DOKUZ EYLÜL UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

SEMANTIC GEOGRAPHICAL INFORMATION SYSTEM BASED DISASTER MANAGEMENT SYSTEM

by
Can AYDIN

July, 2014
İZMİR

SEMANTIC GEOGRAPHICAL INFORMATION SYSTEM BASED DISASTER MANAGEMENT SYSTEM

A Thesis Submitted to the
Graduate School of Natural and Applied Sciences of Dokuz Eylül University
In Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy in Geographical Information Systems, Geographical Information Systems Program

by
Can AYDIN

July, 2014
İZMİR

Ph.D. THESIS EXAMINATION RESULT FORM

We have read the thesis entitled **"SEMANTIC GEOGRAPHICAL INFORMATION SYSTEM BASED DISASTER MANAGEMENT SYSTEM"** completed by **CAN AYDIN** under supervision of **PROF. DR. VAHAP TECİM SUPERVISOR** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Doctor of Philosophy.

Prof. Dr. Vahap TECİM

Supervisor

Prof. Dr. Cahit HELVACI

Thesis Committee Member

Assoc. Prof. Dr. Murat KOMESLİ

Thesis Committee Member

Assoc. Prof. Dr. S. Savaş DURDURAN

Examining Committee Member

Asst. Prof. Dr. M. Güven KOCAK

Examining Committee Member

Prof.Dr. Ayşe OKUR

Director

Graduate School of Natural and Applied Sciences

ACKNOWLEDGEMENTS

First, I want to thank my advisor Professor Dr. Vahap TECİM, for his valuable suggestions, encouragement, patience and support during this study.

I would like to thank my thesis tracking committee members Professor Dr. Cahit HELVACI and Assistant Professor Dr. Murat KOMESLİ and Assistant Professor Dr. Murat Osman ÜNALIR for their contributions to this study and sharing their ideas during the development of this thesis.

I would also thank to Dr. M. Güven Koçak and Dr. Kutan KORUYAN for their precious comments on thesis.

I want to give special thanks to my parents Hikmet AYDIN and Filiz AYDIN for their endless support, patience and encouragement way along the graduate student life.

Can AYDIN

SEMANTIC GEOGRAPHICAL INFORMATION SYSTEM BASED DISASTER MANAGEMENT SYSTEM

ABSTRACT

Disasters are events that cause substantial destruction and loss of lives due to natural or other reasons. Disaster managers need spatial data for managing the disaster efficiently. Often the required spatial data are distributed and heterogeneous, and several data sources need to be combined in order to derive the information. This is accomplished with ontologies by the use of semantic web technologies. Although disaster ontologies are being developed in a variety of domains, literature shows that there is still considerable gap in earthquake domain. This work proposes a disaster geo-ontology for earthquake domain. For this purpose we conceptualized earthquake and disaster management terms, designed ontology, and integrated spatial operations into ontology by using geo-ontology as upper-level. Earthquake damage estimation within the first few hours of devastating earthquake is very substantial from the point of decision making and planning after disaster. Recently; with the development of earthquake engineering, earthquake damage estimation is determined by using strong ground motion network. However, several cities are densely populated under earthquake risk. Besides, several cities do not have strong ground motion network which is costly and needs advanced technological systems. This work proposes to estimate earthquake damage at early post-disaster by only using data sources such as up-to-date earthquake source parameter, population, building quality, roads and bridge locations. But this information is distributed and cannot resolve semantic heterogeneity problems in data sharing. For this reason, deriving damage and loss estimation which is based solely on the source parameter of earthquake is very challenging task. Disaster ontologies are used for ensuring semantic interoperability and description logic (DL) rules are defined for damage estimation. Finally, parliament triple store was used for storing and querying disaster ontology and damage assessment maps were provided for disaster managers to respond the disaster swiftly.

