Maintaining Ontologies for Geographical Information Retrieval on the Web

Christopher B. Jones, Alia I. Abdelmoty, and Gaihua Fu

Department of Computer Science, Cardiff University, Wales, UK
{c.b.jones,a.i.abdelmoty,gaihua.fu}@cs.cf.ac.uk

Abstract. A geo-ontology has a key role to play in the development of a spatially-aware search engine, with regard to providing support for query disambiguation, query term expansion, relevance ranking and web resource annotation. This paper reviews these functions, discusses the user requirements which influence the design of the ontology, with regard to different types of query and fundamental spatial concepts, before presenting a base model for a geographical ontology which will provide a foundation for subsequent implementation as well as experimentation with alternative ontology models. The report also reviews various ontology languages available for expressing ontologies and give examples for encoding the geo-ontology in them.

1 Introduction

This paper is concerned with intelligent web-based information retrieval of geographical information. The assumption is that people may wish to find information about something that relates to somewhere. The most common way to refer to a location is to use place names, which may be qualified by spatial relationships (such as in or near). In order to assist in recognising place names and spatial relationships when they are employed in a search engine query it is proposed to employ an ontology which encodes geographical terminology and the semantic relationships between geographical terms. The idea is that the geographical ontology, henceforth denoted geo-ontology, will enable the search engine to detect that the query refers to a geographic location and to perform a search which will result in the retrieval and relevance ranking of web resources that refer both exactly and approximately to the specified location [Ala01]. This will entail retrieval of resources that refer to alternative versions of a specified place name as well as to places that are spatially associated with it or through relations such as those of containment and adjacency. It is also proposed that an ontology should be used to assist in a process of metadata extraction whereby the geographical context of resources is determined for the purpose of search engine indexing as well as providing the potential to annotate a resource to improve its future geographical visibility.

In this paper, issues and considerations related to the design and maintenance of such an ontology are explored. In section 2, the role of the Place ontology as a component of a spatially aware search engine is described. An overview of related

R. Meersman et al. (Eds.): CoopIS/DOA/ODBASE 2003, LNCS 2888, pp. 934–951, 2003.

research is given in section 3. Design issues and implementation considerations are discussed in section 4. This is followed by a proposal for a conceptual design of the ontology in section 5. Some operations on the ontology are also described. In section 6, possible tools for encoding and maintaining the ontology are reviewed and one tool namely, DAML + OIL is used for demonstration.

2 Roles of the Geo-ontology

The main distinguishing factor of the Spatially-Aware Search Engine envisioned in this paper, hence forth, denoted SPIRIT, is its search for information about a Place. Hence, queries to SPIRIT are characterised by their need to identify, either precisely or vaguely, a Place, which may be an extended region in space. A query to SPIRIT will be associated with a geographical context. The search engine needs to match the geographic content of the query with that of the available resources and the most relevant resources would then be returned. Definitions of the concepts of geographical content and geometric footprint associated with queries and documents are first introduced, and then the roles of geo-ontology in SPIRIT are discussed.

2.1 Basic Definitions

A reference to a geographic Place could be by its name and/or by its location. A location is either absolute or relative. The type of the Place is also an important identifier which facilitates the disambiguation of Places with similar names. Hence, a Place reference, denoted, Pl-Ref, can be either absolute or relative. An absolute place reference can be defined as a tuple of place name, place type and location (location is denoted here as Place Footprint, or Pl-FP): Pl-Ref-Abs = <Pl-name, Pl-type, Pl-FP> where Pl-FP refers to the actual position of the Place in space which may be faithful or approximate. On the other hand, a Place may be defined through its spatial relationship(s) to other Places. Hence, a relative Place reference could be defined as follows: Pl-Rlv = <Spatial Relation, Pl-name, Pl-type, Pl-FP>. Note that, in the latter case, the resulting Pl-FP would normally be computed using the spatial relationship in the expression. An example of an absolute Place reference is: <Eiffel Tower, Monument, $\{< x, y >\} >$. An example of relative Place reference is: <In, Zurich, City, $\{< x, y >\} >$.

A query to SPIRIT will contain one or more references to Pl-Ref. The same is true for web resources to be searched by SPIRIT. The process of query interpretation would result in the identification of the geographic content of the query as defined by the Pl-Ref(s) it is referring to, and similarly the process of (semantic and spatial) metadata extraction in web documents would result in the identification of the geographic content of the document as defined by its contained Pl-Ref(s).

