitoring or fleet management are some of the examples of operational LBS. Technologically LBSs are based on the combination of Geographic Information Systems (GIS), communication and the Internet technologies.

The need for providing semantic LBS has been recognized in mobile GIS and LBS research communities in recent years [4, 5]. More advanced LBS applications need integration of information collected in GIS and information about users' preferences

R. Meersman, T. Dillon, P. Herrero (Eds.): OTM 2009, Part II, LNCS 5871, pp. 1010–1024, 2009.

as well as the application itself and its context. On the other hand, ontology engineering [6] is considered as providing solutions to semantically integrate several data sources. There are many challenges in developing semantic LBS the most important of which are data integration and the providing of dynamic contents.

In order to solve some of the problems related to semantic LBS, we propose an architectural solution – Smart Semantic Space FrameWork, abbreviated as S3FW. The concept would resolve a wide range of ontology-driven LBS for mobile user actions like locating, navigating, searching and identifying. The S3FW is component based, allowing us to implement it incrementally on use-cases demand, but still having an extendable, visionary approach. As a proof of the concept implementation of S3FW is done within the framework of the project "Smart ontology-based spatial information retrieval and integration" of ELIKO Competence Centre in Electronics-, Info- and Communication Technologies [7].

We have designed S3FW as an additional, "semantic" module to be used as an add-on to proprietary GIS products. A proprietary GIS solution needs to be modified to have additional hooks in its business layers to S3FW, which would offer ontology-based services like the retrieval of spatial objects, multilingual verbalization of spatial objects and their relationships, etc.

The architecture of S3FW has the following main components:

- Subsystem of ontologies for formalising explicit knowledge about the domain of interest, for example: ontologies of Point Of Interests (POI), user profile ontologies, ontology for spatial relationships;
- Subsystem of semantic tools (e.g. reasoners);
- Subsystem of Natural Language Processing (NLP) that is to be used to produce natural-language-alike outcomes of LBS.

The aim of the paper is not to provide details of the S3FW due to its limited space. Instead, this paper concentrates on the subsystem of ontologies derived from S3FW.

We will present architecture of ontologies as the basis for development of different semantic LBS. We will show how these ontologies can be used for qualitative reasoning about spatially related objects in order to retrieve objects relevant to the user's interests. In contrast to works on modelling qualitative spatial relationships (see section 6), we provide a novel contribution that makes it possible to reason on *semantic* spatial relationships that are not purely topological relationships but are enriched with meaning that domain experts, application developers and end users will expect.

As a proof of the concept, we developed reverse geo-coding service. When traditional reverse geo-coding provides address or place according to given location [e.g. see 2], we *extended* the traditional approach by providing also search of most relevant objects that have some spatial relation (e.g. near, between) with the given location of the user in order to explain to the user the geographical position the user is interested in or is located in. The explanation is generated from the output of ontology based retrieval result as human-readable, natural-language-alike text. The natural language generation process is not considered as this falls out of scope of current paper.

The paper is structured as follows. Section 2 provides description of motivating use case scenario. In section 3 an overview of ontology architecture is presented and section 4 describes ontology representation. Section 5 is devoted to ontology based retrieval of spatially related objects. Section 6 presents related works and section 7 concludes the work.

2 Motivating Use Case Scenario

In the following section we briefly describe industrial use case scenario that motivated us to create ontology architecture that could be used for retrieval of spatially related objects for LBS.

An end-user asks for a description of a geographical position (which may be carried out by sending a SMS to a reverse geo-coding service number). The position may be the one the user is standing at, a click on a map, or other. Usually, the nearest address-point is returned to the end-user, having the following concerns:

- The nearest address-point may not be accurate, for example the house number may be interpolated;
- If the end-user is in a foreign place/town, the address (a name of a by-street) may not be any help to the user – a description of nearby recognizable objects (main crossings, salient POIs), etc) will be more efficient.

In our extended reverse geo-coding service case, output is a human readable description of the geographic position, which describes the position giving relevant nearby objects, sights, POIs, etc, and their spatial relationship with the position. The textual descriptions of nearby recognizable objects may be accompanied with a map, highlighting these objects.

Input for this LBS may be a geographic position and its positional accuracy or uncertainty (meters radius of standard deviation) as well as user-profile characteristics like preferred languages, POIs, etc (see Table 1).

