ONTOLOGY-BASED QUERY OF TWO DUTCH TOPOGRAPHIC DATA SETS: AN EMERGENCY RESPONSE CASE

S. Zlatanova, M. de Vries, P.J.M. van Oosterom

Delft University of Technology, OTB, GISt, Jaffalaan 9, 2628 BX Delft, The Netherlands
(s.zlatanova, m.e.devries, p.j.m.vanoosterom)@tudelft.nl

Commission IV, WGIV/8

KEY WORDS: databases, integration, query, reasoning, content-based, vector

ABSTRACT:

The integration of geo-information has been greatly advanced by the development of Spatial Information Infrastructures. However, the semantic interoperability, which is essential for the integration of geo-information in open and distributed environments, still not sufficiently supported. One possible way to deal with the problem is to use ontologies to formally describe the meaning of (spatial) data. The use of ontology languages and search techniques is believed to provide means to solve the semantic interoperability and allow machine-automation of data integration, due to the possibility to define semantics explicitly and represent it in a machine-processable way. This semantic interoperability is expected to facilitate information search and query in time-critical situations (as emergency response).

In this paper we verify this hypothesis and present our initial investigations to build and use data ontologies (upper and local). These ontologies describe the data sets (e.g. topographic, utility, cadastre) for general use, independent from the domain of disaster management. They are created and maintained by the respective data providers. The paper presents the ontologies built for two Dutch topographic data sets: GBKN (scale of 1:1000 to 1:2000) and TOP10NL (scale 1:10,000). Both of them are important for disaster management. GBKN is more useful for the field emergency workers, while TOP10NL has fewer details and therefore is more appropriate for decision makers at higher levels. A significant amount of the content from both the data overlaps (with different level of detail), but some information is available only in one of the data sets. In addition to these local data ontologies that are bound to a specific data set, there is a upper data ontology that mediates between the local data ontologies. Using these ontologies, we demonstrate that the user can specify search criteria for spatial information without having to know the underlying data structures of the data sets. The paper elaborates on the developed prototype application, which combines ontology-based query of semantically different data sets, with existing tools for geo-information processing (i.e. GeoTools).

1. INTRODUCTION

Search, access and integration of geo-information data is often quoted as one of the critical issues in emergency response (Borculo et al 2005, Diehl et al 2006, Kevany, 2008, Brecht 2008, Zlatanova et al 2007). The integration of data has been significantly advanced in the last years by developing Spatial Information Infrastructures for emergency management (Parker et al 2008, Grothe et al 2008). Many governments have encouraged development of ‘critical infrastructure’ databases (e.g. the Department of Homeland Security in USA, Geonovum in the Netherlands). However, these developments reflect syntax integration, i.e. the semantic (meaning) of the data sets may remain unclear. Spatial data in emergency response are usable only if they are relevant for a given situation and this requires explicit semantic support (Pundt 2008). Semantic challenges although addressed are still not sufficiently investigated (Hess and de Vries, 2006, Dolbear and Hart, 2008, Fan and Zlatanova 2010, Klien et al 2006, Xu and Zlatanova, 2007, Xu 2007).

Knowledge about schemas, data structures, formats and meta data are often not sufficient for the purposes of emergency response. Data sets, also spatial data sets, are usually created for a specific purpose and under the assumption that the users share a common understanding of certain concepts and share a common vocabulary. This is often not the case in managing emergencies, where actors from many different occupational backgrounds and expertise have to operate together. Moreover these actors are generally not geo-information specialists. A more formal approach, as developed in disciplines such as knowledge engineering, ontology and object-oriented modeling, is required (Harmelen 2008). Formal semantic approach allows deciding whether different domain models (or even models within one domain) are or can be harmonized. Such an approach allows machine handling of spatial information, which is expected to speed up search and integration of data. The important technology progress that has been made in the discipline of knowledge engineering (UML, ontology, semantic web) enables further knowledge formalization in a practical manner (Lieberman and Goad, 2008, Lemmens 2008).