Hence, the geographic content of a query, denoted, $Query_{GC}$ is defined as a set of Place reference expressions, namely, $Query_{GC} = \{$Pl-Ref$\}$. The geometric

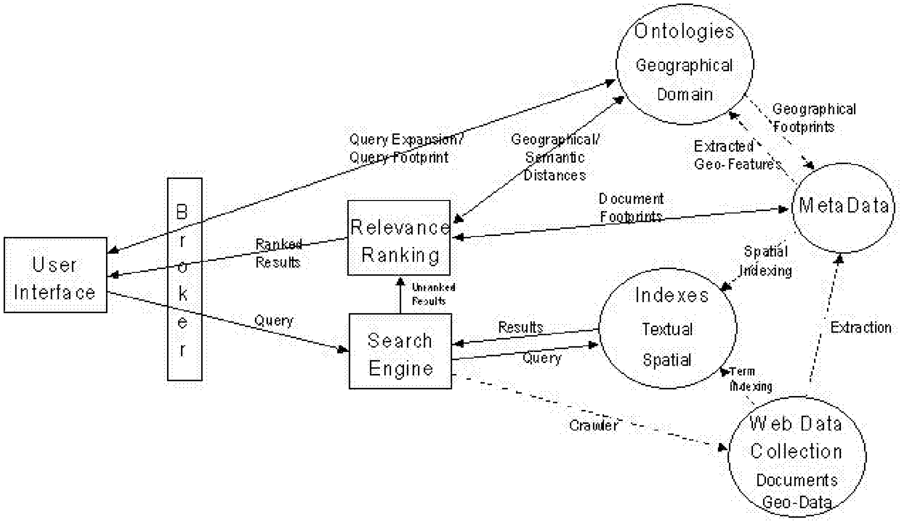

Fig. 1. The role of the geo-ontology as a component of a spatially-aware search engine

footprint of a query could be defined as a function of the footprints of its associated Pl-Ref(s), namely, $Query_{FP} = \{FP(Query_{GC})\}$. Similarly, the geographic content of a document, denoted, Doc_{GC} is defined as a set of Place reference expressions associated with the resource, namely, $Doc_{GC} = \{\text{Pl-Ref}\}$. The geometric footprint of a document could be defined as a function of the footprints of its associated Pl-Ref(s), namely, $Doc_{FP} = FP(Doc_{GC})$.

There are four main areas of application of the geo-ontology in a search engine which are 1) user's query interpretation 2) system query formulation 3) metadata extraction; 4) relevance ranking. These are described below. Figure 1 gives an overview of a possible architecture for SPIRIT and illustrates the central role of the geo-ontology.

User query interpretation. When a place name is employed in a user query, a geo-ontology will serve several purposes. It will facilitate disambiguation of the place name in the event of there being more than one place with the given name. It will also enable graphical feedback of the $Query_{FP}$. The user could then be given the option of accepting or revising the interpretation of the extent of the location. The ontology will also be able to generate alternative names, including historical variants, which the user could accept or reject according to their interests.

Domain-specific ontologies could be used to expand non-geographical terms to include synonyms. In the event of the subject (i.e. the something element) of a query being itself a place type then the place type ontology could also be used to generate similar terms for purposes of query expansion.

Metadata extraction. Ontologies could be used to identify the presence of place names, spatial qualifiers and domain-specific terminology in a free text document. If the geographical terminology was regarded as characterising the geographical context of the document, then the footprints of the respective places could be used to generate a document footprint or set of footprints that were associated with the document. This footprint metadata could be stored in the search engine database, or as metadata that could be attached to the original document using an annotation tool. The metadata might also include the textual place names extracted from the document in combination with the concept terms (or subjects) that were associated with them.

System query formulation. The geo-ontology could be used to generate alternative names and spatially associated names (according to spatial relationships such as inside, near or north of), which could in principle be included in a query expression to a text-based query processor. Alternatively, or as well, the ontology could be used to generate $Query_{FP(s)}$, as indicated above, which may be used to access a spatially indexed database of web document metadata. Thus all documents whose own footprint intersected the query footprint could be retrieved prior to being filtered according to the textual query terms. Equally it could be that text-indexed search preceded spatial filtering (again based on the query footprint).

Relevance ranking. A geographical ontology will provide the potential for geographical relevance ranking that might be combined with non-geographical ranking. The footprints associated with documents could be used to measure the distance between the document and the query footprint in geographic coordinate space. In the case of queries that used a directional qualifier the document footprint could be used to assess geometric similarity with the interpretation of the user's query footprint, according to whether it was near its core or its periphery. It would also be possible to use other aspects of the structure of geographic space for purposes of ranking. Thus for example the similarity of the query footprint and the document footprint might be regarded as a function of the parent (containing or overlapping) regions that they had in common, and those that were non-common [Ala01].

3 Related Research and Resources

The most often cited geographical thesaurus is the Getty Information Institute's Thesaurus of Geographic Names (TGN) [Get02] which is a specialisation of the general thesaurus model. For each place name the TGN maintains a unique id, a set of place types taken from the Art and Architecture Thesaurus (AAT), alternative versions of the name, its containing administrative region, a footprint in the form of a point in latitude and longitude, and notes on the sources of information. Gazetteers also constitute geographic vocabularies but some of them are very limited with regard to their semantic richness. Typically a gazetteer may encode just a single name, a single place type, a point-based footprint and a parent administrative area. As such they constitute fairly crude terminological ontologies. A recent development in the realm of gazetteers is the Alexandria Digital Library (ADL) gazetteer content standard [ADL02] which supports many types of place name metadata that may be represented either as core or optional information. This provides for a relatively rich description of place, but unlike a thesaural model there is no requirement to encode hierarchical relationships.