Table 1. Description of a geographic position

Positional accuracy	Long description of the position
Less than 10m	You are at 2, avenue Mere. It is in the centre of Tallinn, near Viru square, and near the crossing of Pärnu motorway, avenue Mere and Narva motorway; 100m from Viru hotel towards passenger port.
150m	You are at the centre of Tallinn, near Viru square, and near the crossing of Pärnu motorway, avenue Mere and Narva motorway; on a side of Viru hotel towards a sea.
1km	Centre of Tallinn

In Table 1, some examples of target textual descriptions of a geographic position are provided. Usually, descriptions represent one to four salient spatial objects that should meet the user requirements and should be retrieved from a larger set of possible candidate objects.

In order to implement this use case integration of several data sources, domain knowledge as well as knowledge about user preferences and context of LBS is needed.

For capturing this knowledge we provide ontology architecture that will support development of corresponding LBS and also other semantic LBS in the geospatial domain.

3 Ontology Specification and Overview

3.1 Goal, Scope and Formality Level of S3FW Ontologies

The main *goal* of ontologies in the S3FW is to give an explicit and formal meaning to domain concepts in order to make it possible to perform qualitative reasoning on *semantically* enriched spatial relationships for example as "is near to", or "in between", etc. that hold between instances of ontology classes. The ontology will be used as a formal, explicit model of the geospatial domain for S3FW components both at design and runtime phases.

The *scope* of the ontologies developed for S3FW is not limited to the reverse geocoding service described above, but most of these ontologies can be reused and extended for other geospatial applications and LBS. As the application is for mobile users, one of our goal was to limit the scope of application and domain ontologies as much as possible to provide satisfactory performance of the application at run time. We also used modularization principles for that purpose.

The ontology component of the S3FW is processed by software. In addition, human-friendly representation or explanation of ontological terminology in different languages is given in annotations and labels of a particular ontology class or property. This part of ontology description is also used for the verbalization of characteristics of retrieved instances. OWL DL [8] was chosen as the ontology language for S3FW ontology representation as it is formal, decidable and expressive enough.

3.2 The Architecture of Ontologies

In the following Fig. 1 the architecture of ontologies for ontology-driven LBS development framework S3FW is presented. One of the requirements of the ontology was to make it modular. At the highest level the ontology was divided into two separate modules: S3FW domain ontology and application oriented ontology.

This division has major advantage that in the future it would be easier to introduce new application ontologies by reusing the domain ontology.

As depicted in Fig. 1, all of the the ontologies in S3FW are related to each other. The S3FW geo-domain ontology is used as a baseline ontology for all application oriented ontologies in order to provide shared common vocabulary for applications. Application specific class definitions are added and merged to corresponding user profile ontologies in order to form a specific application ontology. Ontology merging means the creating of a new ontology from two or more ontologies. In this case, the new ontology will unify and replace the original ontologies. We have used the union approach of ontology merging, where the merged ontology is the union of all the entities of source ontologies, where differences in representation of similar concepts have been resolved.

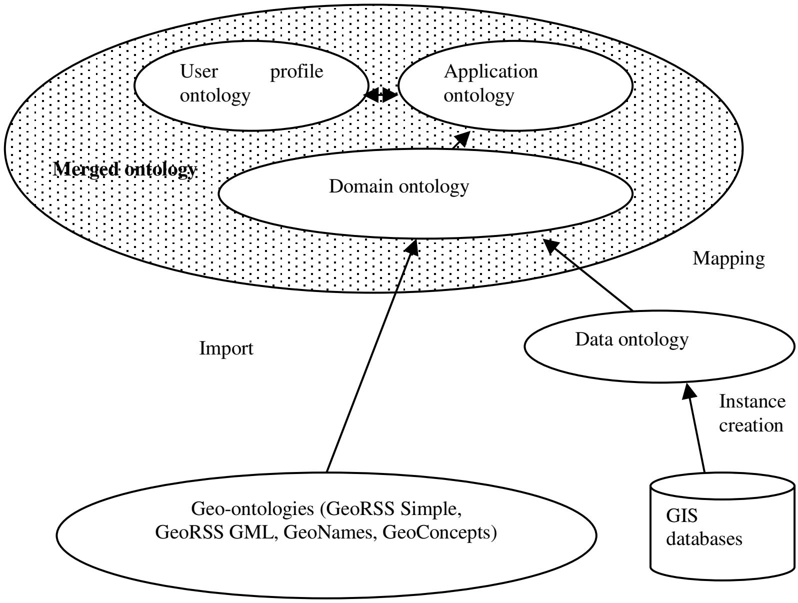

Fig. 1. Architecture of S3FW ontology

Finally we have produced one merged ontology that unified geo-domain ontology, application and user profile ontology, which is used dynamically by the LBS.

