International Journal of Geographical Information Science
Vol. 00, No. 00, Month 200x, 1–24

The Role of Ontology in Improving Gazetteer Interaction

Krzysztof Janowicz* & Carsten Keßler

Institute for Geoinformatics, University of Münster, Germany

(*Received 00 Month 200x; in final form 00 Month 200x*)

Gazetteers are more than basic place name directories containing names and locations for named geographic places. Most of them contain additional information, including a categorization of gazetteer entries using a typing scheme. This paper focuses on the nature of these categorization schemes. We argue that gazetteers can benefit from an ontological approach to typing schemes, providing a formalization that will better support gazetteer applications, maintenance, interoperability, and semi-automatic feature annotation. We discuss the process of developing such an ontology as a modification of an existing feature type thesaurus; the difficulties in mapping from thesauri to ontologies are described in detail. To demonstrate the benefits of a categorization based on ontologies, a new gazetteer Web (and programming) interface is introduced and the impact on gazetteer interoperability is discussed.

1 Introduction and Motivation

Gazetteers are place name directories containing names, spatial references, feature types and additional information for named geographic places. They are key components of all georeferenced information systems, including GIScience applications in many diverse fields of knowledge, Web-based mapping services, and the emerging Web 2.0. A typical use case for gazetteers is information retrieval where queries can be based on place names and coordinates. They are central to the process of geoparsing where references to geographic locations by place name are recognized in text strings and converted to coordinate references. Gazetteers are also components of complex reasoning services such as the identity assumption service for historical places discussed by Janowicz (2006b). From an information theoretic point of view, a gazetteer is defined as a triple (N, F, T) where N corresponds to one or more place names, F represents one or more geographic footprints (i.e., spatial locations) and T is the type of the described feature (i.e., place) (see Hill 2006). In the context of gazetteers, a feature is a real world entity. The feature type which is selected from a typing scheme or ontology[1] is used for feature categorization. A named geographic place is an abstract entity defined to refer to a physical region (extent) in space and categorized (typed) according to commonly agreed upon characteristics. Place is a social concept of interest for a particular community during a certain time span. Its name is a symbol used for communication.

Categorization is a central cognitive process. This paper focuses on two reasons for the categorization of places: communication and cognition. Categorizing into types improves communication about places with which at least one communication partner is unfamiliar, for example when giving directions such as "follow the *path* along the *river* up to the *bridge*, then turn right towards the *market place*". Moreover, typing is the key to prediction, reasoning and decision making which all require an abstraction from entity to type level. What humans experience as a place is, in fact, the set of perceivable characteristics of the region in space the place refers to by its type and name (see also Casati and Varzi 1999). This includes the surface and texture of the physical region of earth, man-made entities such as buildings, and knowledge obtained from maps, books, and other information sources. Beyond those perceivable characteristics, places may also be typed by convention, such as administrative areas. The referenced region (or entity) can also be described in relation to other regions or entities, such as "East Frisia is a coastal region in the northwest of Lower Saxony". The definition of place as a mental handle pointing to real world regions (or entities)

*Corresponding author. Email: janowicz@uni-muenster.de

[1]The concepts we specify in ontologies are representations of the concepts in our mind, which should not be confused.

International Journal of Geographical Information Science
ISSN 1365-8816 print / ISSN 1362-3087 online © 200x Taylor & Francis
http://www.tandf.co.uk/journals
DOI: 10.1080/1365881YYxxxxxxxx

is independent of a specific name or an affixed and stable portion of space. Because the name functions as a symbol for communication, a particular place can be referred to by various names by different people and in different ways through time or by placeholders such as "Anyshire". The spatial extent referred to by the place name may vary over time or be known only in a general sense. The clear distinction between real world and reference also helps to explain how places can disappear without causing inconsistency. One can argue that a place no longer exists when there is no human left who is aware of this place. A place, such as a temporal Normand settlement, moves when the perceivable characteristics move (as opposed to the region on the earth's surface)[2].

The partition into names, footprints and types corresponds to the minimum definition of a gazetteer entry, a subset of the full set of descriptive elements that includes details such as spatio-temporal history[3]. The name of a place is called its textual reference, while the footprint is called the spatial reference. Thus, a gazetteer supports at least two functions. First, it maps between place names and respective footprints: N⟶F; and second, between names and types: N⟶T. Several online gazetteer services support queries by place name, footprint, and type via a Web page or through an application programming interface (API). These access functionalities are integrated into other online services; for example, to translate a place-name query into a footprint query in order to search data sets where only spatial access is supported. A long term vision of gazetteer research is focusing on the development of a distributed local-responsibility service infrastructure instead of a single world gazetteer. Such an infrastructure can be compared to the Domain Name Service (DNS) which maps hostnames on the internet to their IP addresses. In such a gazetteer network, each gazetteer offers lookup for places within its spatial and thematic scope. If the gazetteer cannot answer a request, it redirects the query to a higher level gazetteer which decides whether it or another gazetteer can resolve the query. The underlying idea is that gazetteers should contain and maintain data of interest for the community running the service. This ensures that the stored data is accurate, complete, and up-to-date.

A distributed gazetteer infrastructure raises several challenges for effective cross-gazetteer access. For example, different names can refer to the same place or to different places and the spatial footprints may vary from one gazetteer to the other. Footprints vary because of interpretation or because the boundaries have changed through time or simply because different types of footprints have been used, such as point versus polygon representation (see also Janée 2006 and Hastings 2008). The most problematic interoperability issue, however, is the variety of typing schemes used to categorize gazetteer entries (features). There is no common typing scheme that can be used for cross-gazetteer access because gazetteers are developed for different purposes and communities with varying thematic scopes and spatial scales. For a common feature type specification to be successful, it needs to be generic enough to form a top level for all gazetteers and extensible to allow for local type definitions. It also needs to be grounded in widely accepted definitions of common feature type categories.

In this paper, we argue that the standards-based thesaurus structure (ANSI/NISO 2005) is not a sufficient basis for feature type interoperability among divergent gazetteers; we review existing feature typing schemes to illustrate this issue. Their structure also prevents the development of enhanced Web and programming interfaces. In some cases type-lookup even leads to unexpected results. We propose the development of a feature type ontology to improve both gazetteer interoperability and reasoning capability based on feature typing. To demonstrate this, our approach is to take advantage of an existing feature typing scheme—the Alexandria Digital Library's Feature Type Thesaurus (FTT)—to create a portion of such an ontology. The steps taken to create such an ontology are discussed. Based on the proposed ontology an extended gazetteer Web interface is introduced. This interface applies subsumption and similarity based reasoning (Janowicz 2006a, Janowicz *et al.* 2007, Lutz and Klien 2006) to improve usability.

The paper consists of two parts. The first part reviews three feature type thesauri, the Alexandria Digital Gazetteer, the Getty Thesaurus of Geographic Names and GeoNames.org[4]. Section 2 describes the difficulties observed in mapping from feature type thesauri to an ontology. Then in section 3, the Web and programming interfaces of these gazetteers are examined, giving special focus to the type-lookup

[2]Finally, this leads to the question of place identity which is out of scope for this paper.
[3]For instance, the ADL Gazetteer Content Standard allows for a *Time Period Note* for names, spatial footprints and types.
[4]Which is strictly speaking rather a feature type catalog than a thesaurus.

functionality. The second part of the paper focuses on the conceptual design and implementation of a feature type ontology (section 4), giving examples from hydrography. Applying this ontology, section 5 points out how extended gazetteer interfaces can be implemented using ontology-based reasoning services such as similarity and subsumption. The implications of the new ontology and interfaces on gazetteer interoperability are discussed. Finally, section 6 presents conclusions and directions of further work.

2 An Ontological View on Feature Type Thesauri for Gazetteers

In this section we examine three well-known feature typing schemes, how they compare to an ontological approach, and the issues involved with converting these schemes—particularly, the thesauri—into an ontology. Such schemes were developed for particular gazetteer applications and not with conversion to ontologies in mind. We identify the issues in such a conversion and illustrate the difficulties of using existing thesauri as the basis of ontologies (van Assem *et al.* 2004). A distinction must be made between the thesauri examined here and the theoretical principles of thesaurus construction as described by the ISO2788 (ISO 1986) and ANSI/NISO Z39.19 (ANSI/NISO 2005) standards. We have not attempted a comparison of thesaurus construction principles versus ontology construction principles. Our examination of feature type thesauri focuses on typing schemes currently in use, the role they play in online gazetteer services, and what advantages can be realized if ontologies are used instead.

2.1 *Thesauri and Ontologies*

Thesauri are developed for different purposes than ontologies. To highlight the fundamental differences between thesauri and ontologies, we give a brief overview here without going into detail (see table 1). According to Gruber (1993), "an ontology is an explicit specification of a conceptualization" used to achieve a shared and common understanding of a particular domain of interest (see also Guarino 1998, Sowa 2000, Studer *et al.* 1998). Therefore, an ontology includes specific characteristics of a concept such that one concept is distinct from another (e.g., that a river is a flowing body of water) and these characterizations are enforced so that all subconcepts of *River* are also flowing bodies of water. Structurally, an ontology has an unlimited set of relationships, one of which is the *is-a* hierarchical relationship.

Table 1. Thesaurus versus Ontology.

Aspect	**Thesaurus**	**Ontology**
Purpose	Information retrieval & structuring	Inference & reasoning, information retrieval
Order	generic, whole-part or instance hierarchy	Is-a hierarchy
Relations	Restricted number of relations	Arbitrary number of relations
About	Terms representing concepts	Specifications of concepts
Semantics	No formal semantics	Formal semantics

Thesauri are defined as controlled vocabularies with a fixed number of relationships. These relationships are hierarchical, associative, and equivalency. The hierarchical relationships can be further specified as being generic (is-a), partitive (whole-part), or instance (describing the relation between an instance and its type). Thesaurus standards allow multiple hierarchies (i.e., a concept can occur in more than one hierarchical tree), but most thesauri use a single inheritance hierarchy to simplify maintenance and the display of the relationships. The associative relationships points to similarities between concepts that are not related hierarchically. Equivalency is used to introduce alternative terms that are used to describe the concept or a concept that is semantically equivalent within the scope of the particular thesaurus. This is why a thesaurus is called a controlled vocabulary. One term (the preferred term) is chosen to represent a concept while other possible terms (non-preferred terms) are entered as equivalent terms. These alternative terms are not part of the controlled vocabulary but are considered to be lead-in terms which lead to the appropriate controlled vocabulary term. In contrast to ontologies, thesauri do not have specific characterizations of the concepts that constrain the establishment of relationships. Thesaurus construction is guided by international guidelines but individual thesauri may not rigorously follow these principles.

Thesaurus entries can also have textual notes to explain their intended scope, provide an informal concept definition, or document when the term was added to the thesaurus. For the display of thesaurus structures, there are commonly used notations for the relationships; these are shown in table 2.

Table 2. Labels for relationships and descriptive notes.

Relationship or descriptive note	**Label**
Use / Used For	USE / UF
Broader Term / Narrower Term	BT / NT
- generic	BTG / NTG
- partitive	BTP / NTP
- instance	BTI / NTI
Related Term	RT
Scope Note	SN
Definition	DF
History Note	HN

The hierarchical relations are shared by ontologies and thesauri. The instance and whole-part relationships in thesauri can also be expressed in ontologies; the instance relationship corresponds to the ontological instance-of relationship. The whole-part relationship has no specific pre-defined counterpart but can be modeled as binary relation (see also Bittner *et al.* 2004). The associative relations in thesauri are not defined in any way that is transferable to ontologies (although the most recent thesaurus standards present common subtypes of associations). Instead, ontologies define the type of association explicitly, which allows for additional reasoning capabilities. Since the non-preferred terms of the equivalency relationship are not part of the controlled vocabulary, there is no direct correlation to their role in ontologies, however in some cases they can be thought of as so-called equivalence classes.