Recently the Open GIS Consortium has been developing a Web Gazetteer Service standard for distributed access to gazetteers. Its gazetteer model is based on location instances as defined in ISO DIS 19112 [Iso02], which are related to each other via thesaural hierarchical relationships to parent and child location instances.

In [MSS01], experiments are reported on user cognition of basic concepts in a geographic ontology, which revealed a difference in the interpretation of synonymous concepts of geographic feature and geographic object. In [CEFM01], a study of ontological specification of remotely sensed images is reported, which highlights the requirements for geographic image ontologies and proposes a representation framework for modelling them.

In adopting the term geographical ontology it is intended that more formal ontology models are designed, with a view to exploiting the automatic reasoning that it will facilitate. This move toward formalisation is reflected later in this paper in the use of the ontology language DAML/OIL [Hor00] to encode the specified ontology design. The language, and its successor OWL [W3c02], is associated with various editing and reasoning tools that can be used in defining and maintaining the ontology. An interesting issue in the design of the geo-ontology is to determine the most appropriate set of spatial relationships that might be encoded between places and the appropriate balance between the use of pre-computed spatial relationships between places and the on-line computation of relationships using the geometric footprint. With regard to some of the prominent existing geographical ontologies, the TGN design is limited by the use of only a point form footprint and the restriction to only hierarchical relations between places. The OGC Web Gazetteer Service model is also limited by the use of only hierarchical thesaural relationships between locations. The ADL may hold the potential for forming the basis of a more versatile geographical ontology, provided appropriate relationships between places are defined and used in addition to those specified in the published documentation. In all cases there is considerable scope to experiment with computational geometric and spatial reasoning techniques to exploit the stored place name information for purposes of effective information retrieval.

The focus of this work concerns the modelling of geographic place. An aspect of this process is the modelling of place types. There is also an interest in modelling the terminology of one or more application areas or domains that we may use for evaluation. Here we are referring to the something aspect of a query. It is expected that the modelling of place types and domain-specific terminology can be accomplished using conventional thesaural methods, i.e. without the need to introduce specialised types of relationships and category attributes. Thus equivalent terms or synonyms are represented via USE and USE-FOR relations. Hierarchical relations whether generic (is-a) or meronymic (part-of) are represented with Broader Term (BT) and Narrower Term (NT) relations, though it is appropriate to distinguish between these hierarchical types. If required, other associations between terms that belong to different classification groups or facets can be represented with Related Term (RT) relationships.

4 Design Issues and Considerations

In this section, factors are identified which should be taken into account in designing the geo-ontology. A typology of possible queries that may be issued to the extended search engine is identified. This is followed by a discussion on design issues related to the various elements of a query and other specific maintenance issues.

4.1 A Typology of Possible Queries

In this section, the possible types of queries that an geographically aware search engine is expected to handle are identified. In what follows, a set of atomic query expressions is first identified which can then be used to build more complex queries and scenarios. A basic query expression will consist of a reference to:

- A Place Name, or,
- An aspatial Entity with a Relationship to a Place Name[1], or,
- An aspatial Entity with a Spatial Relationship to a Place Name, or,
- A Place Name with a Spatial Relationship to a Place Name, or,
- A Place Type with a Spatial Relationship to a Place Name, or,
- A Place Type with a Spatial Relationship to a Place Type.

A Place Name is an actual name of a geographic object, e.g. Hannover. Aspatial entities are general non-geographic objects, which may correspond to a physical or an abstract theme, subject or activity, e.g. a person, a publisher, a holiday, etc. A Relationship is an instance of an aspatial, semantic, relationship which may exist between concepts in a conceptual data model, in particular, the is-related-to relationship. A Spatial Relationship is an instance of a relationship between any types of objects in space, e.g. inside, contains and near-to. A Place Type corresponds to a class of Place Names, e.g. City, Town, River and Restaurant. In what follows, Pl-name is used to denote a Place Name, SRel is used to denote a Spatial Relation, Pl-type is used to denote a Place Type, and AS-entity is used to denote an aspatial entity and AS-Rel is used to denote a non-spatial (semantic) relation. The set of basic types of queries is listed in table 1.

The above are atomic query expressions that may be used to generate more complex query expressions using binary logic operators and spatial operators.

[1] An aspatial Entity is a non-geographic entity

Table 1. A list of possible basic query types to be handled by SPIRIT

Query Type	Example
Find <Pl-name>	Zurich
Find <Pl-name SRel Pl-name>	City Hall IN Paris Barry NEAR Cardiff
Find <AS-entity AS-Rel Pl-name>	Books on-the-subject-of (About) Taipei
Find <AS-entity SRel Pl-name>	Scottish Dance groups based IN or NEAR Edinburgh
Find <Pl-type SRel Pl-name>	Hotels NEAR Paris Big Cities IN Japan Rented accommodation NEAR Brussels
Find <AS-entity SRel Pl-type>	Database conferences NEAR Sunny Beaches
Find <AS-entity AS-Rel Pl-type>	Presidents of countries Books on the subject of rivers
Find <Pl-type SRel Pl-type>	Hotels NEAR Airports Airports NEAR Big Cities