3.3 Data and Data Ontologies

Regio Ltd, the leading mapping agency of Estonia, maintains database of the topography of Estonia. The database contains a large number of geo-features representing towns, forests, roads, rivers, individual buildings, etc. This data is a source for many governmental, commercial, and scientific applications.

It is commonly known that most of legacy databases including GIS databases present hidden information in the field values making it hard to understand what these values mean at a domain level. Database tables are also designed according to certain design and performance constraints. In our work, we faced the similar problems when investigating GIS databases owned by our industrial partner Regio Ltd. Therefore we decided to use semantic technologies in order to make this hidden complexity of knowledge explicit in the form of ontologies as well as to present the hidden meaning of data from the application program code as much as possible.

Generally speaking, in a GIS domain data ontologies are needed for mapping from relational spatial geo-databases to domain ontologies. Data ontologies represent the underlying data model of a particular data source, for example a database of a specific geospatial product. In this work we demonstrate the need for data ontology (see Fig. 1) as a part of our ontology architecture. We do not, however, really concentrate on the creation of the data ontology since our main goal is to create domain, application, and profile ontologies for semantic LBS.

3.4 Geo-ontologies

There is a broad range of geospatial ontologies defined by several communities for different applications. In [9] 45 ontologies capturing spatial, event and temporal concepts have been studied. These ontologies are on different level of abstraction and formalisation. As a result, the study recommended seven full spatiotemporal ontologies for reuse. Among those, only SOUPA rcc provides qualitative reasoning that is based on Region Connection Calculus [10]. Most widely used ontologies in this list are GeoRSS Simple [11] that is meant for representing basic geometries (e.g. point, line, box, and polygon) and GeoRSS GML [11] that is created for representing more complex geometries and is standardised.

A geographic feature is an abstraction of a real world phenomenon that is associated with a location relative to the Earth [11]. No standardised feature type ontologies represented in OWL are currently available.

The Semantic Web for Earth and Environmental Terminology (SWEET) project provides ontologies that mainly cover concepts capturing the Earth science but only a small range of geo-feature types [12]. Geonames [2] released ontology of its geospatial information schema in the OWL and this ontology directly maps to the DB schema that is used by Geonames data export. The most important ontology class in this ontology is the *Feature* class that is a set of all geospatial instances in Geonames (a city, a country etc.). All feature instances are uniquely identified by Uniform Resource Identifier (URI). Their geospatial properties are defined using the vocabulary from the W3C geo ontology [13].

In contrast to full spatiotemporal ontologies, for commercial use of our application we are seeking for lightweight spatial ontologies with minimum geospatial representation that would meet our requirements. However, this “light“ ontology should be interoperable with full geospatial ontologies as well as take into account ontology standardizations efforts in geospatial domain. After evaluation of different ontologies, we decided to reuse one of the existing geospatial ontologies namely GeoConcepts ontology developed in the FP6 CINeSPACE project [14]. It is based on GeoOWL [11] standard and Geonames feature type hierarchy. W3C Geospatial Incubator Group [15] developed GeoOWL as a minimum geo-vocabulary which follows GeoRSS [11] guidelines. GeoConcepts is a rather simple ontology that defines some of the geo-concepts and a limited number of spatial relationships. The term “geo-concept“ in this ontology refers to any entity with an inherently or indirectly associated spatial dimension. Expressiveness of its formalization is as of ontology representation language OWL Lite.

4 Ontology Representation

4.1 Domain Ontologies

The baseline geo-domain ontology is created as general as possible in order to simplify its reuse by other applications based on S3FW.

As application and user profile ontologies are created on top of domain ontology, then this ontology module is the most important one. The Fig. 2 depicts the main groups of concepts of this ontology used in the following examples and their relationships represented by object properties as well as some of their datatype properties.

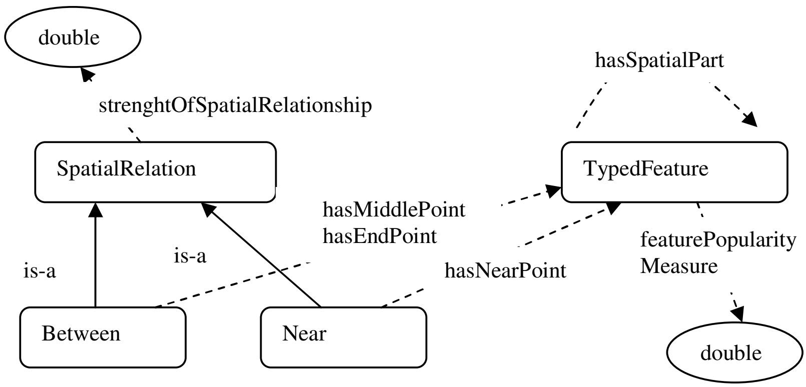

Fig. 2. Set of main groups of named classes of baseline geo-domain ontology and relationships between class instances