Some of the most important issues to be considered in semantic/data discovery for emergency response domain are (Zlatanova et al 2006, Fan and Zlatanova 2010):

- Building of a emergency response domain ontology or link the domain ontology of emergency response institutions in a network of ontology
- Building of ontology/ linking existing ontologies for spatial data in preparation for knowledge discovery and emergency knowledge transaction processing.
- Developing context-aware engines (rule-based) and agents for query and analysis with respect tasks of the emergency response users (considering also to the type of the front-end devices and communication channels used).
- Investigation, adaptation and development of converters to well-known Web standards and formats.

- Developing knowledge-based systems for browsing and analysis in a distributed data environment.
- Investigating and developing intelligent semantic-based engines and corresponding translators for semantic search and analysis.

In this paper we concentrate on the second and (partially) the third aspect of discovery of data. For the scope of this paper it is assumed that the emergency responders have a common domain (i.e. shared concepts and shared vocabulary). The paper presents a practical approach how to search for information in spatial data sets, using the vocabulary of the emergency response domain.

The paper is organized in six sections. The next section presents a simple use case to illustrate the problem. Section 3 presents the data sets and the tools used for the implementation. Section 4 presents the prototype and elaborates on the workflow. Section 5 concentrates on the ontology use. Section 6 discusses the prototype and outlines future research and developments.

2. USE CASE

Crisis management in the Netherlands involves four primary emergency respond institutions: fire brigade, police, medical care and local authorities (Borkulo el al 2005, Diehl et al 2006). In the last couple of years a large number of projects have been devoted to developing systems for command and control, allowing emergency responders to view and share existing geo-information (topographic, cadastral, utility, etc.) and to record in-situ geo-information (affected areas, locations of shelters, damages, plume, etc.) (Scholten et al 2008, Dilo and Zlatanova 2008). Current systems make this information available to all participants in the crisis management no difference what the role of an emergency responder is (Xu et al 2008, Jacobs et al 2009). Figure 1 illustrates the long list of data sets (right section) available in one C&C system. In many cases this may lead to overloading, confusion and misinterpretation, which may affect the decision-making.. The following use case describes how this can be done in a more efficient manner.

Figure 1:Example of the C&C system Eagle One (courtesy Geodan, www.geodan.nl)

A large fire is indicated in the area of the Rotterdam harbour. According to the provided information, the level of emergency is already scaled to '2' (indicating a more than average serious incident) and a Regional Operational Team (ROT) is formed at the Command and Control Centre in the Rotterdam Port Office. The fire officer in duty is requested to drive one of the specialized fire trucks to the location of the incident. He log-ins in the system and selects the incident and the task he has to perform, i.e. FightFire, (see Zlatanova 2010). The system has ontology-based component and can easily provide only the data that is needed for the task. The fire fighting team gets a large-scale map of the area (scale 1:1,000) with the (detailed) route to the location of the fire, the access entrances (access maps) to the industrial area and a map with the fire hydrants. The officer in duty has also a special menu where he can search for additional information. He types 'gas pipeline' and he gets the gas utility network within the area of the incident.

At the same time the ROT discusses the effects of the fire and analyses the need for an evacuation from certain neighbourhoods (task TacticalLead, As in Zlatanova 2010). The discussion is carried out on a smart board showing the entire safety region Rijnmond and information about vulnerable objects in the area (as hospitals, schools, etc.) (i.e. topographic map 1:10,000 and a risk map). After one hour the fire is still not under control. The ROT takes a decision to scale the emergency level up to 3. Some of the city areas have to be evacuated. With the new emergency level the map view is automatically updated, showing the neighbouring safety regions. The ROT obtains also the special menu for search of any relevant spatial information for the incident. One of the ROT members types 'roads', 'shelters', 'helicopter landing places' and 'large open parkings'. The system delivers a map with all these objects.

To deal with this case, the geo-information used is: two topographic maps at different scale, risk map, accessibility map and map of hydrants. It should be noticed that the users are not obliged to have any knowledge on the source data sets. The ontology tool provides the initial geo-information with respect to the task or allows them to search for data using the shared vocabulary. Thus the fire brigade officer gets a large scale map and the ROT small-scale map with respect to the task (upon log-in in the system). Any further requests for data can be specified by simply typing the textual name of the required objects. These objects are found by the system in the corresponding data sets and shown as geometry with their attributes in the graphical section of the user interface.