Hence, in processing the complex queries, atomic expressions are first extracted that correspond to one of the forms above. The following are examples of such queries. In what follows, Op is used to denote a logical operator, e.g. AND, OR, NOT.

```
– Find <(Pl-name Op Pl-name) SRel Pl-name>
  Atomic expressions:
     Find <Pl-name SRel Pl-name> OP Find <Pl-name SRel Pl-name>
  Example:
     Shoreditch and Stratford IN London
– Find <Pl-type SRel Pl-type SRel Pl-name>
  Atomic expressions:
     Find <Pl-type SRel Pl-name> OP <Pl-type SRel Pl-name>
  Examples:
     Hotels NEAR Airports AND IN Washington
     Hotels IN Munich AND Hotels within a short walk from
     Munich's Main Station.
```

4.2 Design Considerations Regarding the Primary Query Elements

From the above, it can be seen that the main query constructs are: a Place Name, an aspatial Entity, a Place Type, a Relation and a Spatial Relation. In this section, an investigation of the issues related to the above constructs is presented.

Place Name. A place name is used to reference a particular geographic object. There may exist different names and variations of names for the same geographic object, e.g. Treforest and Trefforest. The ontology is expected to store as many as possible Place names and known alternatives, including historic names. Ideally, Place names in different languages should also be stored. Places may be referred to that may have no formal definition, such as the south of France, the Midlands or the Rocky Mountains. There are two ways to define such imprecise regions. The Places, and their associated locations, may be pre-recognised and stored explicitly in the ontology, or an interactive dialogue with the user needs to be carried out at the interface to clarify the location and/or extent of those objects. Indeed, both scenarios may be used together to confirm the correspondence between the stored and intended definitions.

Place Location. The ontology must associate a geometric footprint with all the stored geographic objects. The footprint may be approximate, e.g. a representative (centre) point or a bounding box, or more detailed, e.g. approximate shapes, or exact, i.e. a faithful representation of the object geometry. This decision has direct storage implications and the benefits and limitations of the choice need to be carefully studied. Also, more than one type of geometry may be associated with the same object. For example, a region may be associated with both a representative point and a polygon, which may itself be an approximation of the actual shape.

Place Address. The use of an address is a common form of reference to the location of a geographic object. A street name is considered to be a type of Place name as defined above. A postcode or zip-code is normally a part of an address used commonly to group sets of individual addresses or places. The codes may also serve as a constraint on query location during query interpretation and expansion.

Spatial Relations. It is desirable that an ontology of spatial relations be defined in the system to allow for the interpretation of terms given at the interface. The ontology of relations should cater for the different types of spatial relations possible, namely, topological, proximal, directional and size in both quantitative and qualitative expressions. A number of explicit types of spatial relationships between geographic objects may be stored in the ontology facilitating the interpretation and expansion of query expressions by direct matching and reasoning over spatial relations.

Coordinate systems. In view of the objective of a global geographical ontology it would appear desirable to employ a single geometric coordinate system that is global in coverage. The obvious choice is therefore the use of latitude and longitude ("geographical" as opposed to "grid") coordinates. In practice latitude and longitude are not unique as they are based on a specific geodetic datum, which denotes the dimensions of a spheroid that is used to approximate the shape of the Earth. Assuming that the geo-ontology employs geographical coordinates on a specified datum, then all geometric calculations such as distance, orientation and area could be performed directly on the surface of the spheroid.

An alternative approach would be to store coordinates on the various local grid systems (e.g. the UK National Grid) used by the national mapping agencies or other data providers. This might be more efficient relative to spherical (geographical) coordinates for calculations that were confined to the geographic zone of the respective grid system, but would cause problems whenever interzone calculations were required (these could be done via intermediate coordinate transformations). In conclusion the simplest approach to adopt in the first place appears to be to use geographical coordinates on a specified datum. Alternative approaches could then be considered at the implementation stage.

Time. A characteristic of all geographical places is that they are embedded not just in space but also in time. Settlements and other topographic features have some time of origin (though it may not always be known) and in some cases dissolution. The names of many places have changed over time and geopolitical and natural environmental boundaries are subject to appearance, disappearance or re-location over time. Full support for spatio-temporal information is highly desirable in a geographical ontology for purposes of information retrieval, but it is also demanding. On the assumption that some of the data resources for the ontology may have some temporal data relating for example to the date of establishment or duration of a place name it seems appropriate to support the storage of such data with a view to developing procedures for their exploitation. It should be noted that the introduction of support for time would extend the typology of possible queries presented in section 4.1.

Language. The importance of multi-lingual support for geographical information retrieval is highly desirable. Some limited support for encoding alternative language versions of names is relatively simple to provide in the ontology design. Full support for a multi-lingual search engine is beyond the scope of this research.