Keywords: Semantic web, Geographical Information System, disaster management, Ontology, description logic

ANLAMSAL COĞRAFİ BİLGİ SİSTEMLERİ TABANLI AFET YÖNETİM SİSTEMİ

ÖZ

Afetler doğal ve doğal olmayan sebeplerle azımsanmayacak derecede can ve mal kayıplarına sebep olmaktadır. Afet yöneticileri afeti etkin bir şekilde yönetebilmek için mekânsal veriye ihtiyaç duyarlar. Çoğunlukla ihtiyaç duyulan bu mekânsal veri dağıtık ve heterojen bir yapıdadır ve bilgi çıkarımı yapmak için birçok veri kaynağını bir araya getirmek gerekmektedir. Bu problem anlamsal web teknolojileri tarafından ontolojiler kullanarak çözülmektedir. Afet ontolojileri birçok alanda kullanılıyor olmasına rağmen literatür incelendiğinde; deprem alanında herhangi bir çalışmanın yapılmadığı görülmüştür. Bu çalışmada deprem alanında coğrafi tabanlı bir afet ontolojisi önermekteyiz. Bu amaçla deprem ve afet yönetimi terimler kavramsallaştırılacak, ontoloji tasarımı yapılacak, ve coğrafi ontolojileri üst düzey ontoloji olarak mekânsal işlemleri gerçekleştirebilmek için entegre edilecektir. Yıkıcı bir depremin ilk birkaç saatinde deprem hasar tahmini yapabilmek afet sonrası planlama ve karar verme açısından çok önemlidir. Son zamanlarda deprem mühendisliğinin gelişmesiyle beraber deprem hasar tahmini kuvvetli yer hareketi ağları kullanılarak belirlenmektedir. Oysaki nüfusu yoğun birçok şehir deprem riski altındadır. Bununla birlikte, risk altındaki birçok şehirde yüksek teknolojik sistemler gerektiren ve maliyetli olan kuvvetli yer hareketi ağı bulunmamaktadır. Bu çalışma güncel deprem parametreleri, nüfus, bina kalitesi, yol ve köprü konumları gibi verileri kullanarak deprem sonrası depremin verdiği hasarı tahminlemeyi önermektedir. Fakat bu bilgiler dağıtık ve verilerin entegrasyonu esnasında anlamsal heterojenlik problemi çözülememektedir. Bu amaçla, sadece deprem parametrelerine dayanarak hasar ve kayıp tahmini ortaya çıkarmak zorlu bir görevdir. Afet ontolojileri anlamsal birlikte çalışabilirliği sağlamak için kullanılmıştır ve hasar tahmini yapabilmek için de betimleme kuralları tanımlanmıştır. Son olarak afet ontolojisinin saklanması ve sorgulanabilmesi için Parliament triple store kullanılmış

ve afet yöneticilerinin afete hızlı müdahale sağlaması için hasar değerlendirme haritaları sağlanmıştır.

Anahtar kelimeler: Anlamsal web, Coğrafi Bilgi Sistemleri, afet yönetimi, Ontoloji, betimleme mantığı

CONTENTS

Page

Ph. D. THESIS EXAMINATION RESULT FORM ii
ACKNOWLEDGEMENTS iii
ABSTRACT iv
ÖZ vi
LIST OF FIGURES xi
LIST OF TABLES xiv

CHAPTER ONE – INTRODUCTION 1

1.1 Problem Statement 3
1.1.1 Current Solutions 3
1.1.1.1 Traditional Approach 3
1.1.1.2 Alternatives Disaster Management 5
1.1.2 Specific Problems 6
1.2 Research Questions 9
1.3 Contributions 9
1.4 Structure of the Thesis 10

CHAPTER TWO - BACKGROUND AND RELATED WORK 11

2.1 Disaster Management and Foundational Issues 11
2.1.1 Mitigation 16
2.1.2 Prepardness 16
2.1.3 Response 17
2.1.3.1 Prehazard 17
2.1.3.2 Disaster Effects Ongoing 17
2.1.3.3 Disaster Effects have ceased 17
2.1.4 Recovery 19

2.2 Semantic Web 20
2.3 Ontology 21
2.4 Geo-ontologies and Case Studies 22
2.4.1 Geo-ontologies 22
2.4.2 Case studies 24
2.5 Disaster Ontologies and Case Studies 41