The main *TypedFeature* class represents different types of geo-entities (e.g. Grossing, Forest, Shop, Hotel, etc) by its subclasses. The individuals of *TypedFeature* class may be related to each other by spatial relationships represented by object properties or by subclasses of *SpatialRelation* class. Datatype properties describe several attributes of individuals of *TypedFeature* and *SpatialRelation* class as data values.

Representing Geo-features in Baseline Domain Ontology. The baseline geo-domain ontology for S3FW has been created by reusing GeoConcepts ontology by importing it and then adding the needed feature type subclasses or feature types. Fig. 3 shows a fragment of this ontology presenting only the particular subclasses that are used in examples outlined in this paper. By now the ontology contains 60 feature types in total and it will be extended as usage of this ontology in applications grows.

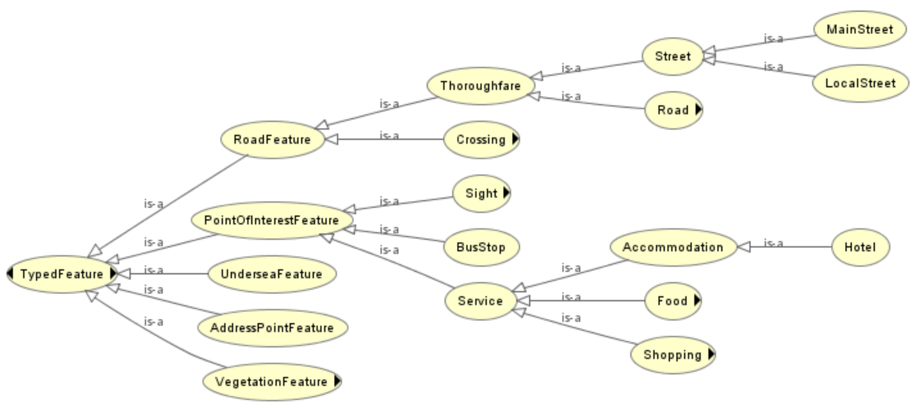

Fig. 3. A fragment of the baseline geo-domain ontology for S3FW (screenshot from the Protégé editor)

The first level subclasses of *TypedFeature* class are reused form GeoConcepts ontology, all other subclasses are originally created for S3FW domain ontology.

Representing the Semantic Spatial Relationships in Domain Ontology. Ontological model that would meet requirements of S3FW should also enable the qualitative reasoning over *semantic* spatial relationships. This means that the domain/application ontology must capture -beside feature types -also the qualitative aspects of space and their meaning to the end user.

On the other hand, current GIS engines support various spatial operators designed to determine the relationship between geospatial objects. These relationships are formalised for example in the Region Connection Calculus (RCC) [10, 16] and can be expressed using OWL and thus enable Description Logic (DL) reasoners to perform inference over spatial data based on topology. RCC8 and SQL spatial operators define very specific relationships between geometries. For example, some of the relationships defined in RCC8 are as follows: Connects with, Disconnected from, Part of, Overlaps, Partially Overlaps, Equal to, etc. However, these usually do not correspond to the *semantic* spatial relationships that domain experts (or end users) would naturally use or imply. Therefore, we provide a novel approach to use ontology to explicitly specify meaning of each of the spatial relationships used. We call spatial relationships enriched with specific meaning *semantic* spatial relationships. This approach is particularly useful for rather vague spatial relations like “near to”, “next to”, “in between”, etc. Some examples of *semantic* spatial relationships are defined in the domain ontology for the S3FW as follows.

For the S3FW domain ontology, we reused ordinary spatial relationships (e.g. hasSpatialPartOf, isPartOf, etc) from GeoConcepts ontology. As GeoCponcepts ontology does not include any *semantic* spatial relationships, then we added some useful for our application *semantic* spatial relationships like “is near to”, “is between” enriched with the meaning as perceived by end-users of the S3FW application. In principle, it is possible to define more *semantic* spatial relationships following our approach, for example, one can make use of *semantic* spatial relationships like “above”, “on corner of” “opposite”, etc. The approach is highly flexible and extendable; one can define *semantics* of any spatial relationship as needed for an application and take into account preferences of a certain user group.

In the ontology, all binary spatial relationships have been represented using object properties in OWL.