Semantic and Geometric Generalisation. It is well known that geographic data may be represented at multiple levels of generalisation. One aspect of generalisation concerns the level of detail with which a specific object is represented. Thus the areal extent of a settlement could be represented for example by a polygonal boundary with detailed sinuosity, representing a large proportion of the humanly perceptible detail. Alternatively it could be represented by a coarsely simplified polygon, a bounding rectangle or simply a representative point or centroid. These types of generalisation are examples of geometric generalisation. For reasons of data availability and usefulness, it would be impractical and also unnecessary to encode all geographic data in a geo-ontology at the highest levels of geometric detail. However, in order for the geo-ontology to fulfill its roles, it is desirable that it can encode geographic data, especially the geometric data, with sufficient geometric detail. For example, encoding the footprint with a single coordinate point might be adequate for a feature which is of type village, of relatively small areal extent, but might not be sufficient for a feature which is of type country, especially when the query expansion, relevance ranking and spatial index are considered. The same argument applies in the case of the semantic level of detail, e.g. geographical information may be recorded in high level classes, e.g. countries, cities, primary roads, etc. as well as lower levels of detail, e.g. counties and towns, side streets, etc. For the ontology to be useful, it should be able to encode geographic data at multiple levels of semantic generalisation.

Explicit vs implicit maintenance of spatial data. It has been noted above that there are several types of spatial information, ranging from coordinate-based geometry, in the form of points, lines, areas and volumes, to the spatial relationships categorised as topology, proximity, orientation and size. The question arises as to what is an appropriate balance between explicit storage of spatial information and the use of online procedures to derive or deduce information from what is stored.

Because of the high storage costs of detailed geometry and the associated computational costs, there is an argument for explicit storage of topological relationships between neighbouring objects. Topological relationships between non-neighbouring objects can often be deduced reliably with spatial reasoning rules. From a computational point of view there might be a case for explicit storage of proximal, orientation and size relationships, at least between neighbouring objects. Clearly this would result in a significant storage overhead. It is also the case however that logical deduction of these relationships (apart from size) between non-neighbouring objects cannot be performed reliably, due to the often imprecise nature of the relations. The cost of explicit storage of all possible such relationships would be combinatorially explosive. Following the above considerations it appears reasonable therefore to decide initially to store geometry at variable levels of detail in addition to storing topological relationships between neighbouring spatial objects. The balance between online computation and explicit storage of other spatial relations and of more detailed geometry will be examined in future work.

4.3 Checking and Maintaining the Integrity of the Geo-ontology

Maintaining the consistency and the integrity of the geo-ontology is essential for supporting the correct functionality of the search engine and for ensuring the viability and the quality of the search results produced. Maintenance tools are therefore needed for the initial set-up and building of the geo-ontolgy. Also, the ontology is expected to be updated and extended frequently, as new classes of geographic objects and new instances of geographic features are identified. Hence, such maintenance tools must be supported as an integral part of the whole system. Examples of possible maintenance tasks needed when building the ontology base are:

- Ensuring that all mandatory relationships are satisfied, e.g. that every geographic feature belongs to at least one geographic type and has at least one footprint.
- If a feature is involved in a containment relationship or an overlap relationship, where it is the parent, then it must have at least one extended footprint, i.e. a polyline or a polygon.
- A polygon footprint with more than two points, must have at least four points, with the first point being equal to the last point.
- For two features in a containment relationship, the bounding box of the child must be enclosed in the bounding box of the parent.
- For two features in an overlap relationship, the bounding boxes of both must intersect.

Note that the assertion of spatial relationships between geographic features needs to be based on detailed geometric representations, as far as possible, even if such representations are not stored subsequently. Although this may be an expensive process initially, it will be limited, as explicit encoding of spatial relations will be limited to parent-child relationships, and also constrained by feature types. Maintenance tools are needed for checking the consistency of stored spatial relations. Such tools can make use of spatial reasoning techniques, e.g. composition tables [Ege89,BA01], to implement rules for constraining the propagation and derivation of such relationships. Spatial reasoning techniques exploit the inherent properties of relations, such as transitivity and symmetry. Examples of rules for topological relationships include:

- $contain(x, y), contain(y, z) \rightarrow contain(x, z)$
- $inside(x, y), meet(y, z) \rightarrow disjoint(x, z)$
- $meet(x, y), inside(y, z) \rightarrow inside(x, z)$ or $covered - by(x, z)$ or $overlap(x, z)$
  Knowledge of size relationships can further enhance the reasoning process; for example, the last rule can be modified with the knowledge that the size of x is larger than the size of z as follows:
- $meet(x, y), inside(y, z), larger(x, z) \rightarrow overlap(x, z)$

5 Conceptual Design of the Geo-ontology

The geo-ontology proposed here is composed of three components , namely, a geographic feature ontology, a geographic type ontology and a spatial relation ontology. A feature type is associated with a feature type name, and a resource from which the feature type is derived. Feature types can be related by normal thesaural relationships. The base schema for the geographic feature ontology is illustrated in Fig. 2. For each geographic feature, it encodes

1. One and only one Feature-ID, which uniquely identifies a geographical feature
2. One and only one Standard-Name, which specifies a name by which a geographical feature is best known. A name is associated with the date when it was used and the language in which it is specified, as well as a resource which contributes the information.