Summing up, one can characterize the increasing expressivity from a pure taxonomy over a thesaurus up to an ontology as follows: while a taxonomy only groups terms using the generic relationship, a thesaurus also supports partonomy, instance-class relations, non-hierarchical associations between preferred terms, and adds non-preferred terms to provide access to terminology by alternative expressions. An ontology explicitly defines all concepts and relations so that the intended scope (i.e. range and domain) and the logical implications of the relations can be validated and used for reasoning services.

For reasons of readability and influenced by Smith (2006), we categorize the difficulties in mapping from feature type thesauri to ontologies into three groups. However, most of them can be regarded as special cases of implicit or missing formal semantics.

2.2 *Representation versus Representation Language*

From an ontological point-of-view, feature type thesauri have no clear distinction between the representation of features as real world phenomena and the usage of representing symbols in the gazetteer application workflow. While some relations link features of given types to each other, other relations apply to the symbols themselves. For instance, hierarchical relations are relations about features (respectively entities) and can therefore be mapped to an is-a hierarchy within an ontology as long as the hierarchy is based on the generic relationship (BTG/NTG). In contrast, the equivalence relation cannot be mapped in this way because it holds neither between entities of the related types nor between the types themselves (as non-preferred terms are not defined types). Instead, it is a relation between concepts and lexical terms used to direct the user to the preferred terms. The same argument holds for relations specifying alternative (e.g. foreign) names for feature types or node labels.

2.3 *Ambiguous Relationships*

Examining feature typing schemes, we found that relationships, such as the generic hierarchical relation, are used in different ways by thesauri and sometimes even within the same thesaurus. When partitive relationships are used for geographic places, for example to show that Berlin is part of Germany, the

ontology conversion process should map this to contained/contains relations holding between entities of certain geographic feature types (such as between City and Country). However, it is not possible to do this automatically if both generic and partitive relations are used within the same thesaurus without identification (i.e., BT/NT are used for BTG/NTG and BTP/NTP). In other words, hierarchical relations in thesauri are not necessarily generalization relations as known from ontologies but could also represent partonomical or instance based hierarchies (see Table 1). Additionally, from an ontological point of view, a distinction between partonomy (in an administrative sense) and spatial containment is required (see also Winston *et al.* 1987), while the partitive relationship is used in thesauri for both cases.

As discussed in section 2, in several cases association or equivalence relationships are used to indicate proximity. This should be part of the search and annotation framework, but not part of the feature type representation itself. In this work, we will use semantic similarity measures to represent proximity. In terms of ontology, the association relation needs to be replaced by concrete relations holding between given types respectively their entities (e.g. via the domain and range specified for a certain relation).

2.4 *Non-Formal Language Semantics*

Most difficulties in mapping from feature type thesauri to ontologies are caused by a lack of formal semantics and language expressivity issues in thesauri. As pointed out in section 2.1, generic hierarchical relations are defined less strictly than sub and super class relationships in ontology engineering. An example is the concept *hydrographic structures* in the ADL Feature Type Thesaurus (FTT). *Hydrographic structures* are defined as "constructed bodies of water". The subconcept *canals* fits this definition, while the subconcept *offshore platforms* does not. Consequently, searching for *hydrographic structures* using the ADL Gazetteer Webclient also returns *offshore platforms* (which is not a body of water). While this also points to a mixture of feature and workflow representation (see section 2.2), it shows that a formal definition (which allows for automated consistency checking) of the used relations is necessary. The same argument holds for the grouping of feature types in the GeoNames.org typing scheme. From an ontological point of view, a first step would be to examine whether hierarchical relations used within a geographic feature type thesaurus are used in a consistent way and whether they are generic, partitive or individual based relations.

In addition to relationships, difficulties arise for feature types themselves. In thesauri, terms are organized as a controlled vocabulary without a formal definition. Their meaning is determined by their hierarchical position within the thesaurus, by their textual definition (if present), and especially by the sets of instances linked to them, i.e. extensionally. Interpretation of terms also varies from one thesaurus to the other. For example, in the ADL FTT, the term *countries* is defined as "[t]erritory occupied by a large group of people organized under a single, usually independent government, and recognized as a country internationally." The non-preferred term *nations* is specified as an equivalent term. The Getty Thesaurus of Geographic Names (TGN), in contrast, gives preference to *nations* and reserves the term *countries* for rare situations such as the divisions of the United Kingdom (e.g., Scotland, Britain, etc.). Consequently, the ADL Gazetteer lists 165 countries while the TGN only lists 11. The Getty typing scheme is in fact an extended version of the Art & Architecture Thesaurus, which defines *nations* as the preferred term for *countries*. However, this seems to contradict the results obtained by type-lookup. The point is not, that conceptualizations may differ, but the lack of a formal definition which would allow to distinguish both concepts without knowing their instances, i.e., intentionally.

2.5 *Conclusions*

The difficulties examined in mapping from feature type thesauri to ontologies lead to the following assertions: first, because the relationships in thesauri are not explicitly defined, existing feature typing schemes can be converted to ontologies only through a process that includes validation of the relationships as required by ontologies. Second, in several cases these relations are not sufficient to disambiguate concepts, so that textual definitions and instances have to be taken into account. For the same reasons, the type-lookup operation defined as one of the two core functionalities of a gazetteer often returns counter intuitive results. The following section describes the type-lookup functionality of the examined gazetteers in more detail.

3 Gazetteer Communication Paradigm

Gazetteer services are the reason for creating feature type descriptions in the first place. Accordingly, a feature type ontology must provide optimal support for the type-look functionality of gazetteer services. For this purpose, it is useful to analyze the functionality and shortcomings of current gazetteer services. Communication with gazetteers is based on two different paradigms: *Web interfaces* provide access to the gazetteer functionality for users, whereas *Application Programming Interfaces* (APIs) allow other services and applications to query the gazetteer. The Alexandria Digital Library (ADL) Gazetteer web interface and API, the Getty Thesaurus of Geographic Names (TGN) web interface and the GeoNames web interface and API are used to show different approaches to the implementation of these communication paradigms, using the feature types *Canal* and *Channel*.

3.1 *Use Case: Canal and Channel*

To demonstrate the feature type interoperability problems of current gazetteer services and our proposed approach to overcoming them, we will use the feature types *Canal* and *Channel* as test feature types. These two feature types were chosen because they are especially suitable to demonstrate the challenges in the description of such types. Canals and channels are related to each other and they can easily be confused. The exact meanings of the words are ambiguous and differ slightly, depending on the source of the definition. For example, the WordNet definition of canal is a "long and narrow strip of water made for boats or for irrigation", whereas the WikiPedia definition is "a manmade water channel". The definitions for channel are "a deep and relatively narrow body of water" (WordNet) and "a narrow, enclosed around the sides, deep, waterway connecting two bodies of water" (WikiPedia). There is agreement that a canal is a constructed feature and a channel is a natural feature but other details of the character and function of these feature types differ. Moreover, both terms can appear in different contexts. For example, a canal can be regarded as a route of transport navigable by ships, as an artifact, or as a hydrographic feature. Ideally, a general, multi-purpose feature type definition should account for all of these aspects. These feature types serve as good examples for how ontologies can be used to disambiguate related concepts. The problems shown by means of these examples repeatedly appear when building a feature type hierarchy.

3.2 *Web Interface*

The Web interfaces presented here provide access to the gazetteer data for users, i.e. human agents as opposed to artificial agents such as applications or services. Accordingly, visitors of such Web interfaces are shown a form which allows them to enter a query. Experienced Web users can be expected to be familiar with filling in HTML forms. Nonetheless, the different levels of complexity and partly ambiguous semantics of the used terminology can be confusing for gazetteer Web interface users, as will be shown in the following.

Alexandria Digital Library Gazetteer Server Client. The Alexandria Digital Library (ADL) Gazetteer Server Client[5] is a Web interface consisting of an interactive map and an HTML form, as shown in figure 1a. The map allows the user to specify the region of the Earth to include in the query; that is, the query is looking for named geographic places *within* or *overlapping* the region shown in the map. The map region is changed by zooming and panning to the desired location and extent. The query string for place name can be typed in and the search function can be refined using operators such as *has any words*, *has all words*, etc. The form also allows for a temporal constraint by selecting a place status, which can be *current*, *former* or *proposed*. Finally, if the ADL Gazetteer identification code of a specific gazetteer entry is known to the user, it can be directly entered into the appropriate form field.

Most interesting in the context of this paper is the client's functionality that allows the user to restrict the query to a specific feature type by selecting it from a predefined list. This list is based on the ADL

[5] http://webclient.alexandria.ucsb.edu/client/gaz/adl/index.jsp

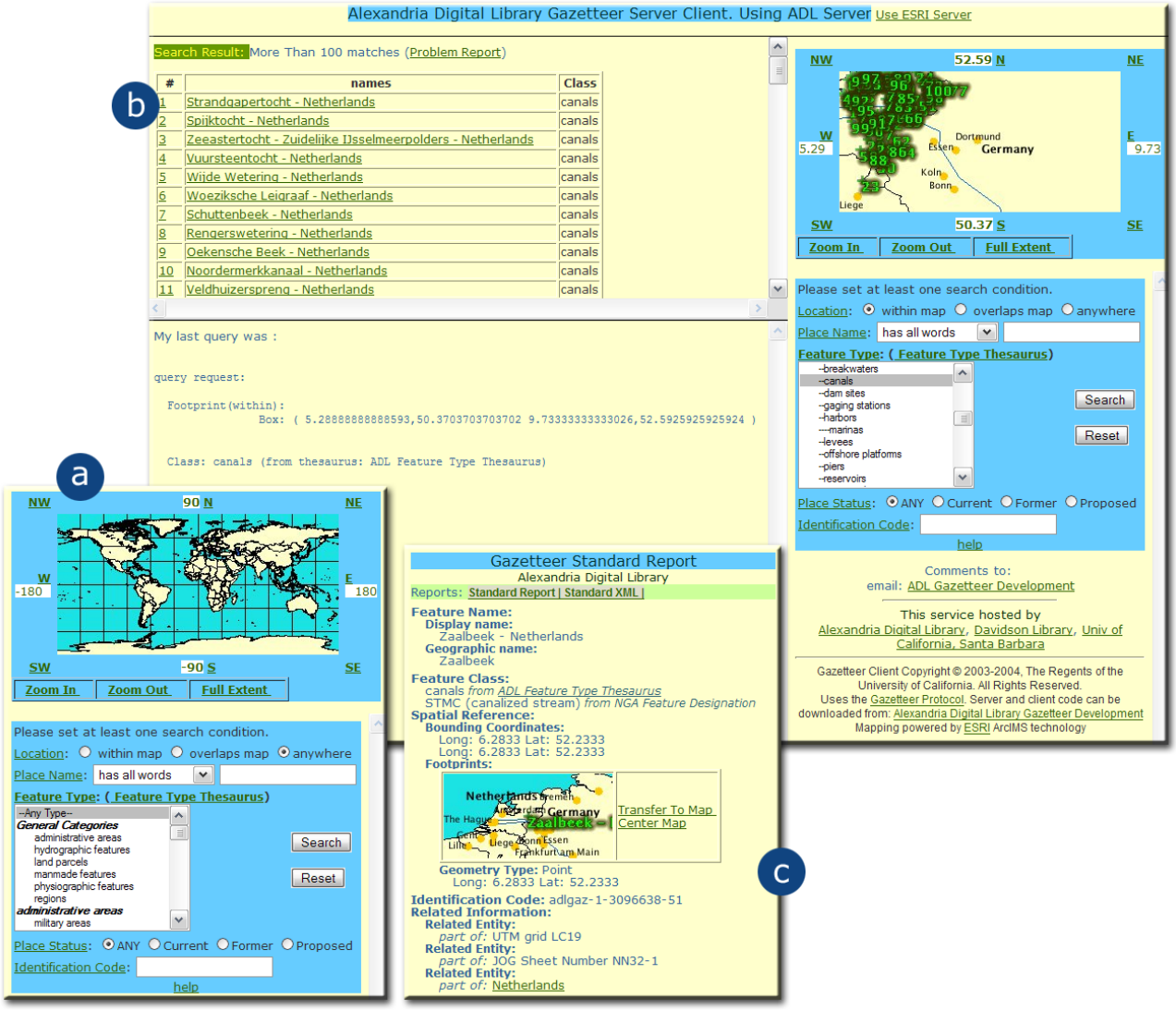

Figure 1. Overview of the ADL Gazetteer Web interface: query form with map (a), list of query results (b) and detail view for a single feature (c).