CHAPTER THREE - ONTOLOGY DESIGN 46

3.1 Description of the Study Area 46
3.2 Modeling Framework 46
3.3 Disaster Ontology Development Methodologies 47
3.4 Disaster Ontology Development 50
3.4.1 Specification 50
3.4.1.1 Describing Domain 51
3.4.1.2 Scenario and Competency Questions 52
3.4.1.3 Identification of Ontology Goals and Scope 53
3.4.2 Conceptualization 57
3.4.2.1 Defining Classes and Hierarchy 58
3.4.2.2 Defining Property and Class Relations 59
3.4.2.3 Representing Rules and Restrictions 62
3.5 Creating Ontology on Protégé 68

CHAPTER FOUR - SPATIAL DATA INTEGRATION WITH DISASTER ONTOLOGY 77

4.1 Spatial Data Transformation 80
4.2 Non-spatial Data Integration 83
4.2.1 Reasoning with Using Jena Library Reasoning Mechanism 87
4.2.2 Reasoning with Parliament Triple Store Reasoning Mechanism 89

4.2.3 Reasoning with Using Reasoner Mechanism on Protégé 90
4.3 Spatial Data Integration 91

CHAPTER FIVE – EARTHQUAKE DAMAGE ASSESSMENT WITH DL RULES.....**94**

5.1 Triple Stores 94
5.1.1 Strabon 94
5.1.2 USeekM 95
5.1.3 OWLIM 96
5.1.4 Parliament Triple Store 96
5.2 Acceleration Calculation Methodology 97
5.3 GeoSPARQL Query Operations 98
5.3.1 Neighborhood Distances from the Main Epicenter Query 98
5.3.1.1 Basement Type Query 101
5.3.1.2 Ground Acceleration Calculation 103
5.4 Querying the Ontology and Results 105

CHAPTER SIX - CONCLUSION.....**116**

6.1 Answering the Research Questions 116
6.1.1 How Geo-ontologies can be Utilized in Earthquake Domain for Disaster Management on Urban Areas 116
6.1.2 How to Integrate Spatial data from Distributed Data Sources 117
6.1.3 How can Damage Assessment Estimate Early Post-disaster by Using DL Languages 118
6.2 Future Work 119

REFERENCES.....**120**

LIST OF FIGURES

Page

Figure 2.1 Natural catastrophes worldwide 2011 13
Figure 2.2 Disaster management cycle 15
Figure 2.3 Topological relations 22
Figure 2.4 A prototype client of the web services 26
Figure 2.5 Overview of system architecture 28
Figure 2.6 Framework for representing and maintaining geo-ontologies..... ..29
Figure 2.7 RCC and OWL hybrid knowledge representation system..... ..30
Figure 2.8 SWING environment framework 33
Figure 2.9 GRDF ontology diagram 34
Figure 2.10 Mineral resource management use case description..... ..35
Figure 2.11 Heteregeneous data translation process with XSLT..... ..35
Figure 2.12 OntoFire ontology structure..... ..38
Figure 2.13 Average point formula for distance calculation..... ..39
Figure 2.14 Comparison of geo-vocabularies with respect to GeoSPARQL requirements 40
Figure 2.15 The MIRIP (Multi-format Information Retrieval, Integration and Presentation) architecture 44
Figure 3.1 Semantic GIS based disaster management system framework..... ..47
Figure 3.2 Ontology development life cycle..... ..50
Figure 3.3 İzmir disaster management structure..... ..52
Figure 3.4 GeoSPARQL ontology components..... ..55
Figure 3.5 Basic concepts of geometry class 56
Figure 3.6 Spatial objects in WKT and GML types..... ..56
Figure 3.7 Equivalent simple features, RCC8 and Egenhofer relations 57
Figure 3.8 Disaster ontology class hierarchy 58
Figure 3.9 Types of object properties in OWL 60
Figure 3.10 Definition of spatial and non-spatial based properties 61
Figure 3.11 Disaster ontology class, property and ‘domain and ranges’ view 62
Figure 3.12 Disaster definition rule 65
Figure 3.13 Neighborhood damage ratio rule 65