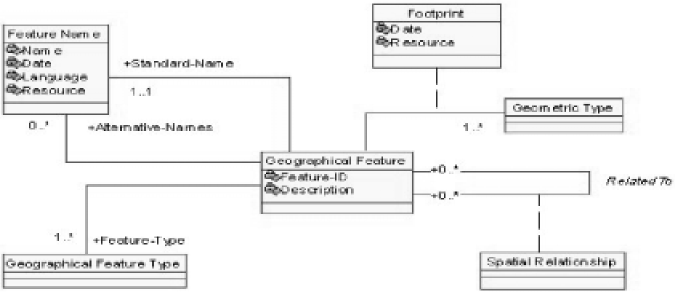

Fig. 2. Base Schema of the geographic feature ontology

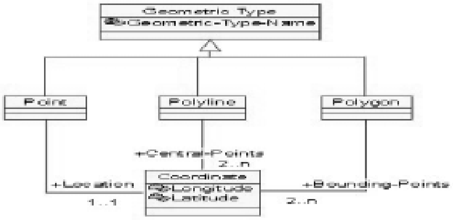

Fig. 3. Basic types of footprints in the geo-ontology

Table 2. Synonymous spatial relationship terms

Spatial Relation	Synonym
Beside	(alongside, next-to)
Near	(close, next-to)
Overlap	(intersect, cross)
Inside	(in, contained-in, within)
Disjoint	(outside, not-connected)
Touch	(adjacent, on the boundary of, next, side by side, close, abutting, adjoining, bordering, contiguous, neighbouring) [wordnet]

3. Zero or more Alternative-Names.
4. One or more Feature-Types as defined in geographical feature type ontology.
5. One or more spatial Footprint. Basic footprints to be supported are points, polylines or polygons, as shown in Fig. 3.
6. Description, a short narrative description of the geographical feature.
7. Zero or more Spatial Relationships, representing how geographical features are related. An ontology of spatial relationships is supported as shown in Fig. 4. Some explicit spatial relationship types may also be supported, e.g. adjacency and containment as shown in the figure. Some examples of synonymous terms to be encoded in the ontology are shown table 2.

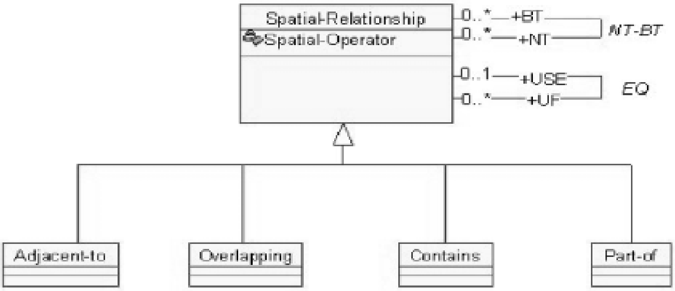

Fig. 4. Ontology of Spatial Relationships

5.1 Spatial Ontology Access Operations

A set of operations on the geo-ontology are defined to facilitate the manipulation and derivation of the stored information. A sample of the set of basic operations is given below.

getFeature(L1, L2). This operation retrieves geographic features using a constraint L1 and returns a set of properties of the feature L2. For example, getFeature(<Feature-Type.Name=city>, <Footprint>) will displays the footprints of features of which the feature type is of city, and getFeature(<Standard-Name.Name=Bremen>, <Identifier, Feature-Type>) displays the Identifiers and the Feature-Types of the feature of which the standard name is Bremen.

getFeatureType(L1). This operation retrieves geographic feature types using an optional constraint L1. For example, getFeatureType(<Feature-Type.NT= city>) displays the feature types of which the narrow term is city, and getFeatureType(<Feature-Type.USE=city>) displays the feature types of which city is used as the preferred term.

getHierarchy (L1, L2, L3). This operation retrieves features in containment hierarchies. This is achieved by transitive traversal of the part-of or contains relationships of the concerned feature to derive the parent or children of that feature. For example, getHierarchy (<high>, <Feature-ID=01079>, <level=3>) retrieves the 3rd level parents of the feature whose identifier is 01079.

6 Tools for Encoding the Geo-ontology

Various ontology-representation languages exist and can be used for modelling the geo-ontology. They differ in the degree of formality, ranging from semi-informal, e.g the text version of ”enterprise ontologies” [UK89], to semi-formal languages, e.g. KIF [GF92]. A language for encoding the geo-ontology should aim to satisfy the following.

– Be compatible with existing Web standards, such as XML, RDF, RDFS, in order to facilitate information exchange with other components of the search engine;
– Have adequate expressive power to represent the geo-ontology, and be flexible enough to allow the extension of the ontology schema;
– Be formally specified to facilitate automated reasoning to support query expansion tasks;
– Have sufficient tools to support the updating and maintenance of the ontology.

In the rest of this section, a variety of ontology languages are reviewed and their suitability for encoding the geo-ontology is discussed. Various languages have been used in the literature for specifying ontologies. Some languages are based on XML, such as XOL [KCT99], SHOE [LH00], OML citeKent-26, RDF [LW99] etc, some are based on Description Logics (DLs), e.g. KIF[GF92], CycL[cyc02], CLASSIC [BBMR89], and some are built based on both of XML and DLs, e.g. OIL [Hor00], DAML+OIL, OWL [W3c02].

