Supporting Frameworks for the Geospatial Semantic Web

A I Abdelmoty[1], P D Smart[1], B A El-Geresy[2], and C B Jones[1]

[1] Cardiff School of Computer Science, Cardiff University, Wales, UK
[2] School of Computing, University of Glamorgan, Wales, UK

Abstract. A lot of information on the web is geographically referenced. Discovering and linking this information poses eminent research challenges to the geospatial semantic web, with regards to the representation and manipulation of geographic data. Towards addressing these challenges, this work explores the potential of the current semantic web languages and tools. In particular, an integrated logical framework of rules and ontologies, using current W3C standards, is assessed for modeling geospatial ontologies of place encoding both symbolic and geometric references to place locations. Spatial reasoning is incorporated in the framework to facilitate the deduction of implicit semantics and for expressing spatial integrity constraints. The logical framework is then extended with geo-computation engines that offer more effective manipulations of geometric information. Example data sets mined from web resources are used to demonstrate and evaluate both frameworks, offering insights to their potentials and limitations.

1 Introduction

Over the past few years, geo-referencing of resources on the web has evolved to become a natural method for organising and linking information with the aim of facilitating its discovery and use. A significant portion of search queries include references to geographic places and spatial relationships [21, 7]. In response, geographic information retrieval has emerged as a research domain to address many challenges facing the development of geographically-aware search engines [16] including, geospatial query interpretation, geo-tagging of resources, spatial search and analysis and ranking and presentation of information.

On one hand, many of these challenges are problems that are addressed within the domain of GIS and spatial databases and could benefit from established approaches to their solution. On the other hand, these challenges are also being addressed, at a general level, within the evolving Semantic Web whose aim is to provide common frameworks that allow the sharing and reuse of data and services across applications, enterprise and community boundaries.

This paper studies the following question; Can the current semantic web technologies be "spatially-enabled" to allow the realisation of the geospatial semantic web? Towards answering this question, two frameworks are proposed. The first is based entirely on semantic web tools and technologies and is a logical integration of rules and ontologies to provide a platform for expressing and reasoning over symbolic geographic knowledge. The second framework is a hybrid extension of the basic framework with geospatial information processors that are more suited to manipulating the geometrical (location) component of the information. The potential and limitations of the frameworks are explored. Both are implemented using available tools and standards and are tested with some realistic data sets collected from web resources.

The nature of geospatial referencing as used on the web is discussed in section 2, followed by a proposal of a simple place model that encapsulates the dimensions of this data. Section 3 is an evaluation of OWL, as a standard web ontology language, for representing the proposed place model. A discussion of OWL's limitation motivates the use of rule layer over the ontology. A homogenous approach to the integration of such a rule layer is used in section 4 to form a basic framework for encoding a geospatial ontology and reasoning engine. The framework is evaluated with data sets extracted from Wikipedia. An extension to the framework that incorporates a spatial database system is proposed and evaluated in section 5 and the paper concludes in section 6.

2 Geospatial Referencing on the Web

Place names provide what is probably the most fundamental method of specifying location in natural language and hence also is the the most common form of geo-referencing used in web documents. A name may be a standardised widely recognised name, or informal being locally familiar in certain communities [**?**]. Further nominal clues are also used to distinguish location, for example, using some address information. If the information described is not exactly associated with the named place, then spatial relationships are used to describe location relative to that place, e.g. ”near” and ”north of”. In addition, the web now offers accessible mapping applications to allow for precise association of resources with a location on a map (e.g. linking photos on Flickr with Google maps). Unless, the resource is geo-located, such as with a GPS, a marker on a map is normally intended as an approximate pointer to the location of the resource.

The same is true when people query geo-referenced information. Typical structure of queries take the form $< subject >< relation >< somewhere >$ in which the subject specifies the thematic aspect of the web resource, somewhere is the name of a place and the relation stipulates a spatial relationship to the named place [15]. For example, the query ”holiday resorts in Southern Europe adjacent to a beach”, is a spatial query involving a combination of spatial joins and requires an estimation of the boundary of the region ”Southern Europe”. Gazetteers typically only provide a single point (centroid) to approximate the location of geographic regions. In addition, some regions, such as ”Southern Europe” are vernacular and do not have official recorded boundaries. To answer this query, additional knowledge is therefore required.

The web itself acts as a valuable source from which place information can be harvested to complement traditional gazetteers. Research methods (geo-parsing, coding and tagging) to find and extract this place information are being sought within the field of GIR [22, 3]. The task is challenging, involving problems not only with the extraction of information from natural language, but also with reasoning over the extracted data which may be incomplete, fuzzy and in cases contradictory.

Two types of geographic place data can be collected from web resources, qualitative data, in the form of place names and qualitative spatial relationships as well as some geometric information, in the form of mostly point data for the location of some of these places. Similar to processes normally undertaken in GIS and spatial databases, new methods for "cleaning" this geographic information are needed before they can be used as a base for spatial search and analysis. Collecting place information through *Crowdsourcing* (or user collaboration) is emerging and some web databases are already accumulating and serving these geographic data as RDF triples, to facilitate their sharing and integrated use.

In this paper, we use a simple place ontology that captures both types of data above as shown in figure 1. The model captures both qualitative and qualitative spatial description of location through the association of a place concept to a geometric footprint and the recording of different possible types of spatial relationships between places.