Semantic spatial relationships “is near to”, “is between” that serve our current application have been represented as n-ary relationships and therefore the corresponding subclasses of the class *SpatialRelations* have been created. The need for n-ary relationships rose because the mentioned spatial relationships are either themselves n-ary (e.g. “in between”) or they themselves have a property that measures strength of a particular spatial relationship. The latter is calculated according to the given spatial proximity (or positional accuracy) and distance giving possibility to take into account this characteristic when recommending spatial objects to the user. The value of this characteristic is between 0 and 1 and it is modelled by the datatype property *strengthOfSpatialRelationship*. Its value is calculated by GIS application program and is given as input to the ontology reasoning component.

For example, the class *Between* describes semantics of spatial relationship “in between” by the following necessary condition presented in class expression language of Protégé ontology editor:

```
SpatialRelation and strengthOfSpatialRelationship
exactly 1 Literal

and hasEndPoint exactly 2 TypedFeature

and hasMiddlePoint exactly 1 TypedFeature,
```

where *hasEndPoint* and *hasMiddlePoint* are object properties that express corresponding relationships between instances of the classes *Between* and *TypedFeature.*

For the reverse geo-coding service, we also added data-type property that captures values (from 0 to 1) to measure the feature's popularity (datatype property *featurePopularityMeasure*). The values are given by GIS for each instance of a feature type used for LBS.

4.2 Application and User Profile Ontologies

The application specific ontology refines the baseline geo-domain ontology by defining classes that enable classification of instances to relevant classes and filter out all irrelevant instances of the reverse geo-coding service. As mentioned above, we merged the application and the user profile ontologies in order to get better runtime performance. Therefore, the user profile ontologies are not considered as separate ontologies but are instead merged to application ontologies in order to refine the classification. For example, according to user profile our application can make classification of instances that would best match users' interests in certain feature types (e.g Hotels, Shops) and at the same time take into account application specific requirements for classification.

Currently, user profile ontology consists of 2 classes (see Fig. 4) and their subclasses that define particular restrictions for certain user profiles. By now, user profile classes are defined by application developers. In principle, we are in process to extend user profile ontology as well as to connect it to an application to automatically collect user profile characteristics.

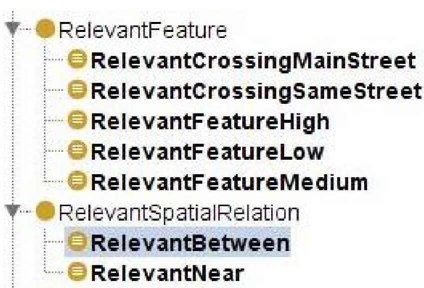

Fig. 4. An example of a user profile ontology (screenshot from the Protégé editor)

For capturing user profile semantics in our current LBS we defined classes *RelevantFeature* and *RelevantSpatialRelation*. The first class is for defining different restrictions to feature types and the second for restricting spatial relations to be considered as relevant. Subclasses of these define particular restrictions corresponding to the application and/or to the user profile. As usual, definitions of relevant spatial relations are built on the definitions of relevant feature types.

For example, for our current LBS we defined application oriented subclasses *RelevantCrossingMainStreet* and *RelevantCrossingSameStreet* as subclasses of *RelevantFeature* class. The class *RelevantCrossingMainStreet* had the following necessary and sufficient condition:

```
Crossing and hasSpatialPart some MainStreet.
```

For example, the class *RelevantFeatureHigh* from a user profile is defined using the following restriction:

```
DescribedLocation or (Hotel and
featurePopularityMeasure some double[>= 0.6])

or (RelevantCrossingMainStreet and
RelevantCrossingSameStreet).
```

This class defines a set of individuals that include the described location and hotels having popularity higher or equal to 0.6 and crossings formed by main streets and the street where the described location lies.

As our ultimate goal in current LBS –the reverse geo-coding service -is to retrieve objects relevant to the user and the user's location spatial relationships, then a class *RelevantSpatialRelationships* is created into the ontology and its subclasses define corresponding spatial relationships.

For example, the class *RelevantBetween* has the following definition:

```
Between and ((hasMiddlePoint some RelevantFeatureHigh)

or (hasMiddlePoint some RelevantFeatureMedium)

and hasEndPoint exactly 2 RelevantFeatureHigh)

or (hasEndPoint some RelevantFeatureHigh

and hasEndPoint some RelevantFeatureMedium

and hasMiddlePoint some RelevantFeatureHigh)
```

This definition means that at least two of the three instances in the relevant between relations must be of the class *RelevantFeatureHigh*: the third may be *RelevantFeatureHigh* or *RelevantFeatureMedium.*

4.3 Ontology Implementation

The ontology has been implemented in the OWL DL [8] by using Protégé ontology editor [17]. The domain ontology module has been implemented in a separate file: for each application a separate application (inc. user profiles) ontology file was created by using the approach described previously.