Figure 3.14 Bridge situations in damaged area rule ..66
Figure 3.15 Road situations in damaged area rule ..67
Figure 3.16 Tent city requirements for damaged area rule ..67
Figure 3.17 GeoSPARQL ontology import ..69
Figure 3.18 Disjoint class of the ontology ..70
Figure 3.19 Domain and ranges in the ontology ..70
Figure 3.20 Restriction result in Protégé ..71
Figure 3.21 Severe neighborhood restrictions ..73
Figure 3.22 Bridge and broken bridge class view ..74
Figure 3.23 Road and closed road class view ..75
Figure 3.24 Tent city requirement rule for neighborhood class ..76
Figure 4.1 Data integration into ontology from XML data sources ..78
Figure 4.2 Spatial data integration methodology ..80
Figure 4.3 FME spatial data transformation method ..81
Figure 4.4 Geographical markup language structure ..82
Figure 4.5 Reading neighborhoods GML in java application ..83
Figure 4.6 Reading default ontology model in java application ..83
Figure 4.7 Inserting individuals into neighborhood class in ontology ..84
Figure 4.8 Reading values for data type properties from GML file ..85
Figure 4.9 Inserting data type properties as triples on ontology ..86
Figure 4.10 Ontology model with OntModel spec ..88
Figure 4.11 Basic ontology model SPARQL result ..88
Figure 4.12 Pellet reasoner ontology model ..89
Figure 4.13 Pellet reasoner SPARQL query result ..89
Figure 4.14 Inferenced SPARQL result ..90
Figure 4.15 Earthquake class individuals view ..90
Figure 4.16 Disaster class inference individuals view ..91
Figure 4.17 Inserting coordinates value with Parliament remote model as WKT ..92
Figure 4.18 Neighborhood as WKT SPARQL result view ..93
Figure 5.1 Strabon graphical user interface ..95
Figure 5.2 Parliament triple stores GUI ..97
Figure 5.3 Acceleration calculation algorithms ..98

Figure 5.4 Neighborhood distance from epicenter calculation 99
Figure 5.5 GeoSPARQL results of distance function between epicenter and neighborhood..... ..100
Figure 5.5 GeoSPARQL results of distance function between epicenter and neighborhood..... ..100
Figure 5.6 MapInfo results of distance function between epicenter and neighborhood..... ..100
Figure 5.7 Basement type selection of neighborhood with java application 101
Figure 5.8 GeoSPARQL results of intersect function between neighborhood and basement type..... ..102
Figure 5.9 MapInfo results of intersect function between neighborhood and basement type..... ..103
Figure 5.10 Acceleration process in java application 104
Figure 5.11 Neighborhood acceleration values are inserted ontology into triple store 105
Figure 5.12 Semantic GIS based disaster management web site graphical user interface..... ..106
Figure 5.13 HTTP request for triple store query java class 107
Figure 5.14 Adding spatial objects into Google map..... ..108
Figure 5.15 Selecting moderate damage neighborhoods 109
Figure 5.16 Moderate damage neighborhood query 109
Figure 5.17 Severe damage neighborhood query..... ..110
Figure 5.18 Broken bridges according to neighborhood damage 111
Figure 5.19 Closed roads and broken bridge according to neighborhood damage ..111
Figure 5.20 Available meeting places within severe damaged neighborhood..... ..112
Figure 5.21 Available hospitals within severe damaged neighborhood 113
Figure 5.22 Available security points within severe damaged neighborhood 114
Figure 5.23 Available logistic distribution centers within severe damaged neighborhood..... ..114

LIST OF TABLES

Page

Table 1.1 Institution data publishing methods in İzmir 7
Table 2.1 Most casualties' natural disaster at 2011 12
Table 2.2 Differences in disaster impact between rich and poor countries 14
Table 2.3 Ontology mapping from database structure 36
Table 3.1 Competency questions 52
Table 3.2 Disaster management scenario 53
Table 3.3 Glossary of terms 54
Table 3.4 Naming conventions of DL 63
Table 3.5 Horizontal acceleration equations 72
Table 3.6 Relations between magnitude, ground acceleration and building type 72
Table 4.1 Ontology model specification and reasoning levels 87