Feature Type Thesaurus[6] (FTT). At the top of the list are FTT's top level categories that can be selected for broad, general searches by type; below this, the complete hierarchical list of feature types is displayed. There are over 200 selectable feature types, organized in 6 top-level categories. The exact definitions of the feature types can be looked up by following a link to the ADL FTT Web page. Users can also consult the FTT if the feature type they are looking for is not on the list. This is the case if a user is looking for a non-preferred term for a feature type, as the form lists only preferred terms. For example, the term *irrigation systems* is not listed, but this term can be found in the FTT with the direction to *USE canals* instead. That is, gazetteer features that are instances of irrigation systems can be found by using the feature type *canals*, which is grouped under *hydrographic structures*, and falls into the class of *manmade features*. Although this classification is correct in principle, a user might wonder why canals cannot be found under *hydrographic features* (which is due to the single inheritance approach underlying the ADL FTT). A search by feature type in the ADL Gazetteer is automatically expanded to all sub-terms of the type entered in the query. That is, if *hydrographic features* is part of the query then the query results will include not only features classified with that term but also all features classified with any of the sub-terms of *hydrographic features*.

[6]http://www.alexandria.ucsb.edu/gazetteer/FeatureTypes/ver070302/index.htm

The presentation of the query results consists of a numbered list, with the corresponding numbers shown at the features' locations on the map (see figure 1b). Additionally, the query that has been produced from the user's form input is provided. Clicking one of the results opens a detailed description for this feature (see figure 1c), containing the feature's name and class, its footprint (including an overview map) and related information such as *part-of* relationships.

Getty Thesaurus of Geographic Names. The Web interface for the Getty Thesaurus of Geographic Names[7] (TGN) only has three input fields: the *name* of the place the user is looking for, its *place type* (corresponding to ADL's feature types) and *nation*, as shown in figure 2a. The user can choose from a list of all place types (figure 2b) and a list of all nations via links next to the corresponding form fields. Although the hiearchy of geographic names (figure 2c) is available, a corresponding link for the place types is missing. Users have to visit the *About the TGN* page to find out about the place type hierarchy. Here it is explained that the typing scheme for the TGN is based on the place type hierarchy developed for the Art & Architecture Thesaurus[8] (AAT). Figure 2d shows the AAT place type hierarchy for *canal*.

Note that the TGN Web interface does not support searching for places within a user-defined map extent. The only spatial restriction that is possible through the query form is the selection of a nation. This is because the Getty TGN is a gazetteer structured as a thesaurus with part-whole hierarchy (e.g. *California part-of United States*) and, although spatial coordinates are available for many of the entries, coordinates are not present for all entries. After submitting a query, a list of results is presented to the user (see figure 3a), comparable to the one provided by the ADL Gazetteer. Every result is linked to detailed information on that place, containing an ID, geographic coordinates and the place's name(s) and type (see figure 3b). Additionally, the hierarchical position in the TGN is shown, and the original sources for the presented data is listed.

GeoNames. The GeoNames[9] service provides two different Web interfaces. The basic interface only consists of an input field for location name and a drop-down menu for country selection, so we focus on the advanced search interface here (figure 4a). In addition to the form fields available in the basic version, it allows for the restriction of a query to a group of feature types and to a specific continent. The user can activate *fuzzy search* through a checkbox. Unfortunately, this option is not explained and thus remains unclear to the user.

Compared to the lengthy sets of feature types provided by the ADL and TGN Web interfaces, GeoNames has a very short list with only 9 feature classes available for selection. Note that this list is not structured as a thesaurus; it is a simple list. Each feature class represents a group of feature types (figure 4b). The classes are broadly defined by 1-3 typical feature types, but users can only guess what other feature types might be included in a particular class. The classification used for this list is (comparable to Getty) found in a different section[10] of the GeoNames Web site, that is only accessible through the sitemap and the "frequently asked questions" in the forum. The complete list of codes (see figure 4e for an extract) is a slightly extended version of the feature codes introduced by the United States National Geospatial-Intelligence Agency (NGA). Users who do not know this classification will have difficulties selecting the appropriate feature class for their query, especially because the classification is partly counter intuitive (e.g., the feature types *continent* and *military base* are classified under "parks, area ..."; see also the "stream, lake,..." group in figure 4e).

The query results in GeoNames can be displayed in two ways: either as a list (figure 4c), or on a map with markers for the query results (figure 4d). Clicking a result gives different information at a different level of detail in each of the two result displays. In the map view, the result markers are linked to pop-ups, which contain the feature's spatial containment hierarchy, geographic coordinates, links to WikiPedia entries and RDF files, etc. In the list view, the results are linked to a detailed view that shows a map with all features that are spatially contained in the corresponding feature. This is not made clear in the query form, (i.e.

[7] http://www.getty.edu/research/conducting_research/vocabularies/tgn/
[8] http://www.getty.edu/research/conducting_research/vocabularies/aat/
[9] http://www.geonames.org/
[10] http://www.geonames.org/export/codes.html

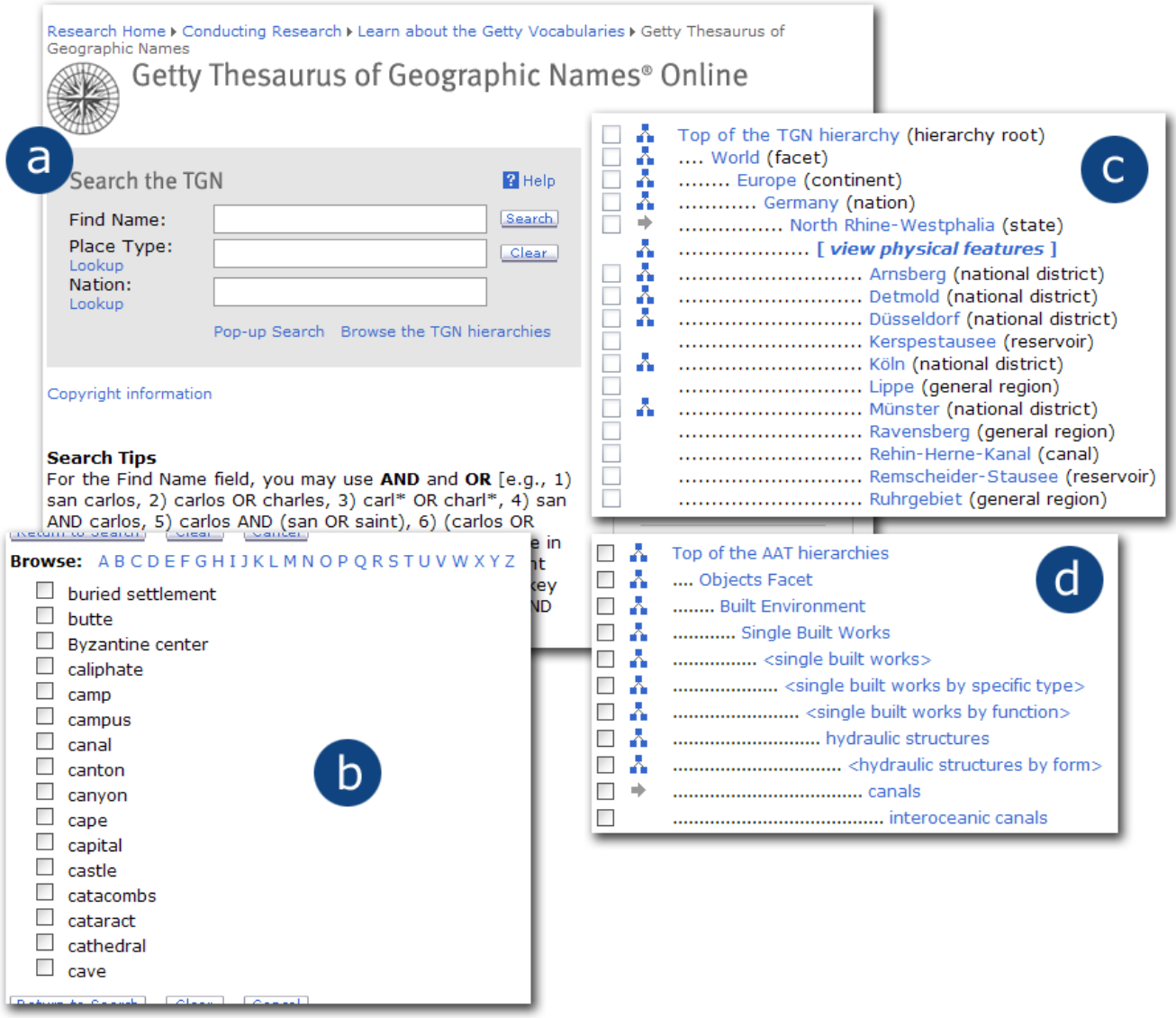

Figure 2. Overview of the Getty Web interface: query form (a), place type list (b), place name hierarchy (c) and place type hierarchy as provided by the Art & Architecture Thesaurus (d).

the user will probably expect the same information, only differing in visualization, when they click either "search" or "show on map").