For a LBS demo program, we populated our domain ontology with instances by using the Jena Ontology API [18] and an application program written in Java that performed the needed queries to geo-databases. The application program also calculated for each instance of geo-entity type (subclass of *TypedFeature* class) the value of datatype property *featurePopularityMeasure* and for each instance of defined spatial relationship the value of datatype property *strengthOfSpatialRelationship*. In addition, the application program computed the existence of certain qualitative spatial relationships (e.g. “is near to”, “is between”) between geospatial entities and the given location of the user. Therefore all the needed quantitative reasoning was performed out of the scope of the S3FW ontology component. For ontology population, a set of candidate entities for relevant instances was selected from all available entities spatially related to the given location. Usually, there are hundreds of candidate relevant entities and to retrieve few (usually one to four) the most relevant entities with respect to user profile and application requirements is not an easy task. Only these few entities will serve for the description of the location to the user in natural language.

5 Ontology-Based Retrieval of Spatially Related Objects

Some reasoning capabilities are needed for the geospatial ontology to be useful. The knowledge base for reasoning consists of the ontologies as defined above and of the actual class instances. Currently, there are several DL inference engines available for reasoning over OWL ontologies, e.g Fact++ [19], RacerPro [20], and Pellet [21]. Using DL terminology, a knowledge base consists of a TBox, which contains the terminology, i.e the concepts within a given domain, and an ABox, which stores assertions (about named individuals) [22]. A DL system offers services to reason about the content of a knowledge base by providing standard reasoning services such as satisfiability, subsumption, and instance checking [22]. We use Pellet 1.5 as DL reasoner.

In this work, we combined quantitative and qualitative approaches for geo-spatial reasoning. First of all, quantitative reasoning was used in order to decide about existence of a certain *semantic* spatial relationship. For example, in the ontology used for our reverse geo-coding LBS we modelled two *semantic* spatial relationships: “near to” and “in between”. In order to decide whether these relationships exist within the given set of entities we used quantitative reasoning performed by GIS database application. This application calculated what individuals hold the spatial relationship “near to” or “in between” with respect to given location and performed certain preliminary selection of candidate entities. After ontology was populated with these candidate entities, qualitative reasoning was used for classification of individuals into predefined classes defining different user profiles and other application restrictions.

For example, using the defined class *RelevantBetween* (for definition see previous section) we may like to retrieve the instances of this concept. This means that such instances must be recognized automatically, the corresponding process is called ontology based retrieval. Reasoning is required in order to obtain these instances since there are no given instances of *RelevantBetween* class.

In S3FW, the task of automatic classification in ontology-based reverse geo-coding LBS is used to filter out geospatial objects that are deemed to be insignificant. Therefore, the defined classes in user profile and application ontologies can also be considered as filters or instance retrieval queries. In the implementation of LBS we delegated these queries to the Pellet DL reasoner. Results of these queries are given in RDF format and form the input for the verbalization component of S3FW.

For example, a fragment of RDF description of the result of retrieved instances by using *RelevantBetween* class as a query is depicted in the following RDF code:

…

```
<rdf:Description rdf:nodeID="A0">
    <rdfs:label xml:lang="en">the crossing of Inseneri
and Mere Street</rdfs:label>
    <rdf:type
rdf:resource="file:/c:/regio_geoonto.owl#Crossing"/>
  </rdf:Description>
  <rdf:Description rdf:nodeID="A1">
    <rdfs:label xml:lang="en">Location</rdfs:label>
    <rdf:type
rdf:resource="file:/c:/regio_geoonto_relations.owl#Desc
ribedLocation"/>
  </rdf:Description>
<rdf:Description rdf:nodeID="A3">
    <rdfs:label xml:lang="en">the crossing of Aia and
Inseneri Street</rdfs:label>
    <rdf:type
rdf:resource="file:/c:/regio_geoonto.owl#Crossing"/>
  </rdf:Description>
<rdf:Description rdf:nodeID="A4">
    <regio-geoonto-rel:hasendpoint rdf:nodeID="A0"/>
    <regio-geoonto-rel:hasendpoint rdf:nodeID="A3"/>
    <regio-geoonto-rel:hasmiddlepoint rdf:nodeID="A1"/>
    <rdfs:label xml:lang="en"></rdfs:label>
    <rdf:type
rdf:resource="file:/c:/regio_geoonto_relations.owl#Betw
een"/>
  </rdf:Description>
```