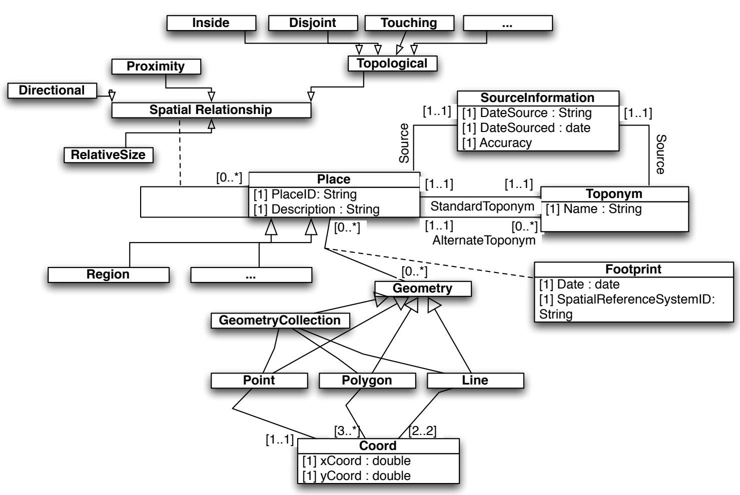


Fig. 1. A Typical Place Ontology Model

To demonstrate and evaluate the frameworks proposed, data sets are mined from the web to populate the place ontology. The following is an example, as RDF(S) triples, of the information mined from Wikipedia articles and stored in the model, where NS is the namespace prefix of: `http://cf.ac.uk/Place/`).

The triples encodes relationships between a set of regions (administrative wards in the city of Cardiff).

```
(<NS:Llanishen> <NS:Inside> <NS:Cyncoed>)
(<NS:Llanishen> <NS:Contains> <NS:Thornhill>)
(<NS:Penylan> <NS:Inside> <NS:Roath>)
(<NS:Penylan> <NS:Inside> <NS:Cathays>)
(<NS:Roath> <NS:Touches> <NS:Penylan>)
(<NS:Llanishen> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <NS:Region>)
(<NS:Cyncoed> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <NS:Region>)
(<NS:Thornhill> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <NS:Region>)
(<NS:Penylan> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <NS:Region>)
(<NS:Roath> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>  <NS:Region>)
(<NS:Region> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
                <http://www.w3.org/1999/02/22-rdf-syntax-ns#Class>)
(<NS:Inside> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
               <http://www.w3.org/1999/02/22-rdf-syntax-ns#Property>)
```

Databases such as Geonames and DBPedia store point coordinates for the places they hold in the form of a latitude longitude pair. The following is an RDF triple extract from both resources[3]. Interestingly, articles in DBPedia are linked to entries in Geonames using the `owl:sameAs` construct, allowing for possible integration of knowledge from both sources.

```
Geonames - Cardiff University

(<gns:Feature> <http://www.w3.org/1999/02/22-rdf-syntax-ns#about>
                  <http://sws.geonames.org/6697669/>)
(<http://sws.geonames.org/6697669/> <gns:Name> <Cardiff University Queens Buildings>)
(<http://sws.geonames.org/6697669/> <gns:FeatureClass> <http://www.geonames.org/ontology#P.PPL>)
(<http://sws.geonames.org/6697669/> <wgs84_pos:lat> <51.483^^XMLSchema:float>)
(<http://sws.geonames.org/6697669/> <wgs84_pos:long> <-3.16^^XMLSchema:float>)

DPPedia - Cardiff

<http://dbpedia.org/resource/Cardiff> <wgs84_pos:lat>
       <"51.4852777778"^^http://www.w3.org/2001/XMLSchema#float>
<http://dbpedia.org/resource/Cardiff> <wgs84_pos:long>
       <"-3.18666666667"^^http://www.w3.org/2001/XMLSchema#float>
```

Integrating these data resources poses many interesting research problems. The rest of this work focusses primarily on the following two basic problems.

- Are the available web languages and tools able to model this data effectively?
- Can these tools be used to reason effectively with the data to ascertain its consistency?

3 Evaluation of Current Semantic Web Tools

Ontologies are key to the development of the semantic web. They provide platforms for expressing and reasoning over common structures and vocabularies to facilitate sharing as well as machine understanding and reasoning of knowledge [12, 11]. Layers of technologies and languages are proposed by the W3C on

[3] where $gns = http://www.geonames.org/ontology\#, dbns = http://dbpedia.org/resource/\#and wgs84\_pos = http://www.w3.org/2003/01/geo/wgs84\_pos\#$

the semantic web stack to allow for the representation of ontologies, including the resource description framework (RDF), a basic schema definition language RDF(S), and a more expressive web ontology language OWL.

RDF provides a simple knowledge representation model using binary predicates or triples $< subject; predicate; object >$ asserting knowledge described by the predicate about the subject and object. RDF Schema (RDFS) [4] is an extension to RDF that provides base ontological constructs for defining custom vocabularies. RDFS can be considered a simple object-orientated language allowing user defined classes and properties. OWL extends RDFS and provides a richer set of modeling constructs and hence semantics and is considered to be the most complete and expressive web ontology language currently being developed. OWL is based on Description Logics (DL) and allow for the representation of concepts, concept hierarchies, roles and individuals. With its formal logical semantics, description logics support the following key inference tasks:

1. Subsumption reasoning - given concept C and D, determine if C is a subset of D. Checking if the concept D is more general than C.
2. Membership checking - check whether an individual i is a member of the concept C, or find all individuals that are an instance of C (a query).
3. Satisfiability checking - given concept C determine if C is consistent with respect to the knowledge base; checking whether a concept expression does not denote the empty set.