3.3 *Application Programming Interface*

Application Programming Interfaces (APIs) are code interfaces (as opposed to the visual interfaces discussed in the previous section), that make a program's functionality available for developers, who can access it from their code. The number of publicly available APIs for Web services has grown remark-ably recently, as more and more companies grant developers access to the their services and data. Well-known examples include Google Maps[11], the social events calendar Upcoming.org[12] or the Flickr photo service[13]. An impressive number of so-called *mashups* has been created combining APIs to form complex and (at least in some cases) very useful services. In addition to such proprietary APIs, there are also efforts for

[11] http://www.google.com/apis/maps/
[12] http://upcoming.org/services/api/
[13] http://www.flickr.com/services/api/

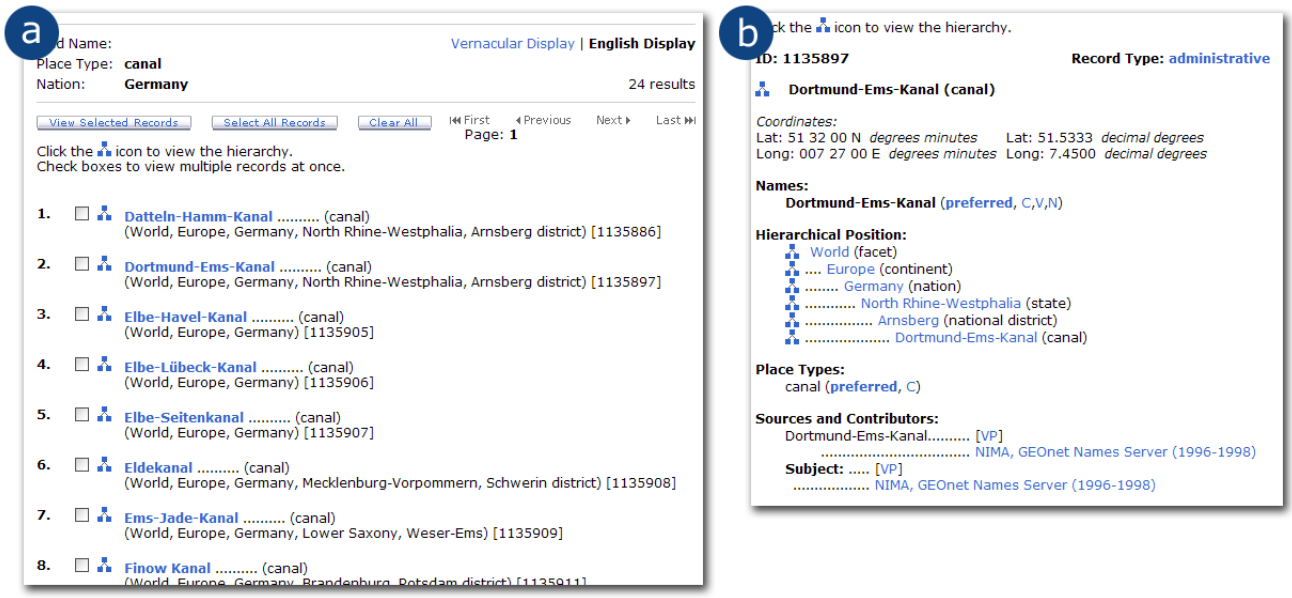

Figure 3. Getty presents the matching results for a query as a list (a); clicking a result opens the detailed description of the place (b).

standardized open APIs, mostly driven by the Open Geospatial Consortium (OGC) for the geospatial domain. Apart from the well-established Web Map Service (WMS) and Web Feature Service (WFS) specifications and activities in various other areas of geospatial service standardization, OGC has also addressed gazetteer functionality as a special WFS profile (Fitzke 2006). It is based on the abstract data model for gazetteers described in the ISO 19112 standard "Spatial referencing by geographic identifiers" (ISO 2003). However, the OGC WFS gazetteer profile has not yet reached specification status. Looking at the three gazetteers in the focus of this section, APIs are available for GeoNames and the ADL Gazetteer; there was no public API available for Getty at the time this paper was being written.

GeoNames Web Services. GeoNames provides a number of Web services, such as for GeoNames search[14], geocoding[15], inverse geocoding in various flavors[16] and access to the GeoNames geographic places ontology[17]. The services are realized in different ways, which are all based on the HTTP protocol. Parameters for the queries are either encoded as part of the URL in GET requests, returning the results in different formats, or they act as Web services (Fielding 2000). All of these services provide only different means for accessing and searching the gazetteer data. The results are given in different formats, but they do not support any further functionality on type level.

ADL Gazetteer Protocol. The API for the ADL Gazetteer is based on the ADL Gazetteer Protocol, which is also used by the ADL Gazetteer Server Client for communication. The protocol defines three independent HTTP-based services: *get capabilities*, *query* and *download*. Unlike the GeoNames services, the requests are encoded as XML files that are sent to the service via HTTP-POST ("XML-over-HTTP"). All requests to the service, as well as the resulting responses must adhere to the XML schema developed for this purpose[18].

The *get capabilities* service allows clients to access the gazetteer metadata, i.e. it provides information on the functionality provided such as supported query operators and available thesauri. The *query* and *download* services provide access to the actual gazetteer entries. The former allows for selection of entries by a query in the gazetteer query language, whereas the latter returns all entries. In both cases, clients

[14]http://www.geonames.org/export/geonames-search.html
[15]http://www.geonames.org/export/free-geocoding.html
[16]http://www.geonames.org/export/reverse-geocoding.html
[17]http://www.geonames.org/ontology/; The GeoNames ontology contains concrete places that are interlinked with each other via the *contains*, *neighbours* and *nearby* relationships, as opposed to the ontology on type level proposed in this paper.
[18]http://www.alexandria.ucsb.edu/gazetteer/protocol/gazetteer-service.xsd

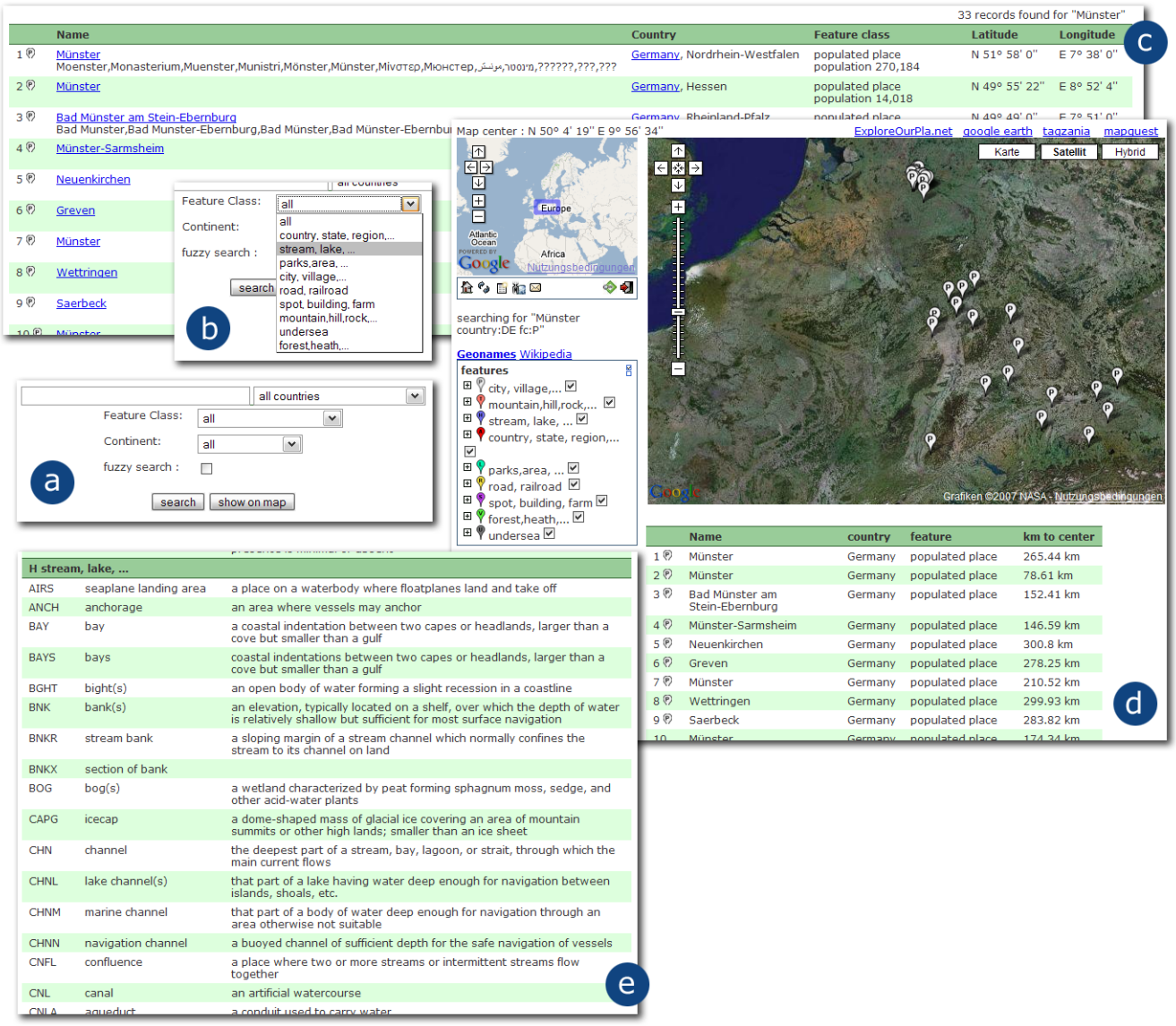

Figure 4. Overview of the GeoNames Web interface: advanced search form (a), feature class drop-down list (b), list view of query results (c), map view of query results (d) and extract from the feature codes classification (e).

can request either standard or extended reports. The standard report format is part of the ADL Gazetteer protocol and contains the elements *identifier*, *codes*, *place-status*, *names*, *footprints*, *classes*, *relationships*, *display-name* and *bounding-box*; footprints are encoded in Geography Markup Language (GML). As the information provided by a specific gazetteer usually exceeds the elements of the standard report, the extended report allows gazetteer providers to specify their own report format. Extended report formats must be defined as XML schemas. Clients can utilize the extended reports by downloading those schemas, which are linked in the capabilities documents.

3.4 *Functional Comparison*

While the core gazetteer functionalities are provided by all three gazetteers under consideration, the services differ in detail (see table 3 for an overview). The available functions focus on place names, coordinates, and administrative hierarchies and require improvement for enhanced functionality regarding feature type searching. From a user's point of view, access to a gazetteer via a Web interface should present the available data in a structured way that allows for easy exploration and intuitive query building. To achieve this, it is essential that the user has a clear understanding of the feature types used by the gazetteer. As shown

above, some of the Web interfaces make accessing the underlying feature type definitions difficult. However, this is not only a user interface issue. We propose that a unifying feature type ontology will alleviate some of the inconsistencies and limitations of existing feature typing schemes, support more effective feature type searching across gazetteers, and consequently support more intuitive Web and API interfaces.

Table 3. Comparison of the functionalities of the three gazetteers under consideration.

	ADL	**Getty**	**GeoNames**
Web Interface Functionality			
Place name search	√	√	√
Place type restriction	√	√	√
Spatial restriction by nation	√[1]	√	–
Spatial restriction by continent	√[1]	√[2]	√
Spatial restriction via map extent	√	–	–
Temporal restriction	√	–	–
Feature type description lookup	√	√[3]	√[3]
Visualization of results on map	√	–	√
API Functionality			
Capabilities descriptions	√	–	–
Query by place type	√	–	√
Geocoding	√	–	√
Inverse Geocoding	√	–	√
Query by spatial containment	√	–	√
Place status query	√	–	–
Query by relationship (instance level)	√	–	–
Query for neighbors / nearby features	–	–	√

[1] Indirectly via map extent.
[2] Indirectly via nation selection.
[3] Not directly accessible from the Web interface.

To enable automatic integration of and access to different gazetteers, distinct formal definitions of at least a core set of the feature types are required. In the following sections, we explain our approach for a feature type ontology and demonstrate how both Web interfaces and APIs can benefit from such an ontology.

4 Development of a Feature Type Ontology

In sections 2 and 3 we argued that feature type thesauri do not provide optimal support for type-lookup functionality. We chose to use the ADL FTT as the source of terminology and structure to populate our experimental feature type ontology. This section describes the steps of the process using the methodology introduced by van Assem *et al.* (2004) and Hepp (2006). We also distinguish between conceptual design decisions and the actual representation in DL. As the definition of all concepts forming the ADL thesaurus is out of scope for this article, the concepts *Canal* and *Channel* will be used as examples to demonstrate our approach.