As we have in mind LBS, then restrictions to instances should be relatively restrictive but at the same time of wide variety. This is in order to provide the user with effective and understandable descriptions of the location. Even we take into account spatial proximity and distance by using values of datatype property *strengthOfSpatialRelationship* we still may face the situation that we have too many relevant objects and spatial relationships to recommend to the user. To avoid this problem, we may run experiments in order to tune the set of restrictions to give more relevant retrieval output. Another solution is that for current application, we may pick up just one individual from each different class of retrieved instances. This is because in order to describe the user's location it is not needed, for example to recommend all the nearby popular hotels as one may be enough to give the user an idea, where the user is located.

According to our experience, we may conclude that using the ontology component makes it very easy to adapt LBS to different user profiles and context (for example, different countries). This can be carried out by defining new classes with corresponding restrictions or by modifying the already existing class definitions.

6 Related Works

Related works concerning geo-ontologies are discussed in Section 2. A lot of effort is devoted to spatial ontologies modeling qualitative relationships like RCC [9, 10, 16]. The RCC is an axiomatization of certain spatial concepts and relations in first order logic. RCC8 and SQL spatial operators define very specific relationships between geometries. For capturing the meaning of spatial relationships used by domain experts or users of geo-spatial LBS-s ontology engineering methods could be used as demonstrated in the current paper.

Ontologies are widely used for modeling user profiles in many application areas including web users' behavior modeling and recommender systems [23, 24].

In [25] a methodology is presented to automatically extract geospatial ontologies from geo-databases. It is shown in this work that adding a semantic layer to databases makes it possible to refer to concepts that have no direct correspondence to the database table. This gives a semantic view to geographical data. As mentioned above, in this paper we do not consider data ontologies, nevertheless, we appreciate work what is done in this field.

There are spatial query-answering systems based on ontologies. Query expansion with ontology concepts is considered in the QuONTO system [26] but in this system no ABox retrieval was needed as database systems are used instead.

The DLMAPS system [27] provides spatio-thematic query-answering in the domain of city maps. However, compared to our work, they do not deal with LBS and also have different goals.

In [28] a method is presented that enables ontology based query of spatial data available from Web Feature Services (WFS) and from data stored in databases. The user queries are rewritten to WFS getFeature requests and SQL queries to database. In contrast to our OWL ontologies, their method uses RDF ontology, RDF views to database tables and SPARQL as a query language.

A similar approach to our ontology architecture can be found in [29]. They propose five types of ontologies that could support a geospatial semantic system and further semantic interoperability of geospatial systems and non-geospatial data sources. However, our ontology architecture is different and aimed at supporting semantic LBS.

7 Conclusion

In this paper we provided the ontology architecture consisting of geo-ontologies, data and domain ontologies as well as application and user profile ontologies in order to support development of different semantic LBS within S3FW. We have shown how these ontologies can be used for retrieval of relevant spatially related objects to the user. Our main contribution is to enable personalised ontology driven LBS by providing a novel approach for defining personalised *semantic* spatial relationships by means of ontologies. Novelty of our contribution lies on that it makes possible to reason on *semantic* spatial relationships that are not purely topological relationships but are enriched with meaning that domain experts, application developers and end users will expect.

As a proof of the concept we developed a reverse geo-coding service where traditional reverse geo-coding is *extended* by providing also the search of objects most relevant to the user that have some spatial relationship of the latters's location in order to explain the user's geographical position the user is interested in or is located in.

Our experience shows that using the proposed ontology architecture would make it easy to adapt LBS to different user profiles and context. The approach is highly flexible and extendable; one can define *semantics* of any spatial relationship as needed for an application and take into account preferences of a certain user group.

Our future work will be to extend the approach in order to automatically generate context-specific route descriptions based on semantic background knowledge for navigation LBS.

Acknowledgments. This research was partially supported by Enterprise Estonia's competence centre programme and by the target-financed theme No. 0322709s06 of the Estonian Ministry of Education and Research as well as by the ERDF funded Estonian Centre of Excellence in Computer Science, EXCS.