CHAPTER ONE
INTRODUCTION

Human beings have been exposed to and have suffered from disasters materially and psychologically for centuries and they are often vulnerable to disasters. In recent years, natural disasters have increased swiftly with negative effects of global climate change and population growth in urban areas. Besides that, prior to occurrence of disaster, preventive measures can be taken by using previous experiences to reduce loss of lives and to manage disaster efficiently. If disaster managed in an effective and efficient way, disaster losses can be reduced quickly. Disaster management consists of legal, executive and technical activities each complementing one another. These activities should be planned and executed for mitigating disaster and reducing its damages (Tecim, 2009). Disaster management systems have interdisciplinary structure which brings several institutions together.

In recent years, all institutions have managed and have stored institutional information using IT technologies by introducing internet and information technologies. With the advent of geospatial technologies, institutions have started to manage spatial information with the help of geographical information system (GIS). Therefore, today there is so much spatial and non-spatial information stored in various isolated institution devices. Disaster managers need to extract information from these distributed and heterogeneous data sources for managing disaster efficiently. Extracting information from data at hand should be ensured only by integrating heterogeneous data sources (van Borkulo, Barbosa, Dilo, Zlatanova, & Scholten, 2006). There have been several studies and application to interoperability problem for many years. Generally interoperability studies between institutions are practiced from the point of technical and syntactic approach.

According to the The European Telecommunication Standards Institute (ETSI), technical interoperability is usually associated with hardware /software components, systems and platforms that enable machine-to-machine communication to take place (ETSI, 2006). This kind of interoperability is often centered on communication

protocols and the infrastructure needed for those protocols to operate. Syntactical interoperability approach is usually associated with communication protocols which are need to have a well-defined syntax and encoding such as XML, HTML, etc. Therefore, technical and syntactic interoperability only guarantees the correct transmission of information, but does not tell anything about the meaning of this information, for this reason semantic interoperability be provided between institutions. But the integration of spatial and non-spatial data from distributed data sources into the system and making new inferences is needed to be done based on available relations between extraordinary and complex data in disaster situation. For this reason, disaster management system which is multilayered and has complex structure should be built in an more effective way by using semantic web technologies for gaining semantic interoperability abilities.

Semantic web technology was proposed as "a new form of web content that is meaningful to computers" in 2001 by Tim Berners Lee (Berners-Lee, 2001). Semantic web technologies can help build more effective disaster management systems having multilayered and complex structure. On the other hand, traditional GIS handle spatial data from the distributed sources in a limited way, because GIS software solutions provide output in their own adopted format. At present, sharing and integrating geographic information has reached the stage of semantic interoperation and integration of spatial information based on ontology theory and technology (Chaves et al. 2005; Jiang et al. 2006; Forte et al. 2008). Ontologies facilitate the content processing and its delivery to the user by means of advanced, meaning-sensitive integration, reasoning and querying methods (Novacek, 2010).

Thus (Zlatanova, 2006, Fan & Zlatanova, 2011) considered researches which are needed to be done on semantic data discovery for emergency response domain as building of ontology for spatial data in preparation for knowledge discovery. In addition to this, a significant progress in solving heterogeneity issues at the syntactic, schematic, and structural level has been achieved. However, solving semantic conflicts is still an open issue which has gained more and more attention during the last years, especially within the Semantic Web community (Langegger, 2010).

1.1 Problem Statement

Interoperability approach and problems are anatomized on disaster management in this section. Research questions have been defined which refer out of a thesis based on the analysis of the problems.

1.1.1 Current Solutions

Firstly, we mentioned about interoperability approach and process on traditional disaster management. After then, we refer to more specific problem and more efficient interoperability approach which have been integrated with semantic web technologies in recent years.