4.1 *Conceptual Design*

This section describes the conceptual design of the feature type ontology without going into detail about the implementation. The questions to be answered here focus on aspects such as the chosen level of abstraction (generalization) and the purpose for which the ontology is developed. While the feature type ontology should describe the nature of real world entities, an ontology is always developed for a specific purpose. Taking Guarino's (1998) distinction into top-level, domain and application ontologies as well as Uschold's (2000) notion of global and local ontologies into account, we consider the feature type ontology to be a (global) domain ontology. Hence the purpose is to specify concepts on a level of abstraction that is generic enough to be used and further refined by multiple gazetteers, but also detailed enough to be directly applicable for type-lookup. Each gazetteer can then add complementary sub concepts to this

domain ontology, i.e. feature types to describe its own perspective within an application ontology, while the domain ontology provides the shared (and formal) vocabulary necessary to perform type lookups spanning over multiple gazetteers.

van Assem *et al.* (2004) describe a method to convert thesauri into RDF and OWL ontologies in order to make existing, agreed-upon community knowledge available for use in the Semantic Web. The transformation methodology is structured into four steps, namely *preparation*, *syntactic conversion*, *semantic conversion* and *standardization*, and contains hands-on guidelines for each step (such as "preserve original naming as much as possible" or "avoid redundant information"). According to the syntactic conversion process described by van Assem *et al.* (2004), the proposed feature type ontology should preserve the structure and naming of the original thesaurus. To achieve this, the terms defined by the ADL FTT are reused as follows:

(D1) Both preferred and non-preferred terms are converted to concepts (i.e. feature types) in an ontology. To preserve the structure of the FTT, concepts stemming from non-preferred terms are either sub classes or equivalent classes of already defined concepts (e.g. previously specified as preferred terms). We do not claim that all these concepts should be part of a global feature type ontology but can be specified as additional concepts by local gazetteers. This should be the case if a particular type is not of general interest or cannot be clearly separated from other domain level concepts. If a non-preferred term is only a spelling variant, it should neither be part of the global ontology discussed here, nor any local ontology. Such variants can be used as alternative names on application level to improve the gazetteer interface, i.e. to make the underlying ontology more accessible for a particular group of users.

(D2) If a scope note contains more than one definition and some of those definitions contradict (in terms of inheritance), new concepts are introduced for these definitions. New concepts are named using one of the terms formally entered as alternative terminology (e.g., equivalent terms), if possible.

(D3) If new concepts have to be introduced or existing ones are renamed, additional feature type thesauri (or catalogs) are taken into account. This includes the Getty TGN typing scheme and GeoNames.org typing scheme, but also WordNet to ensure that the used terms also reflect the intended common understanding of end users.

(D4) As feature types are formal specifications of concepts (in our minds), all feature types are named in singular form. This is also a common convention for ontologies.

Regarding the semantic conversion proposed by van Assem *et al.* (2004), the implicit semantics of a thesaurus has to be made explicit and interpreted in terms of the new representation format. In case of the ADL FTT, one has to decide whether the generic hierarchical relationships should be kept and represented with classical subsumption (is-a) relations or represented in the ontology with a different set of relationships. Unlike several other thesauri, ADL does not use taxonomic relations for partonomic relationships (see section 2.3), therefore it is possible to create an is-a hierarchy based on BT and NT. This conversion cannot be done automatically. For each type it must be checked whether all types formally related via NT or BT are sub classes (respectively super classes) as defined in section 2.2. As subsumption is transitive, the process needs to be repeated for all *Narrower Term* and respectively *Broader Term* relations. Moreover, while the ADL thesaurus is specified as a single inheritance hierarchy, the proposed ontology makes use of multiple inheritance. This leads to changes within the hierarchy and to new concepts that are handled according to D1 - D4.

The ADL FTT has six top terms; all terms in the thesaurus are sub-terms of one of these terms. The feature type ontology has a common super type called feature, indicating that all instances of further specified subtypes are (geographic) features[19]. ADL uses the top term *hydrographic features* for all natural bodies of water (e.g. channels, rivers), while constructed bodies of water (e.g. canals, reservoirs) are *hydrographic structures*, which is a sub type of the top term *manmade features*. This distinction is imposed by the single inheritance structure of ADL and the split between natural and constructed features at the top level of terms. However, from an ontological perspective as well as from a cognitive (i.e. user centered)

[19]The ADL FTT also has an abstract common super type for the top terms; however, it is neither shown in the Web interface nor specified in the thesaurus.

point-of-view, this causes several difficulties. First of all, applying similarity or subsumption based information retrieval (Janowicz 2006a, Lutz and Klien 2006) leads to unsatisfying results. Based on the FTT feature type definitions, lakes and reservoirs do not share a common more general term; the relationship between them is expressed as an *associative* relation (i.e., the reciprocal relationship of *lakes* SEE ALSO *reservoirs*). Second, users of a gazetteer service are often not aware of whether a certain body of water was created by humans or not. Many hydrographic features are influenced by humans to some degree, making it hard to distinguish between natural and artificial. It should be noted that since the associative relationship is present between *lakes* and *reservoirs*, it is possible in a user interface to prompt the user to search by both terms when either of them is entered as a search term, but this does not help with conversion to an ontology because the associative relationships in thesauri are not formally defined. Additionally, in other cases such usage of the associative relationship may be misleading. For instance, in the ADL FTT *canals* has *tunnels* as related term, while it is unlikely that a user searching for canals is interested in tunnels.

Similar arguments can be applied to the FTT term *transportation features* which is defined as a sub-term of *manmade features*. The difficulties arising here become clear if one regards what FTT includes as subterms of *transportation features*. For example, if *aqueducts*, *roadways* and *parking sites* are *transportation features*, why are *canals* not (they are connected via an associative relationship)? From an ontological point-of-view, one may argue that transportation is a Thematic Role (Sowa 2000) or affordance (Gibson 1977) which instances of a certain type may play, and not a feature type itself. Besides problems concerning available representation languages (Kuhn 2007), this kind of modeling would require the definition of transportation devices (and even entities to be transported) for every transportation feature. For the feature type ontology proposed in this paper, we decide to keep the *transportation features* type as part of our multiple hierarchy. We use the concept *TransportationFeature* in three ways. First, feature types classifying entities that were explicitly built for the purpose of transportation are defined as subtypes of *TransportationFeature* (e.g. *Canal*). Second, gazetteers adopting the feature type ontology can specify additional subtypes as intersections of the further distinguished type and *TransportationFeature* (see Figure 5). A local gazetteer, for example, may refine the *River* type by introducing a subtype for rivers that are important waterways within a specific region (such as the Dutch *Grachten*). Third, if a feature of a given type is mostly used for transportation, it can be additionally typed using a subclass of *TransportationFeature*. For instance the river Rhine is one of the most important waterways in Europe and could be annotated as *River* and *TransportationFeature*. The same approach discussed here for *transportation features* is also applied to transform the FTT type *manmade features* into *ManmadeFeature*.

Lastly, this paragraph gives a brief insight into how additional relations can improve the feature type description and support more complex queries. As an example for topological relations, we discuss *hasOrigin* and *hasDestination* here. The relations hold between "linear bodies of water flowing on the Earth's surface"(ADL Feature Type Thesaurus), but can be further generalized. We assume that streams and all subtypes (e.g. rivers) have an origin (usually a spring) and a destination which might be another stream or waterbody[20]. While it is characteristic for streams (e.g. rivers) to have a flowing direction (which influences our way of thinking and interacting with streams), this is not the case for channels and canals. Both connect two or more hydrographic features without a pre-given (flowing) direction (see Figure 5; one could also use the more general *hasConnection* relation). In most examined thesauri, *Channel* either denotes "[relatively] narrow seas or stretches of water between two close landmasses and connecting two larger bodies of water [or deeper] parts of a moving body of water (as bays, estuaries, or straits) through which the main current flows or which affords the best passage through an area otherwise too shallow to navigate."(ADL Feature Type Thesaurus). According to D2 and D3, we use the first part of the definition here and leave navigation channels aside. While canals are man-made features built for transportation, channels are not.

[20]Note that this is a simplification pointing to some interesting ontological questions which cannot be discussed here for lack of space. For instance, rivers can also end in sinkholes (in ADL the top term *physiographic features* should be used instead).

To specify sinkholes as feature types is difficult especially if the river slowly trickles away and no crisp border can be defined. To point out a possible solution we defined *Inlet* (a (narrow) watercourse extending into the land) as having an origin (a *Lake* or *Sea*) but without specifying a destination. Hence, one could specify destinations for individual inlets (or subtypes), but it is not mandatory for the type *Inlet*.

The ADL Gazetteer and the ADL FTT are independent entities; they are related only because the classification of features (e.g., gazetteer entries) in the Gazetteer are selected from the FTT. The ADL Gazetteer Content Standard (GCS) (Hill *et al.* 1999) on which the ADL Gazetteer structure is based allows the establishment of relationships between gazetteer entries. The existing ADL Gazetteer has implemented only one relationship type: the *part-of* relation between features. This is an administrative part-of relationship, not a spatial one, although an administrative relationship infers spatial containment in many cases. In the ADL FTT, there are no feature instances and thus there is no need for a part-of relationship. In our ontology, we propose a spatial containment relation (Winston *et al.* 1987) on feature type level to model facts such as that a capital is located within the borders of a country or that a lake (as inland water body) is surrounded by landmass.

4.2 *Description Logic (DL)-based Representation*

While section 4.1 describes general design decisions, this section gives brief insight into the implementation of the feature type ontology[21]. We have chosen the $\mathcal{SHOIN}$ description logic (DL) as representation language. $\mathcal{SHOIN}$ corresponds to the Web Ontology Language (OWL-DL) which is a well established standard defined by the W3 Consortium. OWL is used by most popular ontology editors (e.g. Protégé), and most DL-reasoners (e.g. Fact++) support subsumption reasoning for OWL-DL. Moreover, there are several theories and prototypical implementations such as the SIM-DL server (Janowicz *et al.* 2007) which support similarity measurement for expressive description logics. In other words, there is a ready-to-use infrastructure at hand to integrate the proposed ontology into existing gazetteers.

A discussion of the $\mathcal{SHOIN}$ description logic is out of scope for this paper; see Horrocks *et al.* (2003) and Baader *et al.* (2003) for detailed information on its semantics and application. The only aspect that needs to be kept in mind here is that the formal semantics underlying $\mathcal{SHOIN}$ maps to set theory.

Figure 5 shows an extract of the proposed feature type ontology displayed in the Protégé ontology editor. As an example, we will have a closer look at the concrete implementation of the feature types *Canal* and *Channel*. Both are specified as subtypes of *Watercourse*, which again is a subtype of *InlandWaterBody*. Formally, *Canal* is defined as

$$\text{Canal} \equiv \text{ManmadeFeature} \sqcap \text{TransportationFeature} \sqcap \text{Watercourse} \sqcap (\forall\ \text{hasDestination HydrographicFeature}) \sqcap (\geq\ \text{hasDestination 2})$$

while *Channel* is specified as follows:

$$\text{Channel} \equiv \text{Watercourse} \sqcap \neg\text{ManmadeFeature} \sqcap (\forall\ \text{hasDestination HydrographicFeature}) \sqcap (\geq\ \text{hasDestination 2})$$

Both canals and channels have at least two connections to other hydrographic features (such as lakes, rivers or even seas). In contrast to *Canal*, *Channel* is explicitly defined as not being man-made. Consequently, while some feature may be specified as a *Channel* used for transportation, nothing can be a *Channel* and a *ManmadeFeature* at the same time. In our definitions we refer to channel as natural connection between waterbodies, not to navigation channel or river bed. One could also omit $\neg ManmadeFeature$ and argue that *Channel* is a supertype of *Canal*, which is not done here. The implications of such design decisions regarding subsumption and similarity reasoning are discussed in section 5.