6.1 DL-Based Ontology Languages

Description logics (DLs) are knowledge representation languages for expressing knowledge about concepts and concept hierarchies. They can be seen as sub-languages of predicate logic. The basic building blocks of DLs are concepts, roles and individuals. Concepts describe the common properties of a collection of individuals and can be considered as unary predicates which are interpreted as sets of objects. Roles are interpreted as binary relations between objects. Each DL language defines also a number of language constructs (such as intersection, union, role quantification, etc.) that can be used to define new concepts and roles. For instance, the following DL expression, based on the language CLASSIC [BBMR89], represents the constraints on the geographic feature; "every geographic feature can have one and only one identifier, at least one feature name, and at least one footprint which is one of the following types: point, polyline and polygon".

```
feature

⊒

(AND  (AT-LEAST, 1, identifier)
                            (AT-MOST, 1, identifier)
                            (AT-LEAST, 1, name)
                            (ATLEAST, 1, footprint)
                            (ALL footprint  (ONEOF  point, polyline,
                                                           polygon)))
```

The potential of DLs lies in their expressiveness and their associated decidable and efficient inference procedures. Limitations of DL languages include their incompatibility with existing web languages, which makes it hard for ontologies represented in them to be shared and exchanged. Also, the tools developed for DLs often do not integrate well with existing web tools, which makes it difficult to import, export and access the ontologies specified in them.

6.2 XML-Based Ontology Languages

Notable examples include RDF [LW99], RDFS [BR99], XOL [KCT99], SHOE [LH00] etc. They restrict XML by providing a set of primitives to express knowledge in a standardized manner to facilitate machine-understanding. A relevant language of this group is GML (Geographical Markup Language) [OGC02], which is proposed by OGC for specifying geographic information, including both spatial and non-spatial properties of geographical features. The basic idea of GML is to provide an open, vendor-neutral framework for the definition of geospatial application schema. It supports the description of geographic data by providing a set of base types and structures and allowing an application to declare the actual feature types and property types of interest by extending basic types in GML. For example, the following code defines a geographic feature type Mountain, which extends base feature type AbstractFeatureType provided by GML, and a specific property elevation is defined for it.

```
<complexType name="Mountain">
      <complexContent>
             <extension base="gml:AbstractFeatureType">
                    <sequence>
                         <element name="elevation" type="Real"/>
                    </sequence>
             </extension>
      </complexContent>
</complexType>
```

Using the above definition, we can encode information for Everest as follows:

```
<Mountain>
     <gml:description>World's highest mountain </gml:description>
     <gml:name>Everest</gml:name>
     <elevation>8850</elevation>
</Mountain>
```

Unlike DLs, the XML-based languages are relatively compatible with existing Web standards since many of them are designed to facilitate machine-understandable web representation. However, the main limitation of this group of languages is the lack of supporting tools necessary for the maintenance of the geo-ontology.

6.3 DL+XML-Based Ontology Languages

Another stream of ontology languages are built on top of both XML and DLs, and thus they are compatible with existing Web standards and at the same time retain the formal semantics and reasoning services provided by DLs. Examples of such languages include OIL [Hor00], DAML-ONT [DAR02], DAML+OIL and OWL [W3c02].

DAML-ONT is developed by the DARPA project [DAR02] and it inherits many aspects from OIL, and the capabilities of the two languages are relatively similar. DAML+OIL layers on top of RDFS and combines the efforts from OIL

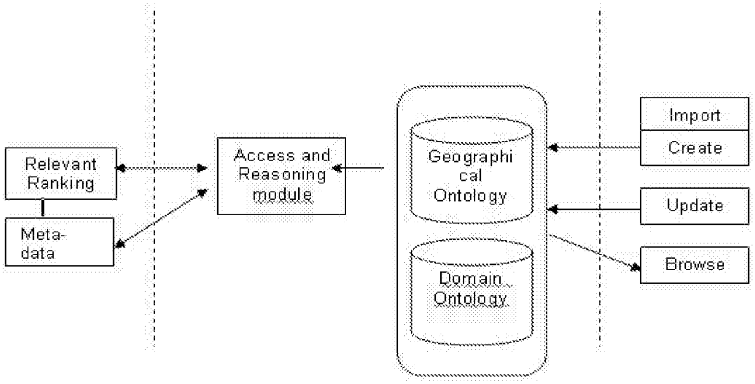

Fig. 5. SPIRIT ontology implementation architecture

and DAML-ONT. It Inherits many ontological primitives from RDFS, e.g. subclass, range, domain, and adds a much richer set of primitives from OIL and DAML-ONT, e.g. transitivity, cardinality, and it allows assertion of axioms. For example, we can specify in DAML+OIL the axiom “A big city is a city which has a population greater than 5 million” as the follows:

```
<daml:Class rdf:ID="BigCity">
   <rdfs:label> Big City</rdfs:label>
   <rdfs:subClassOf  rdfs:resource=#City>
   <rdfs:subClassOf
         <daml:Restriction >
               <daml:onProperty rdf:resource="#population"/>
               <daml:hasClass  rdf:resource=#over5m/>
         </daml:Restriction>
  </rdfs:subClassOf>
</daml:Class>
```

Derived from DAML+OIL, OWL is released by W3C as a semantic markup language for publishing and sharing ontologies on the WWW. It aims to overcome various problems with DAML+OIL, for example, problems with syntax and semantics, mainly related to relationship with RDF.