3.1 Using OWL for representing geographic knowledge

The place ontology in figure 1 can be represented using OWL-DL (the description logic subset of full OWL). A sample using XML/RDF syntax is shown below and a range of OWL-DL constructs used in the representation are given in table 1.

```
<owl:Class rdf:about="#Place">
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty>
          <owl:DatatypeProperty
           rdf:ID="Description"/>
        </owl:onProperty>
        <owl:cardinality rdf:datatype=
          "http://www.w3.org/2001/XMLSchema#int"
        >1</owl:cardinality>
      </owl:Restriction>
      </rdfs:subClassOf>
      ...
   </owl:Class>

<owl:ObjectProperty rdf:ID="Inside">
    <rdfs:domain rdf:resource="#Place"/>
    <rdfs:range rdf:resource="#Place"/>
</owl:ObjectProperty>
```

The expressiveness of OWL makes it a suitable modeling platform for different domains. However, it also has some limitations, as detailed below.

[4] http://www.w3.org/TR/rdf-schema/

Table 1. Sample OWL-DL constructs for the Place model

OWL-DL Construct	Description
Place	A Place is a concept
City $\sqsubseteq$ Place	A City is a sub-concept of Place
Ward $\sqsubseteq$ Place	A Ward is a sub-concept of Place
Place = $\geq$1.Name $\cap$ $\forall$ partOf.Place	A Place has one or more names, and can be partOf another place
SpatialRelationship	A spatial relationship is a property
Topological $\sqsubseteq$ SpatialRelationship	A topological property is a sub-property of a spatial relationship
Overlap $\sqsubseteq$ Topological	An Overlap property is a sub-property of a spatial relationship
PartOf $\sqsubseteq$ Topological	A PartOf property is a sub-property of a spatial relationship
Equal $\sqsubseteq$ PartOf	An Equal property is a sub-property of a spatial relationship
PartOf^{+} $\sqsubseteq$ PartOf	PartOf is a transitive property
partOf $\equiv$ contains^{-}	PartOf is equivalent to the inverse of the Contains property
City $\equiv$ Stadt	A City concept is equivalent to the concept Stadt (City in German)

1. OWL's first order, open world semantics in combination with the non-unique name assumption makes it unsuitable for constraint checking tasks [5]. For example, qualified cardinality constraints can't be used to constrain and check the possible instantiations of a class.
   Consider the following OWL definition of a Polygon,

$$Polygon \;>\; 3.XYCoords$$

   If an individual of type Polygon had two $XYCoords$, the open world assumption would concede that information may exist external to the ontology which can later be added to satisfy the restriction. If an individual had more than three $XYCoords$ then, as OWL does not support the unique name assumption, it will infer that all redundant coordinates are equal.
2. 'Triangular knowledge' is not representable in OWL-DL [13]. In particular, complex property compositions which are inference patterns of the form,

$$\forall x, y, c : R_1(x, y) \wedge R_2(y, c) \rightarrow R_3(x, c)$$

   where R_1,R_2 and R_3 are different relations, can not be handled. OWL v1.1. adds a restricted complex property inclusion axiom that can capture a limited form of an inference rule as follows.

$$R(x, y) \wedge S(y, c) \rightarrow S(x, c)$$

or

$$R(x, y) \wedge S(y, c) \rightarrow R(x, c)$$

Such axioms only permit the conclusion of a property used in the body of the composition, guaranteeing decidability, but will still not handle the more general form of complex property compositions.

3. Tableaux based reasoners (as used in most DL reasoners) are poor for query answering over individuals [5] and hence will pose a scalability problem for typically large spatial knowledge bases.
4. A further issue, particular to geospatial domains, is related to the representation and manipulation of the geometry. Logic-based paradigms are not suitable for the expression of procedural implementation of spatial operations, nor could they offer efficient storage structures or spatial indexes.

The limitations of OWL has led to proposals for enhancing its expressiveness in particular by exploring approaches for the representation of rules over ontologies. Different methods have been proposed and a rule layer is now part the semantic web stack.

Approaches to the integration can broadly be classified as either hybrid or homogeneous [1], reflecting the degree of interaction between the rule and ontology components. A hybrid approach is a modular approach where both the rule and ontology components are kept distinct. Reasoning is performed separately in both components and entailments from one component are treated as constraints to the other.

A homogenous approach is characterised by the complete translation of one language into the other. Approaches exist based on the expressive union of the two languages, as for example in the standard web ontology rule language (SWRL [14]) . However, the union introduces undecidability in the resultant language [2]. More commonly approaches are built around the common intersection of the language, as for example in the web rule language (WRL)[5] and description logic programs (DLP) [9]. Homogenous approaches offer a better reasoning synergy between ontology and rule components, as they form in effect one language. Furthermore, integrations based on the intersection of rule and ontology component can be used within existing, mature and scalable logic programming engines.

Description Logic Programs (DLP) is an example of a homogenous approach to integration and offers the following useful features.