This paper continues further with the ontology tool allowing for search of spatial data using the shared vocabulary of the emergency responders. As mentioned previously, for the scope of this paper it is assumed that the ontology for emergency response domain is available. For simplicity we use only two topographic data sets, i.e. the maps in scale 1:1,000 and 1:10,000 to demonstrate the developed application.

3. DATA SETS AND TOOLS USED

The Large Scale Topographic map (GBKN, 1:1,1000 or 1:2,000) and the Topographic Map 1:10,000 (TOP10NL) data sets are both topographic data sets. Since they are developed for different purposes and collected by different surveying rules, (GBKN by consortia often including municipalities and TOP10NL by the geo-department of the Cadastre), many semantic differences exist, despite the fast that they represent the same phenomena, i.e. the topography of a specific area. For example, the classification of many objects such as buildings, streets is different, some objects might be not present in one of the data sets, etc. (Uitermark et al. 1999). The data sets were examined manually as well as looking at their conceptual models and comparing them to the model of NEN3160. The ontologies presented here are a bit simplified version of the original data considering the predefined vocabulary for emergency responders.

The data ontologies in this case study are specified using the Web Ontology Language (OWL). OWL is 'designed for use by applications that need to process the content of information instead of just presenting information to humans' (OWL 2004). It aims at providing a standard language for the representation of ontologies on the World Wide Web. It is one of the most used ontology representation languages (Baader et al. 2003). OWL has three increasingly-expressive (what can we express using the language) but decreasingly-decidable (by decidable we mean the time spent on the inference of the represented knowledge is finite) sublanguages: OWL Lite, OWL DL and OWL Full. OWL-DL, of which the semantics is based on Description Logic (DL) (Baader et al. 2003), supports the maximum expressiveness without losing the computational completeness and decidability of reasoning. OWL DL will be used as the ontology representation in this paper. The remainder of the paper will simply use OWL to refer to OWL DL. More on OWL can be found in 'OWL Web Ontology Language Semantics and Abstract Syntax' (Patel-Schneider et al. 2004).

The prototype is based on Protégé API, GeoTools API and Java. Protégé (protege.stanford.edu/) is a development environment for creating ontologies. It can connect to an ontology reasoning engine (for instance, RacerPro), send the ontology to the engine and get the result back from the reasoning engine (for consistency checking, class hierarchy inferencing and so on). The Protégé OWL API (protege.stanford.edu/plugins/owl/api/) is a Java programming interface, which is used to manipulate the ontologies that are represented in OWL. The OWL file is parsed and converted into Java objects. Actions made on the corresponding Java objects can be written back to the OWL file. GeoTools (geotools.codehaus.org/) is an open source Java code library which provides standards compliant methods for the manipulation of geospatial data.

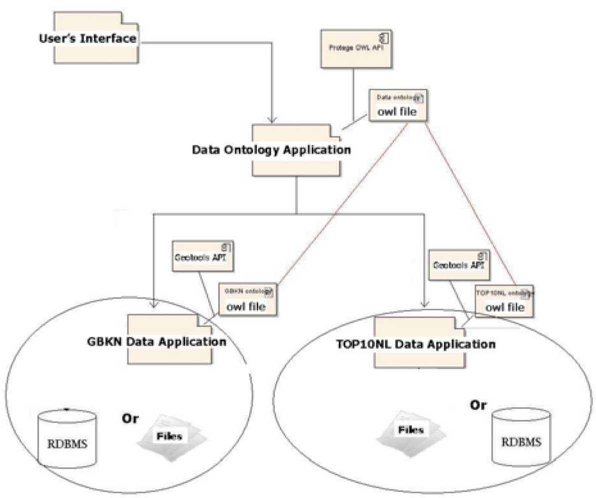

Figure 2: Prototype system architecture

4. SYSTEM OVERVIEW

The system architecture consists of a User Interface (UI), Data Ontology Application (DOA) and Local Data Applications (LDA), which can be located on different servers in files or in database management systems (DBMS)(Figure 2). In our prototype we have two LDAs (for GBKN and TOP10NL).