1.1.1.1 Traditional Approach

Disaster management systems are developed on the purpose of disaster mitigation and prevention loss of lives and property as mentioned above. In the management of a disaster, different people from various organizations are involved, and large amounts of heterogeneous data are required to be created, collected and integrated. Spatial data are used further with the development of GIS. In the light of these developments, spatial decision making techniques' usages are necessity for managing disaster.

One crucial component in spatial decision making is the availability of accurate, up-to-date and complete spatial data, which often have to be sourced from many government agencies. Traditional approach of the disaster decision maker is the creation of reports by performing some processes. First, acquire relevant information, usually in the form of data files from different providers, structure and prepare these files and analyze the contained information from big data sets in GIS and finally produce reports with maps for decision makers. But this process may cause to make slow, deficient decision at disaster which required quick response.

Hence, making geospatial data available in the disaster management poses various challenges to the data provider (Maue & Schade, 2010). Open Geospatial Consortium (OGC) who defines standards about spatial applications has published standard catalog related GIS to overcome these problems. This standard provides data homogeneity. OGC developed Geographic Markup language (GML) for OGC spatial data models. But it is unable to cope with discrepancies in the meaning, interpretation and intend to use of geographic concepts (Nikolaos, 2005). In the last decade spatial data have been made available as web services, following the principles of service oriented architectures (Erl, 2005).

OGC provides Web Map Service (WMS) and Web Feature Service (WFS) for publishing spatial data into standard format. WMS is a standard protocol for serving georeferenced map images over the internet that is generated by a spatial server. WFS provides an interface allowing requests for geographical features across the web. Therefore, Spatial Data Infrastructure (SDI) is starting to use OGC spatial data processing and web services standards. But WMS and WFS standard cannot provide semantic interoperability and machine understanding for spatial data.

For example; Organization problem is tackled about leading humanitarian aid to the victims, Population data are needed from registration office and available road information from municipality from different spatial data sources in different formats to deal with this problem. People know that ‘population’ means total person number in district and located population table at registration office and ‘road’ means route for people and vehicles and located road table at the municipality. But machines don’t know that total person number means population or route means for people road.

The main reason of the problem is detailed in the example below in the way of how the resources are processed by traditional approaches as technical and syntactical for interoperability. Although several studies can be made related to the provided interoperability until this time, results are not proper for distributed and heterogeneous application.

1.1.1.2 Alternatives Disaster Management

Alternative approach starts with disaster management process incorporating with semantic interoperability approach. Alternative approach accommodates to provide interoperability solution for the distributed institutions with semantic web technologies. This can be achieved either by providing description of the concepts and relationships in several domains or by resolving data into machine-readable format for enabling data integration or reasoning mechanism. Approaches seeking for solutions along these lines are presented in Chapter 2 as a part of the related work overview. Semantic Web Initiative finds out machine-readable data notion for the first time (Daconta, Obrst & Smith, 2003). Thus agents can find all heterogeneous data about the related domain and bring more meaningful results into decision makers with by using relations between concepts. For this purpose semantic web initiative publishes some standards for gaining meaning and publishing between institutions. Resource Description Framework (RDF) and Resource Description Framework Schema (RDFS) are the most standard language by Semantic Web Community. RDF was constituted on the purpose of bringing a standard for data annotation and describing metadata of web-based sources in 1999 by W3C (Komesli, 2007). Although RDF and RDFS concepts defined the relations between concepts, these languages were soon found to be too limited in expressive power. Web Ontology Language (OWL) and OWL 2 languages are more expressive language based on description logic. OWL is extensively used in the life of sciences community, where it has rapidly become a de facto standard for ontology development and data interchange. OWL language was revealed officially by Web Ontology Working Group in 2004 and has become a formal proposal of W3C.