Ontology Implementation. Figure 5 sketches the different components required for implementing the geo-ontology within SPIRIT. As shown in the figure, the ontology repository consists of a geo-ontology and a domain specific ontology. A module for accessing and reasoning over the ontology is built on top and acts as an interface to the other components of SPIRIT. Maintenance tools for importing, creating, updating and browsing the ontology are also required.

7 Conclusions and Future Work

This paper identifies and describes the central role of a geographic ontology in the development of a spatially-aware web search engine. The functionality associated with it is directly associated with the three areas of the user interface, metadata extraction and relevance ranking. The ontology may be used to disambiguate the place name expression in user queries and subsequently generate alternative place names and associated place names for query expansion. The geo-ontology could also be used to identify the presence of place names, spatial qualifiers and domain-specific terminology in a free text document which may be used to annotate those documents in the web repository. Geographical relevance ranking in the search engine needs to use the geo-ontology for the derivation of footprints and for the application of similarity measures over the query footprints and the footprints associated with the web resources. The paper also introduces a typology of queries over places and place types which the search engine is expected to handle. Various requirements which influence the design of the geo-ontology are reviewed. Maintenance issues for ensuring the consistency of the ontology are also discussed. A base ontology model is then proposed for the geographic ontology which aims to provide a foundation for subsequent implementation and experimentation. The paper also reviews various ontology languages available for expressing ontologies, namely, DLs, XML-based languages, and the combination of both, and gives examples of their use for encoding the geo-ontology.

Acknowledgment. This work is done within the scope of the SPIRIT project. The project is funded by EC Framework V IST programme under the Semantic Web Technologies action line, for which their support is gratefully acknowledged.

References

[ADL02] ADL Gazetteer Content Standard http://alexandria.sdc.ucsb.edu/ lhill/adlgaz/, 2002.

[Ala01] Alani, H. and Tudhope, D. and Jones, C.B. Geographical information retrieval with ontologies of place. In *Spatial Information Theory Foundations of Geographic Information Science, COSIT 2001*, volume Lecture Notes in Computer Science 2205, pages 323–335, 2001.

[BA01] El-Geresy. B.A. and A.I. Abdelmoty. Towards a general theory for qualitative space. In *Proceedings of the Thirteenth Int. Conf. On Tools with Artificial Intelligence*, pages 111–120, 2001.

[BBMR89] A. Borgida, R.J. Brachman, D.L. McGuinness, and A.L. Resnick. Classic: A structural data model for objects. In *SIGMOD Conference*, pages 58–67, 1989.

[BR99] D. Brickley and Guha R.V. Resource description framework (rdf) schema specification – www.w3.org/tr/pr-rdf-schema, 1999.

[CEFM01] G. Camara, M.J. Egenhofer, F. Fonseca, and A.M. Monteiro. What's in an image? In *COSIT'01. Lecture Notes in Computer Science*, volume 2205, pages 474–487, 2001.

[cyc02] Cycorp, the syntax of cycl – http://www.cyc.com/cycl.html, 2002.

[DAR02] Darpa, the darpa agent markup language homepage http://www.daml.org/, 2002.

[Ege89] M.J. Egenhofer. A formal definition of binary topological relationships. In *International Conference on Foundations of Data Organization and Algorithms*, pages 457–472, 1989.

[Get02] Getty, getty thesaurus of geographic names http://www.getty.edu/research/tools/vocabulary/tgn/, 2002.

[GF92] M.R. Genesereth and R.E Fikes. *Knowledge Interchange Format, version 3.0 reference manual*, 1992.

[Hor00] I. Horrocks. OIL in a Nutshell. In *ECAI Workshop on Application of Ontologies and PSMs*, 2000.

[Iso02] Iso19112, geographic information – spatial referencing by geographic identifiers, 2002.

[KCT99] R. Karp, V. Chaudhri, and J. Thomere. Xol: An xml-based ontology exchange language – www.ai.sri.com/ pkarp/xol, 1999.

[LH00] S. Luke and J. Heflin. Shoe 1.01 proposed specification – www.daml.org/ 2000/12/reference.html, www.cs.umd.edu/projects/plus/shoe/spec1., 2000.

[LW99] O. Lassila and R. Webick. Resource description framework (rdf) model and syntax specification – www.w3.org/tr/pr-rdf-syntax, 1999.

[MSS01] D.M. Mark, A. Skupin, and A. Smith. Ontological distinctions in the geographic domain. In *COSIT'01. Lecture Notes in Computer Science*, volume 2205, pages 488–502. Springer Verlag, 2001.

[OGC02] Opengis geography markup language (gml) implementation specification – http://www.opengis.org/techno/implementation.htm, 2002.

[UK89] M. Uschold and M. King. The Enterprise Ontology. *Knowledge Engineering Review*, 13(1):31–89, 1989.

[W3c02] W3c, owl web ontology language – http://xml.coverpages.org/owl.html, 2002.