UI provides the interface to search and view data. Generally, it can be realized as a web or a stand alone rich client application. The current prototype uses a Java class GUI (Figure 3). The class creates the graphical user interface allowing a user to input his/her query as a word. The interface consists of three sections: a list with earlier used words/terms, a section to type the word and a visualization section. For simplicity, the upper left part suggests a list of words for query. The list of words is a kind of indication to the user what is available in the ontology and is based on the most recently used words for searching. If a term is not found, it is not registered in the list. The list of words is kept in a text file, which can be modified by the application designer. The list is kept updated as each new word used for search is added to the list. Please note that list does not contain all the words as in the shared emergency vocabulary (as it can become too long). Theoretically, an incident can begin with an empty list. The lower left section of the user interface is a text field, where the user can type a word, i.e. name of object, which has to be found in the data sets. The query should be a single word, such a 'school', 'post_office', etc. The search starts after the submit button at the lower left section is pressed. The right section of the interface is for displaying the result of the geometry.

The DOA is responsible for the search of the requested information (word, term) in the available ontologies. The DOA receives the word (string) for searching and reads the upper data ontology (the owl file), maps the upper data ontology into the logical model (i.e. java classes) and matches the word against the logical model.

- If there are no matches in the logical model with the word, the DOA returns to the UI 'not found in the upper data ontology'.
- If there is a match (or are matches) in the logical model with the word, the DOA finds a quadruple set (DataSetName, GeoObjectName, AttributeName, Value).

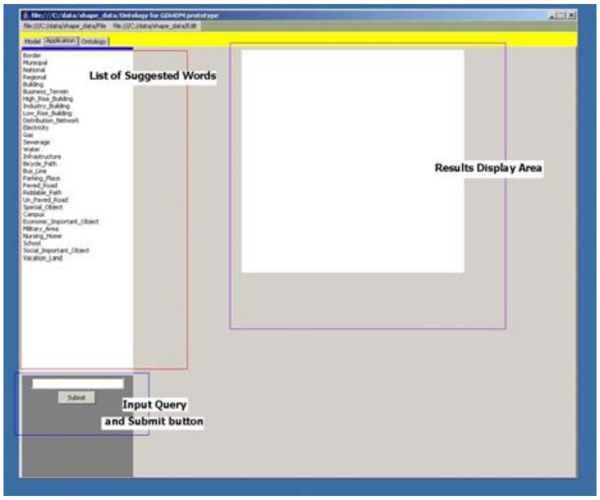

Figure 3: User Interface Application

The DOA searchers for a set (DataSetName, GeoObjectName, AttributeName, Value) in the upper data ontology. When the set (DataSetName, GeoObjectName, AttributeName, Value) is found, the DOA knows which data set may contain the geometry that matches the user's query by the first parameter 'DataSetName'. Then the DOA sends the triple set

(GeoObjectName, AttributeName, Value) to the relevant LDA according to the ‘DataSetName’.

The LDA receives the triple set (GeoObjectName, AttributeName, Value) and reads the local data ontology (an owl file), maps the local ontology into a logical model. Then LDA translates the triple set into another triple set (GeoObjectName_Local, AttributeName_Local, Value_Local) according to the local data ontology. When the LDA receives the triple set, there is still a possibility that the terms used in the triple are not those supported by the underlying data source. At this stage, the LDA translates the triple set into (GeoObjectName_Local, AttributeName_Local, Value_Local) where the terms used are supported by the underlying data source.

The next step is query of the local data source (as file or in DBMS) by the triple set (GeoObjectName_Local, AttributeName_Local, Value_Local). LDA finds the geometry from the underlying data source. There might be cases where no geometries can be found in the data source (e.g. certain objects does exist in the given area), the LDA returns ‘not found in the data source’ to the UI, and otherwise it returns the geometry.