- A significant number of commonly used constructs of OWL-DL can be captured within DLP[23].
- DLP is considered a sound, practical and extensible paradigm [17] and is the base for the core web rule language WRL.
- DLP can be run by existing forward chaining production systems such as RETE or backward chaining classical logic programs without modification.

[5] http://www.w3.org/Submission/WRL/

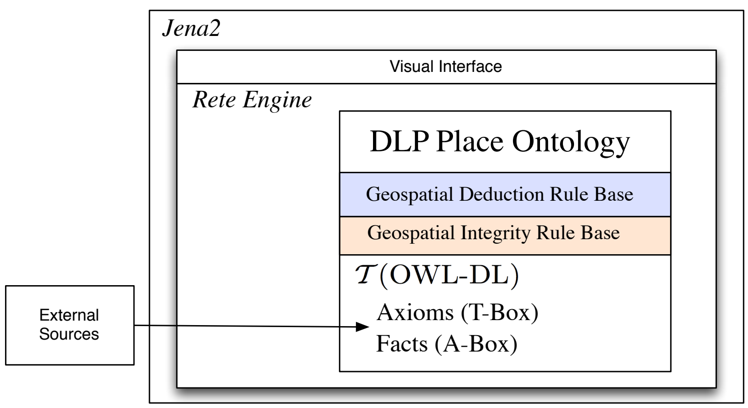
– Logic programming engines are better at reasoning with large stores of individuals (as in the case of geospatial knowledge bases) than tableaux-based DL reasoners [18, 17].
– DLP assumes a more intuitive closed world and unique name assumption and is consequently a suitable language for expressing and implementing integrity constraints, in addition to deductive rules.

In the rest of this paper DLPs are used as a base framework for managing our geospatial ontology bases.

4 Description Logic Programs Framework

A Description Logic Program (DLP) framework is proposed here as a base for representing and reasoning over geospatial knowledge base. First, we show how the modeling constructs in OWL can be transformed and expressed in DLP and then how it can be used to represent spatial rule bases for deduction and integrity checking.

Fig. 2. DLP Place ontology Framework



4.1 Mapping Geospatial Ontologies from OWL to a DLP

A transformation function $\mathcal{T}$, as defined in [10], is used here to map the OWL-DL representation of the place ontology into a DLP as shown in table 2. In practice this transformation can be performed using the KAON2 DLP convert program [18].

OWL-DL Syntax	DLP Horn Syntax
Place $\sqsubseteq$ Thing	Place(x) $\rightarrow$ Thing(x)
Region $\sqsubseteq$ Place	Region(x) $\rightarrow$ Place(x)
$\top \sqsubseteq \forall$ PlaceID.xsd:string	PlaceID(x,y) $\rightarrow$ xsd:String(y)
$\top \sqsubseteq \forall$ PlaceID^{-1}.Place	PlaceID^{-1}(x,y) $\rightarrow$ Place(x)
Topological $\sqsubseteq$ Spatial_Relationship	Topological(x,y) $\rightarrow$ Spatial_Relationship(x,y)
Touches $\sqsubseteq$ Topological	touches (x,y) $\rightarrow$ Topological(x,y)
...	...

Table 2. Sample DLP Place Ontology using the transformation function $\mathcal{T}$

.

Note, that the following constructs of the OWL-DL place ontology could not be represented in a DLP (see [10] for a more in-depth description of features not supported in a DLP):

- Functional properties, for example that each place has a unique ID.
- Cardinality restrictions, for example that each place has only 1 standard name.

In addition to representing the base axioms of the place ontology, a DLP allows for the definition of arbitrary (Horn) rules. Two principle types of rules can be expressed, namely, deduction and integrity, as shown below.

4.2 Deduction Rules

DLP can represent arbitrary deduction rules that can capture certain spatial compositional inferences that result in a definite conclusion (one head predicate) i.e. rules of the form:

$$Inside(A, B) \wedge Disjoint(B, C) \rightarrow Disjoint(A, C)$$

Although not strictly part of a DLP, procedural attachments can be easily added within all logic programming reasoning engines [17]. These are described later in the paper.

4.3 Integrity Rules

The logic programming equivalent of Horn logic used by a DLP assumes a more intuitive closed world and unique name assumption and is consequently a suitable language for expressing and implementing integrity constraints. The bodies of integrity and deduction rules are identical in both specification and functionality. An integrity rule differs from a deduction rule in the use of its head atom. An integrity rule does not assert new information into the ontology[6], instead it asserts errors into an error ontology.

For example, consider the following rule with (where A, B and C are variables).

$$Inside(A, B) \wedge Inside(B, C) \wedge Equal(A, C) \rightarrow error(t_1, ..., t_n) \quad (1)$$

Here the head predicate is an error predicate that is inferred if the body predicates (relations) exist in the DLP knowledge base. In this rule, if a place bound to the variable A is inside one bound to B, and B is inside a third place bound to C. An invalid state is reached and an error inferred, if a contradictory fact is explicit in the DLP that states that A is Equal to C. An set of integrity rules to capture possible invalid states for different types of spatial relations need to represented in the DLP. The resulting inferred error predicates are recorded and can be examined at the end of the inference process to identify the inconsistencies and trace their sources.

5 Framework Implementation

A system has been developed that implements the DLP proposed framework above within the Jena2 Semantic Web toolkit [7]. Jena2's rule engine is based on the Rete pattern matching production system [8] and an XSB [20] logic programming engine.