Note that the type *Harbor* in figure 5 was defined as man-made hydrographic feature. This was done with respect to D1-D4 and the scope note ("Sheltered areas of water where ships or other watercraft can anchor or dock.") from the FTT. From another point of view, a harbor (respectively port) can be further distinguished into its basin, docks, piers, etc. which are not necessarily hydrographic features. While this may be reasonable for application ontologies refining the presented ontology, one has to keep in mind that this definition would require to create individual features of type *Basin*, *Dock*, etc. to define a harbor.

[21]The ontology is under development, updated versions are available at http://sim-dl.sourceforge.net/downloads/.

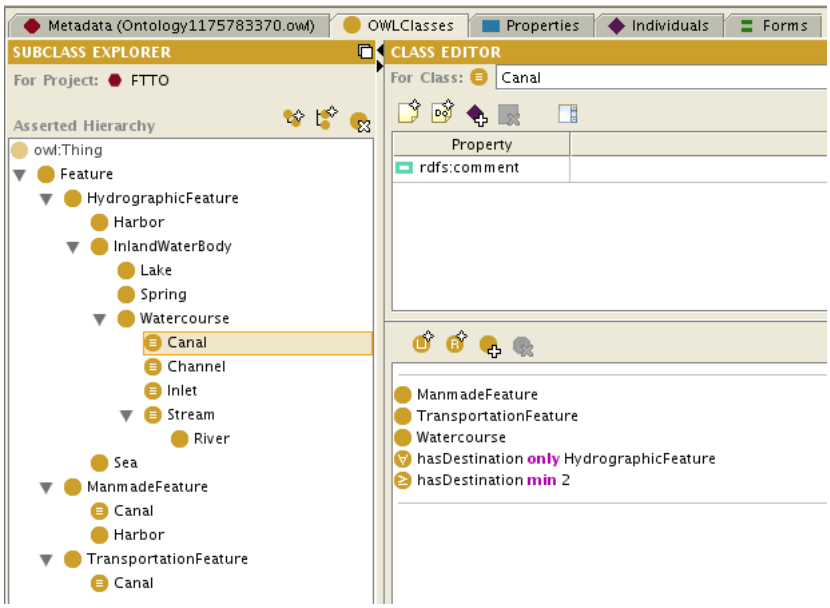

Figure 5. Extract of the feature type ontology in Protégé.

5 Subsumption and Similarity-Based Reasoning

In the previous section, we discussed the steps that are required to create a feature type ontology starting with the ADL FTT. Since this process is bound to a significant effort, the benefits of a feature type ontology must be pointed out to justify the cost involved. This section describes how the proposed feature type ontology can be integrated into the gazetteer communication paradigm. Starting with a brief insight into similarity and subsumption based information retrieval, the extended type-lookup functionality will be discussed by introducing a prototypical Web interface and pointing out possible extensions to the ADL Gazetteer Protocol.

5.1 *Subsumption & Similarity based Information Retrieval*

The notion of similarity originated in psychology and was established to determine why and how entities are grouped to categories, and why some categories are comparable to each other while others are not (Goldstone and Son 2005, Medin *et al.* 1993). The main challenge with respect to semantic similarity measurement is the comparison of meanings. A language has to be specified to express the nature of entities and metrics are needed to determine how (conceptually) close the compared entities are. While entities can be expressed in terms of attributes, the representation of types is more complex. Depending on the expressiveness of the representation language, types are specified as sets of features, dimensions in a multidimensional space, or formal restrictions specified on sets using various kinds of description logics. While some representation languages have an underlying formal semantics (e.g. model theory), the grounding of several representation languages remains on the level of an informal description. Because similarity is measured between types which are representations of concepts in human minds, similarity depends on what is said (in terms of computational representation) about compared types. This is again connected to the chosen representation language, leading to the fact that most similarity measures cannot be compared. Besides the question of representation, context is another major challenge for similarity assessments. In many cases, meaningful notions of similarity cannot be determined without defining in respect to what similarity is measured (Goodman 1972, Medin *et al.* 1993).

Similarity has been widely applied within GIScience for the past few years. Based on Tversky's feature model (1977), Rodriguez and Egenhofer (2004) developed an extended model called Matching Distance Similarity Measure (MDSM) that supports a basic context theory, automatically determined weights, and asymmetry. Raubal (2004) utilized conceptual spaces (Gärdenfors 2000) to implement models based on

distance measures within geometric space. The SIM-DL measure (Janowicz 2006a, Janowicz *et al.* 2007) was developed to close the gap between geo-ontologies described through various kinds of description logics, and similarity measures that had not been able to handle the expressiveness of such languages. Different similarity theories (Li and Fonseca 2006, Nedas and Egenhofer 2003) have been developed to determine the similarity of spatial scenes.

Subsumption-based reasoning has its origins in computer science and especially within knowledge representation. It is the most prominent of several inference techniques used within ontology based information retrieval. The idea behind subsumption-based retrieval as described by Lutz and Klien (2006) is to rearrange a queried application ontology taking a search concept into account and to return a new taxonomy in which all subconcepts of the specified search concept satisfy the user's requirements. However, using this approach forces the user to ensure that the search concept is specified in a way that it is neither too generic (and therefore at a top level of the new hierarchy) nor too specific to get a sufficient result set. In fact, the search concept is a formal description of the minimum characteristics all retrieved concepts need to share.

The benefits similarity offers during information retrieval, i.e. to deliver a flexible degree of conceptual overlap to a searched concept, stand against shortcomings during the usage of the retrieved information, namely that the results do not necessarily fit the user's requirements. To make the difference between both approaches more evident one can imagine a search concept specified by using a shared vocabulary (such as the proposed feature type ontology) to retrieve all concepts whose instances *overlap* with waterways. In contrast to the subsumption-based approach, similarity measurement would additionally deliver concepts whose instances are located *inside* or *adjacent* to waterways, and indicate through a lower degree of similarity that these concepts are close to, but not identical with the user's intended concept.

In this work, we use the SIM-DL measure (Janowicz 2006a, Janowicz *et al.* 2007) to compare feature types for similarity. SIM-DL supports high expressive description logics and combines both subsumption and similarity reasoning to achieve the best possible results[22]. Similarity between concepts is measured by comparing their definitions for overlap. A value of 1 indicates that two compared concepts are specified by the same set of superconcepts, while 0 indicates that the concepts have nothing in common. A similarity between 0 and 1 states that the compared concepts share at least some common superconcepts. In general, similarity values should not be interpreted separately but used to derive a ranking. Based on the specifications given in section 4.2, channels are more similar ($\sim$0.71) to canals than streams ($\sim$0.43), as both have no explicit direction, while a stream needs to have an origin. On the other hand, a canal is more similar to a stream than to a lake ($\sim$0.28). Note that these similarities are computed on the type level, and not between particular features described in a gazetteer.

The integration of similarity and subsumption based reasoning into a shared gazetteer infrastructure is depicted in figure 6. The proposed feature type ontology can be extended by local gazetteers, which align their own concepts as subtypes of existing ones. The user can perform type-lookup by selecting a search concept (C_s) using the new interface proposed below. In the case of subsumption, the result is a list of subconcepts of C_s. In contrast, the cloud representing the similarity query indicates that the result is a descending list of proximity values describing how close particular feature types are to the search concept.

5.2 *Web Interface Implementation Approach*

In this section, we demonstrate how Web interfaces can benefit from a feature type ontology supporting subsumption and similarity-based reasoning as outlined above. The focus during the development of the conceptual design of the gazetteer Web interface presented in the following was on overcoming the problems with the current Web interfaces shown in section 3. The proposed interface covers both the search functionality and the presentation of the results, allowing for a workflow in which the users can continuously refine their queries until the desired results are returned. We will concentrate on conceptual aspects and the user workflow in the following, leaving implementation details aside.

[22] A (prototypical) server implementation is available at http://sim-dl.sourceforge.net/downloads.

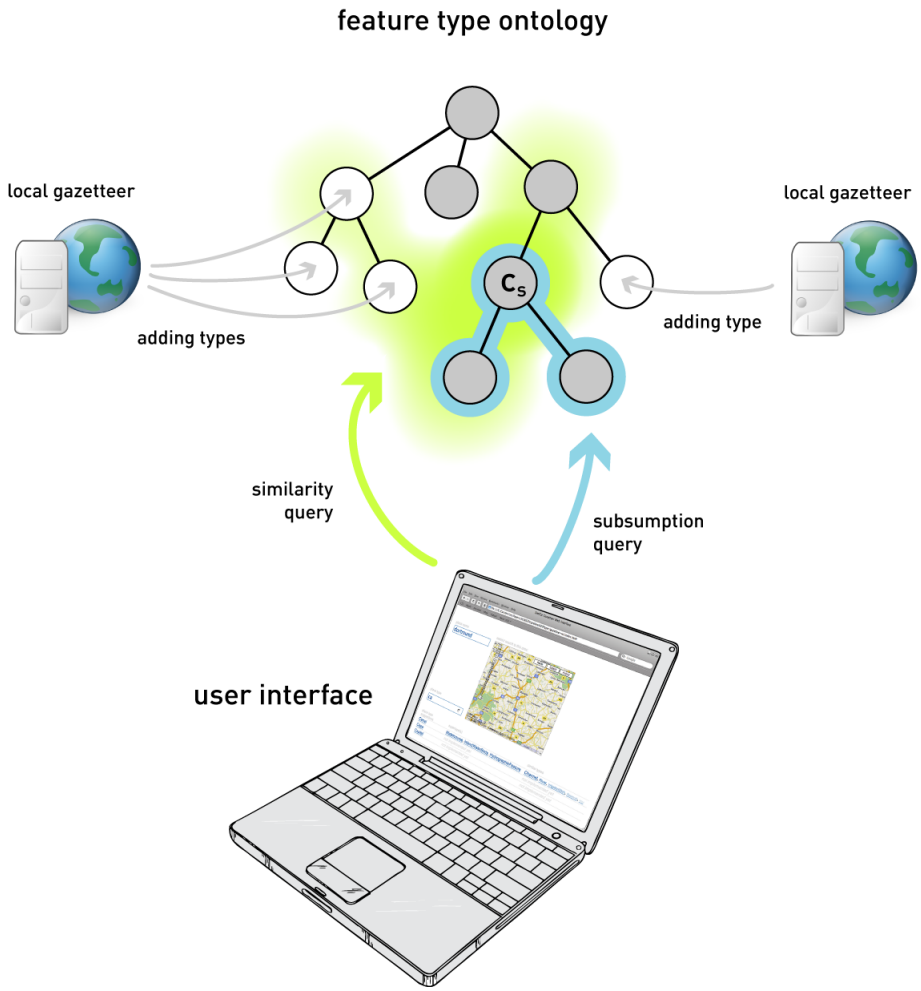

Figure 6. Subsumption and similarity based information retrieval within the proposed gazetteer infrastructure.

The proposed search interface consists of two input fields and a map as shown in figure 7a, reducing the cognitive load for the user as far as possible. The form fields allow the user to specify a place name and place type to search for. If they want to restrict the search to a specific region, they can use the map to specify the corresponding extent. All of these inputs are optional, so that the users can choose any combination of place name, place type and map extent, depending on what they are looking for.