Apparently, when a new data set is used, a new LDA should be created in such a way that it receives a triple set (GeoObjectName, AttributeName, Value) from the DOA, translates the triple into (GeoObjectName_Local, AtributeName_Local, Value_Local) according to the local data ontology and finds geometry in the data source through the triple set (GeoObjectName_Local, AtributeName_Local, Value_Local). In this way, there is no need to change the existing applications when new data source comes. The only change required is the owl file, which contains the upper data ontology. The next sections explain the evolvement of the ontology.

5. BUILDING THE ONTOLOGIES

For this prototype, the upper data ontology is built based on the existing local data ontologies at hand (in this case GBKN and TOP10NL) and it will grow if new data sets are used. Every new data set will require the creation of a local data ontology for it. It will be the ontology maintainer’s task to ensure that each item in the local ontology has a corresponding representation in the upper data ontology. In other words, the new local data ontology will first be mapped to the upper data ontology as good as possible. Those object classes that cannot be mapped to existing classes in the upper data ontology will be added as new classes (and thereby extending the upper data ontology).

To illustrate the development of the upper data ontology, we will use only the object class ‘Building’. As soon as a new object class needs to be included, the upper data ontology has to be extended. Let us assume that at certain moment there is only ‘Building’ and its attributes in the upper data ontology. Following an Object-Oriented method to model the real world, we use classes (and attributes). Geographic objects belonging to the same class can be distinguished from each other by having different attribute values. For instance, a ‘hospital’ and a ‘school’ may belong to the same class ‘building’. ‘Building’ class has an attribute ‘Building_Occupancy’ which specifies the function (or the main purpose of usage of the building). Thus ‘hospitals’ can be distinguished from ‘schools’ on the basis of ‘Building_Occupancy’ attribute. For hospitals it is ‘medical’ and for schools it is ‘educational’. The upper data ontology follows similar approach, i.e. the geographical world is represented by a triple set (Class, Attribute, Value). The ‘value’ indicates the allowed data values specified as enumeration types of code lists.

In order to structure well the ontology, a root class ‘Data Ontology’ with two top level subclasses, GeoObjectName, AttributeName, is created. All the classes representing the real world feature are created as subclasses of GeoObjectName. All the attributes belonging to the real world features are made as the subclasses of AttributeName. The values, the attribute can take, are given as the subclasses of each of the attributes. The relation between the class and the attribute that belongs to the class is done through making a owl:objectProperty by setting the domain to be the class and the range to be the attribute. For instance, the real world feature ‘Building’ is modeled as a class having three attributes as shown in Figure 4.

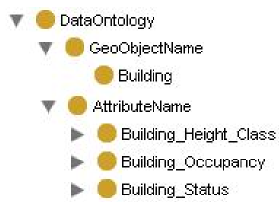

Figure 4: Building and its three attributes

Building is created as a subclass of GeoObjectName. Building’s three attributes are made as the subclasses of AttributeName. One of the building’s attribute’s (Building_Height_Class) values is given as the subclass the attribute. The Building_Height_Class is a property of Building, (i.e. defined as owl:objectProperty). Figure 5 and Figure 6 illustrate the class, attributes and its values.

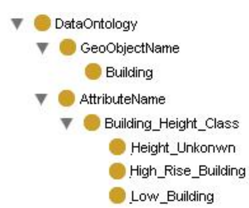

Figure 5: The attribute Building_Height_Class and its values (enumeration type or code list)

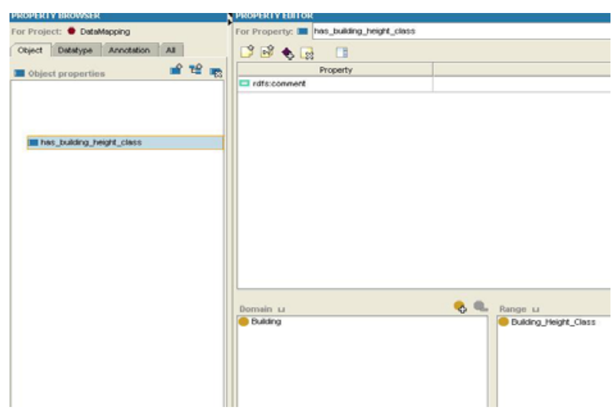

Figure 6: Building_Height_Class is related with Building by owl:objectProperty (in Protégé)