The system has been tested on real world place information mined from both Wikipedia pages and general web pages. The mined information is stored using the place ontology in OWL and then converted to a DLP program using the KAON2 DLP convert tool, and loaded into Jena2 as a set of logical rules in RDF triple format. A spatial rule base representing the composition of spatial relations has been developed using topological composition table [6, 4, 19]. The design and implementation of the spatial reasoning methods are assumed here and are outside the scope of the current paper.

Example: The instantiated place ontology contains 40 regions or neighborhoods within Cardiff, UK and roughly 200 explicit topological spatial relationships between these regions. The following are example of facts

```
(NS:Penylan rdf:type  NS:Ward)
(NS:Penylan NS:Inside NS:Roath)
(NS:Penylan NS:Inside NS:Cathays)
```

The engine checks the consistency of the ontology and reports the detected problem facts. A visual interface has been designed to allow for the visualisation

---

[6] As is common in logic programming literature, a rule without head is referred to as an integrity rule

[7] http://dsonline.computer.org/0211/f/wp6jena.htm

and editing of the ontologies and rules, as shown in figure 3. The result of the reasoning process is shown on the interface where problem relations (edges) are highlighted. In addition, a trace of the reasoning process can be produced to localise the source of the inconsistency in the data set.

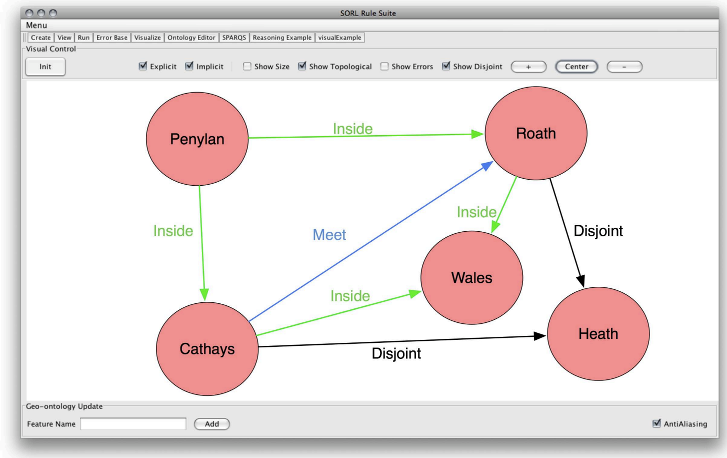


Fig. 3. Place Ontology Visual interface with a sample of the individuals in the ontology

An example of the error detected in this sample data set are the three relationships between the districts Cathays, Roath and Penylan, shown in figure 4(a). In reality, Penylan and Roath are neighbours, as shown in the Google maps view in figure 4(b). To find this inconsistency, the following integrity rules were triggered.

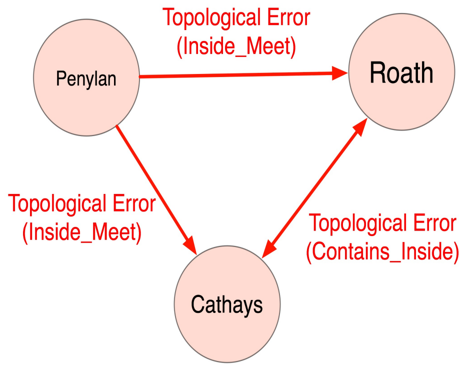
```
 [Inside_Meet :  (?x rdf:type NS:Region) (?y rdf:type NS:Region)
Region(?z rdf:type NS:Region) (?x NS:Inside ?y) (?y NS:Meet ?z)
 (?x NS:Inside ?z) -> error(?x ?z)]
```

Where Penylan is inside Cathays and Roath meets Cathays implies that Penylan can not be inside Cathays, and hence the rule implies an error.

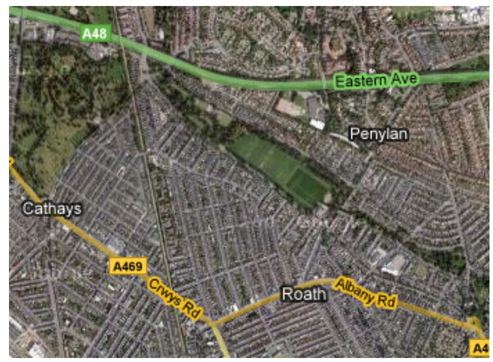
```
[Contains_Inside: (?x rdf:type NS:Region) (?y rdf:type NS:Region)
Region(?z rdf:type NS:Region)  (?x NS:Contains ?y) (?y NS:Inside ?z)
(?x NS:Meet ?z) -> error(?x ?z)]
```

Where Roath contains Penylan and Roath meets Cathays means that Cathays can not contain Penylan and hence the rule implies an error.

The DLP framework reasons with explicitly stored spatial facts in the ontology base but will not compute the facts if they are stored. Hence, its effectiveness



(a)



(b)

Fig. 4. a) Inconsistencies found between the regions Cathays, Roath and Penylan, b) Google Maps View of the three regions

is related to the number and types of spatial relations defined. Figure 5 demonstrates how the number of definite as well as indefinite spatial relations between regions in the ontology varies depending on the number of pre-defined explicit relations. The figures is based on the experiment with the ontology built from web resources used in the example. The total coverage refers to how many spatial relation in the ontology that are not the universal relation (a disjunction of all possible eight base topological relations). For instance, if the coverage is 100% then every region is connected to every other region by either a definite or indefinite topological relation. The number of definite relations is the percentage of region to region relations that are definite (only one topological relation).