References

1. Open Geospatial Consortium, `http://www.opengeospatial.org`
2. Geonames, `http://www.geonames.org`
3. Google Maps, `http://maps.google.com`
4. Zipf, A.: User-Adaptive Maps for Location-Based Services (LBS) for Tourism. In: Woeber, K., Frew, A., Hitz, M. (eds.) Proc. of the 9th Int. Conf. for Information and Communication Technologies in Tourism. LNCS, pp. 329–338. Springer, Heidelberg (2002)
5. Rao, B., Minakakis, L.: Evolution of mobile location-based services. Communications of ACM 46(12), 61–65 (2003)
6. Gómez-Pérez, A., Fernández-López, M., Corcho, O.: Ontological Engineering with examples from the areas of Knowledge Management, e-Commerce and the Semantic Web. Springer, Heidelberg (2004)
7. ELIKO Competence Centre in Electronics-, Info- and Communication Technologies, `http://www.eliko.ee`
8. OWL DL, `http://www.w3.org/TR/owl-guide`
9. Ressler, J., Dean, M., Kolas, D.: Geospatial Ontology Trade Study, National Geospatial-Intelligence Agency (2007), `http://projects.semwebcentral.org/docman/?group_id=84`
10. Randell, D.A., Cui, Z., Cohn, A.G.: A spatial logic based on regions and connections. In: Proc. 3rd International Conference on Knowledge Representation and Reasoning, pp. 165–176. Morgan Kaufman, San Francisco (1992)
11. GEO RSS, `http://www.georss.org/`
12. SWEET project, `http://sweet.jpl.nasa.gov/2.0/`

13. W3C geo ontology, http://www.w3.org/2003/01/geo/#vocabulary
14. CINeSPACE project, http://www.cinespace.eu
15. W3 GeoIncubator Group, http://www.w3.org/2005/Incubator/geo/XGR-geo-ont-20071023
16. Grütter, R., Scharrenbach, T., Bauer-Messmer, B.: Improving an RCC-Derived Geospatial Approximation by OWL Axioms. In: Sheth, A.P., Staab, S., Dean, M., Paolucci, M., Maynard, D., Finin, T., Thirunarayan, K. (eds.) ISWC 2008. LNCS, vol. 5318, pp. 293–306. Springer, Heidelberg (2008)
17. Protégé, http://protege.stanford.edu
18. Jena Ontology API, http://jena.sourceforge.net/ontology/index.html
19. Fact++, http://owl.man.ac.uk/factplusplus
20. RacerPro, http://www.racer-systems.com
21. Pellet, http://www.clarkparsia.com/pellet
22. Baader, F., et al. (eds.): The Description Logic Handbook. Cambridge University Press, Cambridge (2003)
23. Robal, T., Haav, H.-M., Kalja, A.: Making Web Users Domain Models Explicit by Applying Ontologies. In: Hainaut, J.-L., Rundensteiner, E.A., Kirchberg, M., Bertolotto, M., Brochhausen, M., Chen, Y.-P.P., Cherfi, S.S.-S., Doerr, M., Han, H., Hartmann, S., Parsons, J., Poels, G., Rolland, C., Trujillo, J., Yu, E., Zimányie, E. (eds.) ER Workshops 2007. LNCS, vol. 4802, pp. 170–179. Springer, Heidelberg (2007)
24. Middleton, S., De Roure, D., Shadbolt, N.: Capturing Knowledge of User Preferences: Ontologies in Recommender Systems. In: 1st Int. Conference on Knowledge Capture, pp. 100–107. ACM Press, New York (2001)
25. Baglioni, M., Masserotti, M.V., Renso, C., Spinsanti, L.: Building Geospatial Ontologies from Geographical databases. In: Fonseca, F., Rodríguez, M.A., Levashkin, S. (eds.) GeoS 2007. LNCS, vol. 4853, pp. 195–209. Springer, Heidelberg (2007)
26. QuONTO project, http://www.dis.uniroma1.it/~quonto/
27. Wessel, M., Möller, R.: Flexible Software Architectures for Ontology-Based Information Systems. Journal of Applied Logic, Special Issue on Empirically Successful Systems 7(1), 75–99 (2009)
28. Zhao, T., Zhang, C., Wei, M., Peng, Z.-R.: Ontology-Based Geospatial Data Query and Integration. In: Cova, T.J., Miller, H.J., Beard, K., Frank, A.U., Goodchild, M.F. (eds.) GIScience 2008. LNCS, vol. 5266, pp. 370–392. Springer, Heidelberg (2008)
29. Kolas, D., Hebeler, J., Dean, M.: Geospatial Semantic Web: Architecture of Ontologies. In: Rodríguez, M.A., Cruz, I., Levashkin, S., Egenhofer, M.J. (eds.) GeoS 2005. LNCS, vol. 3799, pp. 183–194. Springer, Heidelberg (2005)