Similar to the structure of the upper data ontology, the local data ontology has also a root class and it is given the name of the data set, such as ‘TOP10NL’ and ‘GBKN’ in our case. Each of the local data ontologies also has a class GeoObjectName_Local and AttributeName_Local (similar to the two top level classes in the upper data ontology). The subclasses, however are specific for the different local data ontologies

6. CREATING AND USING MAPPINGS BETWEEN COMMON AND LOCAL ONTOLOGIES

Each of the data source is provided with a local data service. When a new data set is used, a new local data service with a local data ontology should be available. Thus the number of local data services equals to the number of data sources. Each of the Local Data Services is developed in such way that it takes a triple set (GeoObjectName, AttributeName, Value) from the Data Ontology Service, translates the triple set into (GeoObjectName_Local, AttributeName_Local, Value_Local) according to the local data ontology. By the triple set (GeoObjectName_Local, AttributeName_Local, Value_Local), the local data service finds/gets geo objects (with geometry). As mentioned previously, in our demo application, there are two data sources: GBKN and TOP10NL (i.e. shape files). In our case there are two local data services: one for GBKN and one for TOP10NL.

As mentioned above, through the Protégé OWL API the owl file is mapped as a logical model (OWLModel) in memory, which is easier to manipulate. Four important methods are used for the matching the ontologies: 1) OntologyTools() is to initiate the ontologyModel (a logical model of the owl file in memory), 2) receiveObjectName(String) receives the word (a Java String object) for searching, 3) getQuadruple(String) searches the user’s query against the logical OWL model and returns (DataSetName, GeoObjectName, AttributeName, Value), and 4) send2LocalDataService sends (GeoObjectName, AttributeName, Value) to corresponding local data service according the ‘DataSetName’.

As described for the upper data ontology, the AttributeName has two levels of subclasses: the first level of subclasses corresponds to the attribute names, the second level of subclasses correspond to the possible values for each attribute. The second level of subclasses will be matched against the user’s query, thus we call them ‘searchable items’. After the Java class initiates the logical model out of the owl file, the user’s query will be matched against the searchable items. The match is done through substring matching (in the getQuadruple(String) method), i.e. if the user’s query is a substring of the searchable item’s name, we consider it as a matching against the user’s query (if none is found, the method return ‘none’). The method getQuadruple(String) is to return the set (DataSetName, GeoObjectName, AttributeName, Value). We will explain how the set is obtained through the method:

Each of the searchable items are with prefix LocalData setName, the method getQuadruple(String) will extract the LocalDataSetName as DataSetName and the rest as the Value. When the user’s query is found as one (several) of the searchable-items, the method getQuadruple(String) will find its super class as AttributeName, which belongs to the first level subclass of AttributeName. Since each of first level sublcasses of AttributeName is related to a subclass of GeoObjectName by owl:objectProperty, it is easy for the getQuadruple(String) to find a subclass of GeoObjectName as ‘GeoObjectName’, which relates to the AttributeName.

7. DISCUSSION

This paper presented a practical application based on ontology for search of geo-information in two data sets. The application allows querying spatial data sets by words from a predefined shared vocabulary for emergency response. The approach ensures that spatial information can be searched by a wide group of none geo-specialist having no knowledge on the data structure, syntax or semantics of the data. The system operates on the set of local data ontologies, which are linked via the upper data ontology. The method looks very promising but several aspects need further careful consideration.

As mentioned previously, the upper data ontology is progressively extended when a new data set is added to the data sets available for the emergency response application. This step is currently done manually (assuming that a data ontology manager will be available for management of the ontology). Further investigations are needed to clarify whether this update can be automated.

Bearing in mind the large amounts of data sets in use during emergency, it is perhaps unrealistic to consider only one upper data ontology. The data sets can vary largely in scope and semantic description, which may create difficulties in matching new local data ontologies with the upper data ontology. In this case the upper data ontology will quickly grow in size and will be inefficient for emergencies. Other approaches for mapping and search in local data ontologies must be investigated.