6 The Extended Framework

Information on the object's location, shape and size can be used to directly compute its relationships to other object. A system for managing geo-referenced data need therefore to be able to make effective use of available geometric representations. Logic programming does not naturally support the representation and manipulation of these facts, but it can link up with processors that are more suited to these tasks. In addition, coordinate data representing boundaries of geofeatures can increase the storage (and memory) overhead significantly for an ontology base and stretches the capabilities of current technologies for reasoning with them. A sample geographic ontology base with 10 classes and around 10,000 individuals was created for a data set of European administrative boundaries. Classes were associated with 2 properties and 3 datatype properties. The detailed representation of the boundary data resulted in an OWL ontology that occupied 100MB of persistent storage space and approximately 800MB of memory.

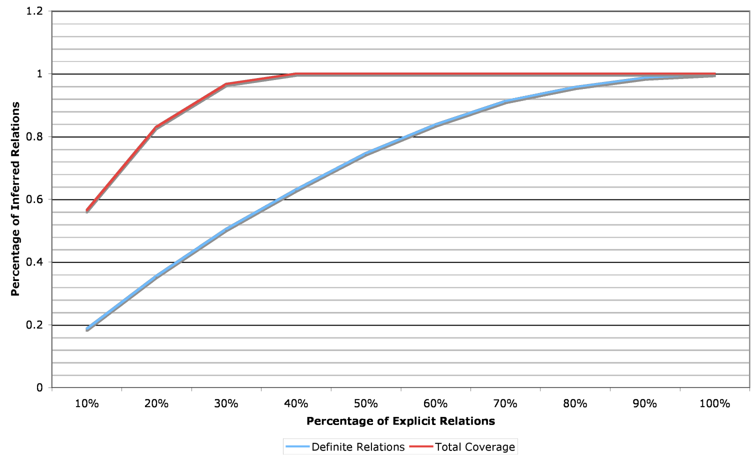


Fig. 5. Percentage of explicit (raw) relations vs. percentage of inferred relations in the sample ontology data set

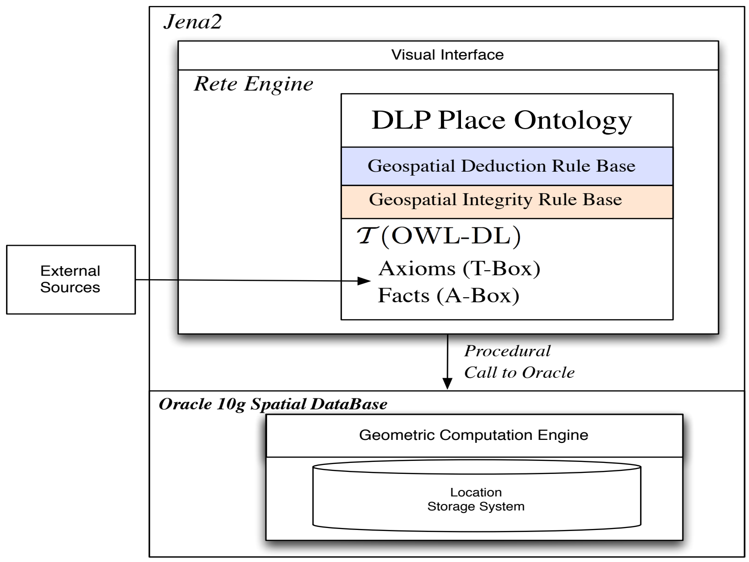
A hybrid extension to the framework is therefore proposed here to integrate an external geometric computation engine, to which the storage and manipulation of the geometric component of the geospatial ontology bases can be delegated. The extended framework is shown in figure 6. The Location Storage System (LSS) can in practice be a spatial database system (Oracle spatial is used in our case). All geometries are mapped directly in the LSS. An example of the mapping is shown in table 3.

Table 3. Example Geometry Mapping

Place Geometry	(Oracle) Table
District(Roath)→	INSERT INTO locationBase
Geomtry→	VALUES('http://cf.ac.uk/Roath',
polygon→	MDSYS.SDO_GEOMETRY
Coord(3,13)	(2003,8307,null,
Coord(11,13)	MDSYS.SDO_ELEM_INFO_ARRAY
Coord(11,21)	(1,1003,1), MDSYS.SDO_ORDINATE_ARRAY
Coord(3,21)	(3,13,11,13,11,21,3,21,3,13)))
Coord(3,13)	

The unique URI reference to a place instance in the DLP ontology is maintained in the LSS. This allows place instances in the DLP to be linked to their associated geometries in the LSS. In practice, all calls to the LSS take place through procedural attachments from the core DLP.

Fig. 6. Extended DLP Framework



6.1 Procedural Attachments for Spatial Operators

Many logic engine implementations provide a set of static predefined procedural attachments, denoted *builtins*. *Builtins* commonly revolve around simple arithmetic procedures or comparison procedures. Extending a DLP with procedural attachments can lead to a more complicated semantic treatment if the attachments are allowed to effect the logic program in any way, by for example removing facts from the knoweldge base. Semantically clean builtins are those that only test or compute facts and will not change or remove facts in the DLP.