The biggest problem of the Web interfaces discussed in section 3 was the incorporation of the feature types into the user interface. To avoid lengthy lists and ambiguous categorizations, we propose a *search-while-you-type* approach for this problem: the feature type selection only consists of a text field, so that the users can fill in whatever feature type they think is suitable to describe what they are looking for. As the users enter the feature types they have in mind, lexical matches are suggested by the Web interface (see figure 7b). When the client is loading new data from the server, this is pointed out by an activity indicator in the place type[23] input field. For every suggestion, direct supertypes and up to five of the *most similar* feature types from the ontology are presented, so that the user gets a quick overview of related types. The similarity of the types is indicated by font size and color: the less similar the type is to the suggested feature type in the leftmost column, the smaller and paler it is displayed. This visualization paradigm is adapted from so-called *tag-cloud* navigation. For the selection of the most similar types, the proposed system uses a threshold similarity value to make sure that only reasonable feature types are displayed. The user can select any of the types suggested by the system by clicking on it. The feature type is then transferred to the input field and used for the query.

[23]The technical term *feature* is avoided in the user interface in favor of the more common *place*.

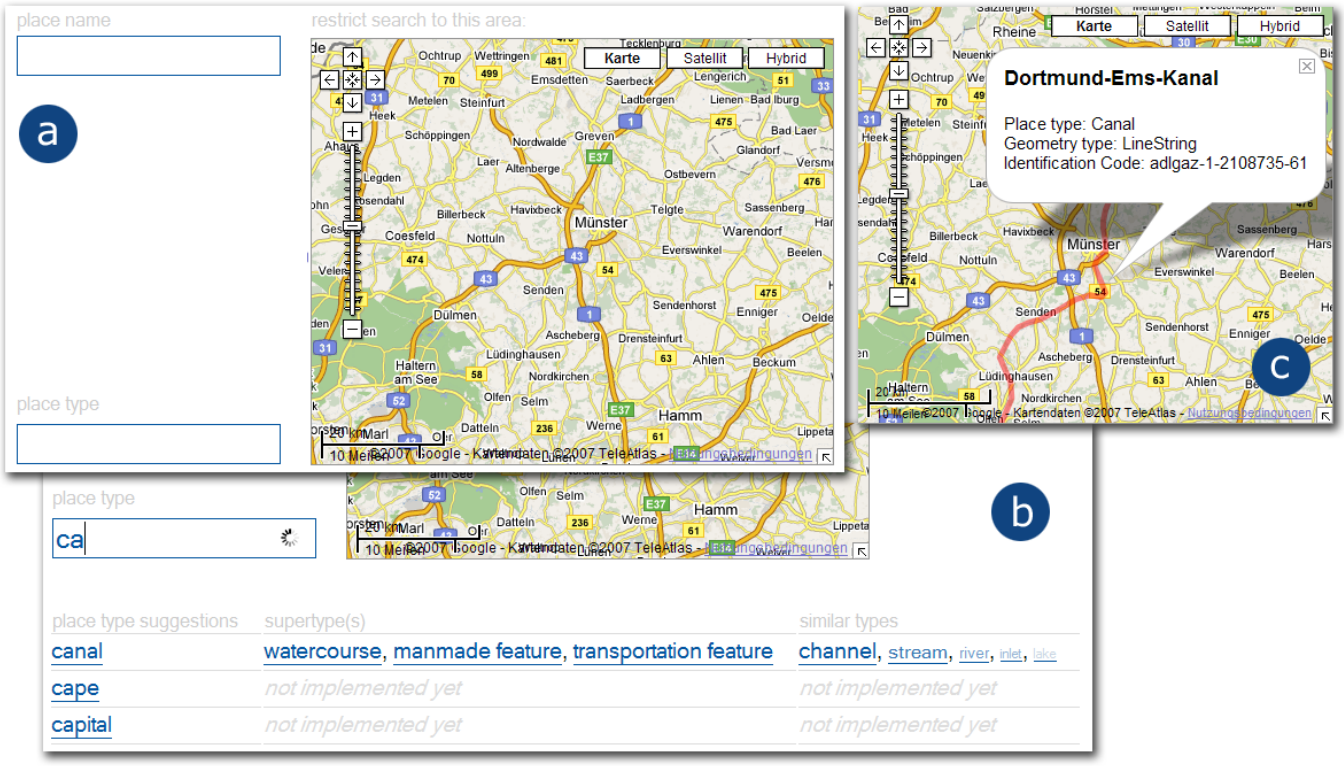

Figure 7. Conceptual design for the gazetteer Web interface: search interface with input fields for place name and type, and map for spatial restriction (a); automatic suggestion of place types during user input (b); display of results as map overlays (c).

While the user adapts his query, results are continuously loaded from the server and displayed on the map. To avoid overloading the map with too many results, query results are only displayed if the number is below a predefined threshold. If the current query returns too many results, a corresponding note is shown on the form. Results are first shown as overlays on the map; the user can then click on them to retrieve detailed information on the feature, as shown in figure 7c. Although there are no human subjects tests on the efficiency, effectiveness, and ease of use of the proposed Web interface yet, the simplification of the search form can be expected to significantly speed up retrieval of the desired results, especially for users who are not familiar with the structure behind the Web interface. The core functionalities of the proposed interface are made possible in the first place by the formal feature type definitions in the ontology and the reasoning methods presented in section 5.1.

5.3 *API Approach*

Comparable to the Web interfaces, current APIs suffer from the fact that the API users have to know *exactly* what they are looking for. Detailed knowledge about the gazetteers internal organization of the feature types is required to compose a query that returns the desired results. In particular, the lack of support for similarity-based queries hampers the usability of the APIs. The user is not provided with features that do not exactly match his query, but that are very similar and could thus be useful. Moreover, support for more complex query functionality that addresses the relationships between feature types, as described in detail within a feature type ontology, would be beneficial. In the following, we will sketch an API that makes use of the multiple inheritance structure of a feature type ontology to provide advanced API functionality.

We use the ADL Gazetteer Protocol (see section 3) as a starting point for our proposed API, as it provides the most enhanced functionality of the existing APIs. Due to its implementation as *XML-over-HTTP*—the current quasi-standard for Web-based APIs—it can easily be combined with other APIs and XML-based data sources. This is also in line with the AJAX-based Web interface proposed in the previous

section, which also relies on XML as the data transfer format. To make full use of the structural information provided by a feature type ontology, the ADL API needs to be extended with functionality to support reasoning and similarity measurement. To reuse existing specifications that have proven useful in practice, we use the DIG[24] interface as a starting point for the reasoning functionality of our gazetteer API.

The DIG interface is an API specification for reasoning in Description Logics (DL) systems (Bechhofer 2003). In particular, it can be used to reason on formal knowledge representations written in OWL. The design of DIG can be compared to the ADL API: it is based on XML-over-HTTP, and the functionality of a concrete DIG instance can be queried by an identification request. The document returned by the server contains information on the supported DL language constructs, so that clients know which reasoning functionality can be performed on this server instance. This is especially important because of the variety of DL languages, i.e. not every DIG server will support all constructs that are part of the specification (the basic constructs are compulsory, however). Once the client is aware of the functionality provided by the server, all further communication is based on two types of documents: *Tells* documents allow the client to build a knowledge base on the server, whereas *Asks* documents allow the client to perform reasoning tasks on this knowledge base.

The original DIG specification has been developed for standard reasoning tasks such as checking for subsumption and disjointness, or queries based on the concept and role hierarchies. To enable similarity measurement, DIG's *Asks* language must be extended by queries that allow for the computation of the similarity of two or more concepts. A combination of such a similarity-enabled DIG version with the ADL API provides all functionality required to implement the Web interface proposed above. In addition, this combined API enables complex queries that go far beyond what is possible with current gazetteers. As an example, we show how a query for *all rivers which have their springs in California* would look in our proposed API. Note that this query is not possible with current gazetteers, as the underlying feature schemas lack information on the relationships between rivers and springs. We assume that the knowledge base on the server is already present and can now be used. The following extract shows the relevant parts of the query document:

```
...
1    <gazetteer-query>
2     <class-query term="River">
3      <property-query term="hasOrigin">
4       <class-query term="Spring">
5        <spatial-relation-query term="inside">
6         <feature-type-query term="FederalState">
7          <name-query term="California" />
8         </feature-type-query>
9        </spatial-relation-query>
10      </class-query>
11     </property-query term="hasOrigin">
12    </class-query>
13   </gazetteer-query>
...
```

The pseudo-code in the example shows how the restrictions for the query are hierarchically defined, based on different types of query commands. First, all features of type *River* are selected (line 2). Then, this result set is further restricted to those rivers which have an origin (line 3) of type *Spring* (line 4). The set of springs under consideration is then restricted by a spatial query to those which lie *inside* (line 5) features of type *FederalState* (line 6) that are named *California* (line 7). The example shows the support for complex queries based on the ontological structure; language constructs based on similarity can be employed accordingly. Comparable to the Web interface (which actually builds upon this API), the improvements in functionality compared to current gazetteer APIs are based on the detailed formal

[24]Description Logic Implementation Group, http://dig.sourceforge.net/

feature type descriptions in the ontology. The information in current thesauri is not sufficiently structured to support detailed queries that make use of subsumption and similarity based reasoning.

What was discussed here for DIG, can also be applied to the similarity enabled iSPARQL Protocol and RDF Query Language (Kiefer *et al.* 2007) developed for the upcoming Semantic Web infrastructure.

6 Conclusions and Further Work

In this paper, we have discussed steps towards the development of a shared feature type ontology, based on a survey of current feature typing schemes and their interfaces. The benefits of such an ontology for Web interfaces, application programming interfaces and a forthcoming gazetteer infrastructure were examined in detail. New kinds of interfaces have been proposed based on the extended type-lookup functionality realized by integrating the ontology with gazetteers and reasoning services. The combination of similarity and subsumption based information retrieval has been portrayed as a promising development towards intuitive and reliable gazetteer interfaces.

In conclusion, current gazetteers provide only suboptimal support for users during the query process. Users require detailed knowledge of the structure behind the gazetteer and its feature type scheme to make full use of its functionality. While some of the difficulties in the query process stem from user interface issues, the lack of expressiveness of the feature type thesauri poses the biggest challenge for gazetteer improvement. We have proposed a distributed feature type ontology, based on formal specifications of both the feature types and the relations among them. The envisioned ontology solves several shortcomings of current gazetteers. It enables distributed gazetteer management, as opposed to the current centralized approach, so that maintenance tasks can be completed by local gazetteer operators. This distributed approach allows for both efficient updates as well as the incorporation of local particularities such as special names for certain places. Second, from the users' perspective, a feature type ontology opens up possibilities for improved gazetteer interaction. Web interfaces can benefit from subsumption and similarity-based search functionality that no longer require the user to know what is meant by a specific feature type: they can refine their initial query based on the interface's suggestions until they find the desired information. New gazetteer APIs can include support for complex reasoning tasks and similarity queries.

Although an ontology-based gazetteer infrastructure promises numerous improvements compared to the current state of gazetteers, most components that are required to put such an ontology into practice are still under development or in the conceptual design phase. First results from human subject tests indicate that the SIM-DL theory produces cognitively plausible results (Janowicz 2007); however, further tests are required to verify the behaviour of SIM-DL. The proposed gazetteer interface (Web interface and API) is currently under development. Human subjects tests for parts of the feature type ontology are currently being prepared and will be followed by tests for the gazetteer Web interface when the implementation is finished. Whether the presented reasoning capabilities and interfaces are useful depends mostly on the underlying ontology as subsumption and similarity is determined with respect to the specifications made there. Further work has to focus on developing such a feature type ontology as a common agreement between several interest groups. To this end, gazetteer researchers and geographers have to agree on how to define a common (and generic) domain model for relevant feature types, but the user's perspective also needs to be taken into account. Tests point out that people tend to mix up certain feature types (such as canals and channels) and topological relations (Riedemann 2005). Additionally, categorization and the perception of similarity depend on cultural background, consequently research from ethnophysiography (Mark and Turk 2003) needs to be taken into account. A follow up workshop of the NCGIA Gazetteer Research & Practice Meeting in 2006 could be a good starting point to debate such issues.