Another critical aspect is the search in the local data ontologies. As implemented at the moment, the data ontology contains information about which class belongs to which data set. Practically the data ontology is a union of the local data ontologies. Further investigations and developments are needed for more flexible representations, e.g. by creating additional ontology with the data sets.

The next logical step is providing ontology-based context-aware services, which also needs a formalization of users (tasks) and context. The first default information as described in the use case, can be obtained automatically by defining rules and using an ontology reasoner.

ACKNOWLEDGEMENTS

This research was done in the framework of the RGI-239 project Geographical Data infrastructure for Disaster management (GDI4DM) (http://www.gdi4dm.nl/), supported by the Dutch program Space for Geo-Information. Herewith, we would like to thank all project partners for their input, critical comments and support.

REFERENCES

Baader, F., Horrocks, I., and Sattler, U., 2003, Description logics as ontology languages for the semantic web. In Lecture Notes in Artificial Intelligence

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, UK.

Borkulo, E van, H. J. Scholten, S. Zlatanova and A van den Brink, 2005, Decision making in response and relief phases, in: van Oosterom, Zlatanova & Fendel (eds.), Geo-information for disaster management - late papers, pp. 47-54.

Brecht, H., 2008, The application of geo-technologies after the hurricane Katrina, in: Zlatanova&Nayak, 2008, Remote Sensing and GIS technologies for monitoring and prediction of disasters, Springer-Verlag, Berlin Heidelberg, pp. 25-36

Diehl, S., Neuvel, J., Zlatanova, S. and Scholten, H. 2006, Investigation of user requirements in the emergency response sector: the Dutch case, in: Second Symposium on Gi4DM, 25-26 September, Goa, India, CD ROM, 6p.

Dilo, A. and Zlatanova, S. 2008, Spatiotemporal data modelling for disaster management in the Netherlands, in: Van der Walle, Song, Zlatanova&Li (eds.), Information Systems for Crisis Response and Management, Joint ISCRAM-CHINA, Gi4DM Conference, 4-6 August, 2008, Harbin, 4–6 August, China, pp. 517–528

Dolbear, C. and Glen Hart, 2008, Opportunities and challenges in Exploring semantics as an aid to information integration: a national maping perspective, in: van Oosterom&Zlatanova (eds.) Creating Spatial Information Infrastructures, CRCpress (Taylor&Francis group), Boca Raton, London, New York, pp. 89-101

Fan, Z. and S. Zlatanova, 2010, Exploring ontology potential in emergency response, Proceedings of the 6th Gi4DM, 2-4 February, Torino, 2010, CDROM, 6 p.

Grothe, M.J.M, H.C. Landa and J.G.M. Steenbruggen, 2008, The value of Gi4DM for transport and water management, in: Zlatanova&Li Geospatial Information technology for emergency response, ISPRS book series, Taylor & Francis, London, Leiden, pp: 313-328

Guarino, N., and P. Giaretta (1995) Ontologies and knowledge bases: Towards a terminological clarification, Towards Very Large Knowledge Bases: Knowledge Building and Knowledge Sharing, IOS, pp. 25-32.

Harmelen, F. van, 2008, Semantic Web Technologies as the foundation for the Information Infrastructure, in: van Oosterom&Zlatanova (eds.) Creating Spatial Information Infrastructures, CRCpress (Taylor&Francis group), Boca Raton, London, New York, pp. 37-52

Hess, C. and M. de Vries, 2006, From models to data: A prototype Query Translator for the cadastral domain, in: Computers, Environment and Urban Systems, Volume 30, pp. 529-542

Jacobs, C, A. Riedijk, A. Scotta, P. Brooijmans and H.J. Scholten, 2009, Application of spatial data infrastructure and GIS for disaster management, in: Krek, Rumor, Zlatanova&Fendel, CRCpresss (Taylor & Francis group), London, pp. 277-285

Kevany, M., 2008, Improving geospatial information in disaster management through action on lessons learned from major events, in: Zlatanova&Li (eds) Geospatial Information technology for emergency response, ISPRS book series, Taylor & Francis, London, Leiden, pp: 3-19

Klien, E., M. Lutz and W. Kuhn, 2006, Ontology-based discovery of geographic information services: an application in disaster management, Computers, Environment and Urban Systems, 30 (2006) 102–123.