In addition to standard *builtins*, a set of spatial *builtins* (or spatial operators) needs to be defined to link between the DLP component and the external geo-computation engine. Examples of these procedural attachments are given in table 4.

6.2 Interleaved Reasoning

Typically, all rule body antecedents are matched from existing stored facts (facts derived by rules or explicitly represented). Interleaving forward and backward reasoning modes in a logic program allow for the derivation of facts on the fly if they are not explicitly stored. Consider the following rule:

$$[Region(?x) \wedge Region(?y) \wedge Region(?z) \wedge Inside(?x?z) \wedge Inside(?z?y) \rightarrow Inside(?x?y)]$$

The conclusion of `Inside(?x ?y)` would only be inferred if both the atoms `Inside(?x ?z)` and `Inside(?z ?y)` can be satisfied. These atoms are either

Table 4. DLP Spatial Procedural Attachments

Procedural Attachment	Arguments	Oracle
exAdjacent	(Ind_1, Ind_2)	SELECT c_b.rdfID, c_d.rdfID, SDO_GEOM.RELATE (c_b.shape, 'TOUCH', c_d.shape, 0.005) FROM <tableName> c_b, <tableName> c_d WHERE c_b.rdfID = <ind1> AND c_d.rdfID = <ind2>
Area	(Ind_1, R)	SELECT SDO_GEOM.SDO_AREA (loce.shape, 0.005,'unit= <unit>) FROM <tableName> loce WHERE loce.rdfID = <ind1>
exDisjoint	(Ind_1, Ind_2)	$\cdots$
Distance	(Ind_1, Ind_2, R)	$\cdots$

satisfied by facts directly stored in the ontology (explicit), or inferred using reasoning rules, or as a last resort satisfied by a rule that calls the external geo-computation engine.

For example, the following is a subset of rules used to derive the inside relationship between two regions. The fifth rule is a call to the external (*exInside* predicate). Hence, `Inside(?x ?y)` will return either true or false, based on whether the relationship exists in the ontology, can be inferred, or whether it can be determined from the geometry.

$$Inside(?x\ ?y) \leftarrow Region(?x) \wedge Region(?y) \wedge Region(?c) \wedge Inside(?x\ ?c) \wedge Equal(?c\ ?y)$$
$$Inside(?x\ ?y) \leftarrow Region(?x) \wedge Region(?y) \wedge Region(?c) \wedge Inside(?x\ ?c) \wedge Inside(?c\ ?y)$$
$$Inside(?x\ ?y) \leftarrow Region(?x) \wedge Region(?y) \wedge Region(?c) \wedge Inside(?x\ ?c) \wedge CoveredBy(?c\ ?y)$$
$$Inside(?x\ ?y) \leftarrow Region(?x) \wedge Region(?y) \wedge Region(?c) \wedge CoveredBy(?x\ ?c) \wedge Inside(?c\ ?y)$$
$$Inside(?x\ ?y) \leftarrow Region(?x) \wedge Region(?y) \wedge Region(?c) \wedge exInside(?c\ ?y)$$

Example: The following qualitative relations were mined from Wikipedia related to the region "South Glamorgan"; an administrative subdivision of Wales.

$$contains(\text{Wales,Vale-of-Glamorgan})$$
$$inside(\text{Vale-of-Glamorgan, South-Glamorgan})$$

The spatial deduction rules suggest that South Glamorgan must be connected to Wales through a number of possible relations using the following rule.

$$inside^{-1}(\text{A,B}) \wedge inside(\text{B, C}) \rightarrow Overlap(A,C) \vee Contains(A,C) \vee Inside(A,C) \vee Equal(A,C) \vee Covers(A,C) \vee CoveredBy(A,C)$$

Consequently, South-Glamorgan can't be disjoint from Wales, as identified by the following integrity rule.

$$inside^{-1}(A,B) \wedge inside(B,C) \wedge disjoint(A,C) \rightarrow error(A,C) \tag{2}$$

Data are also recorded for the boundary points of Wales as well as point locations for the all the regions concerned (retrieved from Geonames). Firing integrity rule (2) results in interleaved reasoning where each of the predicates (spatial relations) in the rule are determined using the set of spatial composition rules in the system. The relation *disjoint* however, is not stored explicitly. To check this relation, an external call to the geo-computation engine is fired using the builtin `exDisjoint(A,B)`. The call returns ”True” indicating the fact that the geometry point location of South-Glamorgan is in fact outside the boundary of Wales. This contradicts with the facts already stored and hence an error is implied. Figure 7 shows the point location for South-Glamorgan, falling in the sea, as recorded in Geonames.

Fig. 7. Geonames South Glamorgan Geometric Error

The example demonstrates how the two types of reasoning; qualitative and quantitative, supported by this framework can be complementary to one another. Spatial relations are computed on the fly, when needed, within a logical reasoning framework.

7 Conclusion

In this paper we explore the idea of ”spatially-enabling” the semantic web. As geo-referencing of resources on the web becomes more popular, methods to support the search, sharing and linking of these resources are needed. The semantic web offers standard languages and tools to enable the representation and reasoning with the data. This paper demonstrates how these tools can be used for geospatial domains.

In particular, OWL-DL is used to store a basic model of place and spatial relationships. A homogeneous approach to integrating rules with OWL, namely, description logic programs DLPs, was shown to allow the expression of spatial deduction and integrity rules. A framework based on DLPs is proposed and is shown to support, terminological as well as spatial reasoning over geographical ontology bases.