From a formal ontology point of view and taking into account the standardization step proposed by van Assem *et al.* (2004), further work should focus on aligning the presented feature type ontology to top level ontologies such as DOLCE[25].

In terms of software development and infrastructure, the introduced SIM-DL server needs to be extended

[25]Descriptive Ontology for Linguistic and Cognitive Engineering: http://www.loa-cnr.it/DOLCE.html

to handle more expressive description logics. While the current implementation of the identity assumption service for historical places (Janowicz 2006b) mentioned in section 1 uses the ADL FTT for it's similarity assumptions, later versions would also benefit from the feature type ontology.

At last, reviewing existing gazetteers and feature type thesauri led us back to the question of place identity. For example, ADL and Getty list more than 15 alternative or historical names for Istanbul and also point to the former and recent upper level geopolitical units; but what does it actually mean—that Constantinople, Byzantium and Istanbul are the same *place*? Is there a need for a formal theory of identity of places? To what degree does a certain place persist, despite periodic changes in names, geometries and dominions (including cultural and religious aspects)?

Acknowledgments

We would like to thank Linda Hill for giving us a detailed insight into the development and conceptualization of the Alexandria Digital Library Feature Type Thesaurus and for fruitful comments concerning the proposed conceptual design. Discussions with Naicong Li and Catharina Riedemann have shaped our ideas on how to implement the presented feature type ontology. The ideas for a shared gazetteer infrastructure and a new kind of Web interface were outcomes of the NCGIA Gazetteer Research & Practice Workshop held in December 2006 in Santa Barbara, USA. Partial funding for this work came from the SimCat project granted by the German Research Foundation (DFG).

References

ANSI/NISO, 2005, *ANSI/NISO Z39.19 - 2005 Guidelines for the Construction, Format, and Management of Monolingual Controlled Vocabularies* (NISO Press).

Baader, F., Calvanese, D., McGuinness, D.L., Nardi, D. and Patel-Schneider, P.F. (Eds) , 2003, *The Description Logic Handbook: Theory, Implementation, and Applications* (Cambridge University Press).

Bechhofer, S., 2003, The DIG Description Logic Interface: DIG/1.1. In *Proceedings of the DL2003 Workshop*, Rome.

Bittner, T., Donnelly, M. and Smith, B., 2004, Individuals, Universals, Collections: On the Foundational Relations of Ontology. In *Proceedings of the Third Conference on Formal Ontology in Information Systems (FOIS-04)*, A. Varzi and L. Vieu (Eds) (IOS Press), pp. 37–48.

Casati, R. and Varzi, A.C., 1999, *Parts and Places. The Structures of Spatial Representation* (Cambridge and London: MIT Press).

Fielding, R., 2000, Architectural Styles and the Design of Network-based Software Architectures.. PhD thesis, University of California.

Fitzke, Jens; Atkinson, R., 2006, OGC Best Practices Document: Gazetteer Service - Application Profile of the Web Feature Service Implementation Specification Version 0.9.3. Technical report, Open Geospatial Consortium.

Gärdenfors, P., 2000, *Conceptual Spaces - The Geometry of Thought* (Cambridge, MA: Bradford Books, MIT Press).

Gibson, J., 1977, The Theory of Affordances. In *Perceiving, Acting, and Knowing - Toward an Ecological Psychology*, R. Shaw and J. Bransford (Eds) (Hillsdale, New Jersey: Lawrence Erlbaum Ass.), pp. 67–82.

Goldstone, R. and Son, J., 2005, Similarity. In *Cambridge Handbook of Thinking and Reasoning*, K. Holyoak and R. Morrison (Eds) (Cambridge: Cambridge University Press).

Goodman, N., 1972, Seven strictures on similarity. In *Problems and projects*, N. Goodman (Ed.) (New York: Bobbs-Merrill), pp. 437–447.

Gruber, T., 1993, A Translation Approach to Portable Ontology Specifications. *Knowledge Acquisition*, **5**, 199–220.

Guarino, N., 1998, Formal Ontology and Information Systems. In *Formal Ontology in Information Systems*, N. Guarino (Ed.) (Trento, Italy: IOS Press), pp. 3–15.

Hastings, J.T., 2008, Automated Conflation of Digital Gazetteer Data. *International Journal of Geographical Information Science*, **this volume**.

Hepp, M., 2006, Products and Services Ontologies: A Methodology for Deriving OWL Ontologies from Industrial Categorization Standards. *International Journal on Semantic Web & Information Systems (IJSWIS)*, **Vol. 2, No. 1**, 72–99.

Hill, L.L., 2006, *Georeferencing: The Geographic Associations of Information (Digital Libraries and Electronic Publishing)* (The MIT Press).

Hill, L.L., Frew, J. and Zheng, Q., 1999, Geographic Names: The Implementation of a Gazetteer in a Georeferenced Digital Library. *D-Lib Magazine*, **Vol. 5, No. 1**.

Horrocks, I., Patel-Schneider, P.F. and van Harmelen, F., 2003, From SHIQ and RDF to OWL: The Making of a Web Ontology Language. *Journal of Web Semantics*, **1**, 7–26.

ISO, 1986, ISO2788:1986—Guidelines for the establishment and development of monolingual thesauri. Technical report, International Standards Organization.

ISO, 2003, ISO19112:2003, Geographic information—Spatial referencing by geographic identifiers. Technical report, International Standards Organization.

Janée, G., 2006, Rethinking Gazetteers and Interoperability. In *Proceedings of the International Workshop on Digital Gazetteer Research & Practice (Santa Barbara, California; December 7-9, 2006)*.

Janowicz, K., 2006a, Sim-DL: Towards a Semantic Similarity Measurement Theory for the Description Logic ALCNR in Geographic Information Retrieval. In *SeBGIS 2006, OTM Workshops 2006*, R. Meersman, Z. Tari, P. Herrero *et al.* (Eds), 4278 of *Lecture Notes in Computer Science* (Berlin: Springer), pp. 1681 – 1692.

Janowicz, K., 2007, Similarity-Based Retrieval for Geospatial Semantic Web Services Specified using the Web Service Modeling Language (WSML-Core). In *The Geospatial Web - How Geo-Browsers, Social Software and the Web 2.0 are Shaping the Network Society*, Lecture Notes in Computer Science, A. Scharl and K. Tochtermann (Eds) (Berlin: Springer).

Janowicz, K., Kessler, C., Schwarz, M., Wilkes, M., Panov, I., Espeter, M. and Bäumer, B., 2007, Algorithm, Implementation and Application of the SIM-DL Similarity Server. In *Proceedings of the Second International Conference on GeoSpatial Semantics (GeoS 2007)*, Lecture Notes in Computer Science (Springer), pp. 128–145.

Janowicz, K., 2006b, Towards a Similarity-Based Identity Assumption Service for Historical Places. In *Proceedings of the Geographic Information Science, 4th International Conference, GIScience 2006*, M. Raubal, H.J. Miller, A.U. Frank and M.F. Goodchild (Eds), 4197 of *Lecture Notes in Computer Science*, pp. 199–216.

Kiefer, C., Bernstein, A. and Stocker, M., 2007, The Fundamentals of iSPARQL - A Virtual Triple Approach For Similarity-Based Semantic Web Tasks. In *Proceedings of the Proceedings of the 6th International Semantic Web Conference (ISWC)*, Lecture Notes in Computer Science (Springer), pp. 295–309.

Kuhn, W., 2007, An Image-Schematic Account of Spatial Categories. In *Proceedings of the 5th Conference on Spatial Information Theory (COSIT 2007). Lecture Notes in Computer Science*, 4736, Melbourne, Australia, pp. 152–169.

Li, B. and Fonseca, F., 2006, TDD - A Comprehensive Model for Qualitative Spatial Similarity Assessment. *Spatial Cognition and Computation*, **6**, 31–62.

Lutz, M. and Klien, E., 2006, Ontology-based retrieval of geographic information. *International Journal of Geographical Information Science*, **20**, 233–260.

Mark, D.M. and Turk, A.G., 2003, Landscape Categories in Yindjibarndi: Ontology, Environment, and Language. In *Proceedings of the Spatial Information Theory. Foundations of Geographic Information Science, International Conference, COSIT 2003, Ittingen, Switzerland, September 24-28*, W. Kuhn, M.F. Worboys and S. Timpf (Eds), Lecture Notes in Computer Science (Springer), pp. 28–45.

Medin, D., Goldstone, R. and Gentner, D., 1993, Respects for Similarity. *Psychological Review*, **100**, 254–278.

Nedas, K. and Egenhofer, M., 2003, Spatial Similarity Queries with Logical Operators. In *SSTD '03 - Eighth International Symposium on Spatial and Temporal Databases, Santorini, Greece*, T. Hadzilacos, Y. Manolopoulos, J. Roddick and Y. Theodoridis (Eds), 2750 of *Lecture Notes in Computer Science*, pp. 430–448.

Raubal, M., 2004, Formalizing Conceptual Spaces. In *Formal Ontology in Information Systems, Proceedings of the Third International Conference (FOIS 2004)*, A. Varzi and L. Vieu (Eds), 114 of *Frontiers in Artificial Intelligence and Applications* (Amsterdam, NL: IOS Press), pp. 153–164.

Riedemann, C., 2005, Matching Names and Definitions of Topological Operators. In *Proceedings of the Spatial Information Theory. Foundations of Geographic Information Science, International Conference, COSIT 2005, Ellicottville, NY, USA, September 14-18*, A.G. Cohn and D.M. Mark (Eds), 3693 of *Lecture Notes in Computer Science* (Springer), pp. 165–181.

Rodriguez, A. and Egenhofer, M., 2004, Comparing geospatial entity classes: an asymmetric and context-dependent similarity measure. *International Journal of Geographical Information Science*, **18**, 229–256.

Smith, B., 2006, Against Idiosyncrasy in Ontology Development. In *Proceedings of the Formal Ontology in Information Systems (FOIS 2006)*, B. Bennett and C.F. (Eds.) (Eds) (Amsterdam: IOS Press), pp. 15–26.

Sowa, J.F., 2000, *Knowledge Representation: Logical, Philosophical and Computational Foundations* (Pacific Grove, CA: Brooks Cole Publishing Co.).

Studer, R., Benjamins, V.R. and Fensel, D., 1998, Knowledge Engineering: Principles and Methods. *Data Knowledge Engineering*, **25**, 161–197.

Tversky, A., 1977, Features of Similarity. *Psychological Review*, **84**, 327–352.

Uschold, M., 2000, Creating, Integrating and Maintaining Local and Global Ontologies. In *Proceedings of the 14th European Conference on Artificial Intelligence (ECAI)*, Berlin, Germany.

van Assem, M., Menken, M., Schreiber, G., Wielemaker, J. and Wielinga, B., 2004, A Method for Converting Thesauri to RDF/OWL. In *Proceedings of the 3rd International Semantic Web Conference (ISWC2004)*, Hiroshima, Japan.

Winston, M.E., Chaffin, R. and Herrmann, D., 1987, A Taxonomy of Part-Whole Relations. *Cognitive Science*, **11**, 417–444.