Lemmens, R., 2008, Using formal semantics for services within the Spatial Information Infrastructure, in: van Oosterom&Zlatanova (eds.) Creating Spatial Information Infrastructures, CRCpress (Taylor&Francis group), Boca Raton, London, New York, pp. 103-118

Lieberman, J. and C. Goad, 2008, Geosematic Web standards for the Spatial Information Infrastructure: nice to have or hopeless without?, in: van Oosterom&Zlatanova (eds.) Creating Spatial Information Infrastructures, CRCpress (Taylor&Francis group), Boca Raton, London, New York, pp. 119-128

OWL 2004, OWL Web Ontology Language, available at http://www.w3.org/TR/owl-features/, last visited January 2010.

Parker, C.J, R. MacFarlane and C. Phillips, 2008, Integrated emergency management: Experiences and challenges of a national geospatial information provider, Ordnance Survey, in Zlatanova&Li (eds) Geospatial Information technology for emergency response, ISPRS book series, Taylor & Francis, London, Leiden, pp: 275-310

Pundt, H., 2008, The semantic mismatch as limiting factor for the use of geospatial information in disaster management and emergency response, in: Zlatanova&Li (eds.) Geospatial Information Technology for Emergency Response. Taylor & Francis Group London, pp. 243–255.

Scholten, H., S. Fruijter, A. Dilo, and E. van Borkulo, 2008, Spatial Data Infrastructure for emergency response in Netherlands, in: Nayak&Zlatanova (eds) Remote Sensing and GIS technology for monitoring and prediction of disaster, Environmental Science and Engineering. Springer-Verlag, Heidelberg, pp 177–195

Patel-Schneider, P.F., P. Hayes and Ian Horrocks, 2009, OWL Web Ontology Language Semantics and Abstract Syntax, available at: http://www.w3.org/TR/owl-semantics/ (last accessed February 2009)

Uitermark, H., P. van Oosterom, N. Mars, and M. Molenaar, 1999, Ontology-based Spatial Data Set Integration. In proceedings of STDBM'99, workshop on Spatio-Temporal Database Management, Edinburgh, Scotland, september 10-11, pp. 60-79

Xu, W., 2007, Geo-information and formal semantics for disaster management, PhD Research Proposal, GISt report 46, OTB, TUDelft, available at www.gdmc.nl/publications (last accessed February 2009), 32p.

Xu, W., A. Dilo, S. Zlatanova and P. van Oosterom, 2008, Modelling emergency response processes: Comparative study on OWL and UML, in: Van der Walle, Song, Zlatanova and Li (Eds.); Information systems for crisis response and management, Joint ISCRAM-CHINA Gi4DM Conference, August 2008, Harbin, pp. 493-504

Xu, W. and S. Zlatanova, 2007, Ontologies for Disaster Managmement, in: Li, Zlatanova&Fabbri (Eds.) Geomatics Solutons for Disaster Management, Lecture Notes in Geoinformation and Cartography, Springer-Verlag Berlin, Heidelberg, pp. 185-200

Zlatanova, S., 2010, Formal modelling of processes and tasks to support use and search of geo-information in emergency response, Proceedings of Climate Change and Disaster management, MapIndia, 19-22 january, Delhi, India, 8.p

Zlatanova, S., D. Holweg and M. Stratakis, 2007, Framework for multi-risk emergency response, in: Tao&Li (Eds.) Advances in Mobile Mapping Technology, Taylor&Francis, London, ISPRS Book Series, pp. 159-171

Zlatanova, S., P. van Oosterom and E. Verbree, 2006, Geo-information supports management of urban disasters, Open House International, Vol. 31, No.1, March 2006, pp.62-79