The logical framework will however, not cope well with the demands of the geometric representations of geo-features. An extended framework is proposed to link the DLP with external geometric computation processors. It is shown how this link can be established using procedural attachments. The resultant framework supports both logical and geometric manipulation of geospatial facts and data, thus combining the strengths of both paradigms. Some realistic data sets mined from web sources are used for demonstration and for evaluating the proposed frameworks.

The contribution of the work is in demonstrating possible approaches to geospatial data management on the web and in highlighting the needs of geospatial domains that stretches the current semantic web tools and languages. Future work will consider the issue of scalability and other challenges related to problems of integrating and linking of geospatial data from different sources.

References

1. Antoniou, G., Damásio, C. V., Grosof, B., Horrocks, I., Kifer, M., Ma, J., and Peter. Combining Rules and Ontologies. A survey., 2005.
2. Brachman, R. J., Borgida, A., Mcguinness, D. L., Patel-schneider, P. F., and Resnick, L. A. The classic knowledge representation system, or, kl-one: The next generation. In *Preprints of the Workshop on Formal Aspects of Semantic Networks, Two Harbors* (1989), MorganKaufman, pp. 1036–1043.
3. Buyukokkten, O., Cho, J., Garcia-Molina, H., Gravano, L., and Shivakumar, N. Exploiting geographical location information of web pages. In *Proceedings of Workshop on Web Databases (WebDB'99)* (June 1999). Held in conjunction with ACM SIGMOD'99. Available at http://dbpubs.stanford.edu/pub/1999-4.
4. Cohn, A., and Hazarika, S. Qualitative spatial representation and reasoning: an overview. *Fundamenta Informaticae 45* (2001), 1–29.
5. de Bruijn, J., Lara, R., Polleres, A., and Fensel, D. Owl dl vs. owl flight: conceptual modeling and reasoning for the semantic web. In *WWW '05: Proceedings of the 14th international conference on World Wide Web* (New York, NY, USA, 2005), ACM Press, pp. 623–632.
6. Egenhofer, M. Deriving the composition of Binary Topological Relations. *Journal of Visual Languages and Computing 5* (1994), 133–149.
7. Fonseca, F. T., Davis, C. A., and Câmara, G. Bridging ontologies and conceptual schemas in geographic information integration. *GeoInformatica 7*, 4 (2003), 355–378.
8. Forgy, C. Rete: A fast algorithm for the many patterns/many objects match problem. *Artif. Intell 19*, 1 (1982), 17–37.
9. Grosof, B. N., Horrocks, I., Volz, R., and Decker, S. Description logic programs: combining logic programs with description logic. In *WWW* (2003), pp. 48–57.

10. Grosof, B. N., Horrocks, I., Volz, R., and Decker, S. Description logic programs: combining logic programs with description logic. In *Proceedings of the twelfth international conference on World Wide Web* (2003), ACM Press, pp. 48–57.
11. Gruber, T. R. A translation approach to portable ontologies. *Knowledge Acquisition 5*, 2 (1993), 199–220.
12. Guarino, N. Formal ontology, conceptual analysis and knowledge representation. *International Journal of Human-Computer Studies 43*, 5/6 (1995), 625–640.
13. Horrocks, I. Owl rules, ok? In *Rule Languages for Interoperability* (2005).
14. Horrocks, I., Patel-Schneider, P. F., Tabet, H. B. S., Grosof, B., and Dean, M. Swrl: A semantic web rule language combining owl and ruleml. Internet Report, May 2004. http://www.w3.org/Submission/2004/SUBM-SWRL-20040521/.
15. Jones, C., Abdelmoty, A., and Fu, G. Maintaining ontologies for geographical information retrieval on the web, 2003.
16. Jones, C. B., Purves, R., Ruas, A., Sanderson, M., Sester, M., van Kreveld, M., and Weibel, R. Spatial information retrieval and geographical ontologies an overview of the spirit project. In *SIGIR '02: Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval* (New York, NY, USA, 2002), ACM, pp. 387–388.
17. Krötzsch, M., Hitzler, P., Vrandecic, D., and Sintek, M. How to reason with OWL in a logic programming system. In *RuleML* (2006), T. Eiter, E. Franconi, R. Hodgson, and S. Stephens, Eds., IEEE Computer Society, pp. 17–28.
18. Motik, B., c Vrande, Hitzler, P., Sure, Y., and Studer, R. dlpconvert – converting owl dlp statements to logic programs, 2005.
19. Nebel, B., and Renz, J. Efficient methods for qualitative spatial reasoning, June 19 1998.
20. Sagonas, K., Swift, T., and Warren, D. S. Xsb: An overview of its use and implementation. Tech. rep., Nov. 02 1993.
21. Sanderson, M., and Kohler, J. Analyzing geographic queries, Aug. 09 2004.
22. Silva, M. J., Martins, B., Chaves, M. S., Afonso, A. P., and Cardoso, N. Adding geographic scopes to web resources. *Computers, Environment and Urban Systems 30*, 4 (2006), 378–399.
23. Volz, R. *Web Ontology Reasoning with Logic Databases.* PhD thesis, Universität Karlsruhe (TH), Universität Karlsruhe (TH), Institut AIFB, D-76128 Karlsruhe, 2004.