Spatial data annotation studies were started in using spatial data in ontologies due to the produced massive amounts of spatial data and these data were started to be used in interdisciplinary studies such as disaster management. But OWL, an ontology representation language can’t describe determining and reasoning rules of spatial relationships owing on account of its limitation (Huang & Deng, 2009). For this reason there are several studies on adopting geo-ontology to represent semantic and

spatial relationship and applying Semantic Web Rule Language (SWRL) to express spatial rules, therefore, spatial reasoning is used by (Grütter & Messmer, 2007). Geographical ontologies are used in several domains, but yet researchers started to use ontologies with disaster management. We believe that there are still challenges in the field of semantic web-based disaster management such as spatial data annotation as proprietary data formats from static data sources into geo-ontology. Although several OWL-based ontologies are used in disaster domain, there exists no earthquake ontology specifically for the purpose of earthquake modeling. At the same time after modeling earthquake by using earthquake ontology, disaster assessment is required swiftly for efficient disaster management. Based on this requirement, the other unstudied subject is DL-based damage assessment. These challenges are explained in the following section.

1.1.2 Specific Problems

Earthquake is a prevailing natural event threatening more than 55% of cities with at least 2 million people who are located within 200 km of a plate border zone (Bilham, 2009). Risky urban areas cannot be moved to safer locations and their population is increasing continuously together with spatial expansion. As a result, complex urban environment should be better prepared for an earthquake disaster.

Earthquake should be modeled in detail for reducing damages of earthquake. Efficient and machine understandable disaster management can be made by defining earthquake concepts and identification of relations and rules between concepts. Decisions are taken by machines reduced intervention duration of the disaster and also more accurate decision can be taken. Ontology is used for earthquake modeling and also for designing machine understandable system.

Several studies concentrated on creating disaster ontologies with OWL language and applied these to various disaster management scenarios (Tanasescu et al. 2006, Kructhen et al. 2007, Truptil et al. 2008, Liu et al. 2009, & Babitski et al. 2011).

There is no ontology which has been founded for the disaster management in earthquake domain which motivated us to create earthquake ontology.

In a literature review, geographical ontology based spatial semantic data annotation is usually used for the annotation of spatial web services which was explained in Chapter 2 in detail. Institutions in developed countries can publish spatial data via web service by using GIS and web technologies together. Besides, developing countries hold spatial data in isolated and heterogeneous structure owing to institutions' policies and still cannot reach the desired level on interoperability approach. Providing interoperability between institutions in developing countries can be realized as different approaches than in developed countries, for this reason providing interoperability in developing countries is very challenging.

While investigating organizational structure of İzmir is as a case study of our work which are explained in Chapter 2 in detail, there are eight different institutions which provide information for disaster management. For example; Gediz electricity institution has yet digitized spatial data from hard-copy maps and cannot publish spatial data by using SDI over internet. Besides that, police and health institution save spatial data in TAB format for using in MapInfo and İzsu waterworks in shapefile by using ArcMap. Only municipality publishes information via web services (Table 1.1). Developing a new method is required for spatial semantic data annotation of spatial data for providing interoperability of institutions

Table 1.1 Institution data publishing methods in İzmir

Institution Name	**WFS / WMS**
Waterworks	✗
İzmir Municipality	✓
Health Authority	✗
Governorship	✗
Telecommunication Authority	✗
Gediz Electricity	✗

In recent years, earthquake damage estimation studies have shown great progress with the development of earthquake engineering. These studies were carried out at different levels and scenarios. For example, rapid response system takes part immediately after the earthquake and provides assessment of the distribution of ground motion intensity. Earthquake rapid response system takes to provide ground motion measurement with strong ground motion network and data processing system for assessment (Erdik, Şeşetyan, Demircioğlu, Hancılar & Zülfikar, 2010). Although strong ground motion network already exists in several cities of the world such as Tokyo, Yokohama, İstanbul, strong ground motion network is not available for other high risk cities due to its high cost and technology. Damage estimation is carried out with information retrieval from different data sources in cities which have no available strong ground motion network. At the same time, strong ground motion network only estimates the geological formation of the damages, but only geological formation is not sufficient for complex damage estimation. Integrated damage estimation by using earthquake source parameter, building inventory, population, land use maps information enable to take more effective decisions. However, deriving damage which is solely based on the source parameter of earthquake is very challenging task. Lang, (2008) performed damage estimation by using the decision-tree method on earthquake source parameters. They were carried out near-real time application by using damage estimation tool named SELENA.

There are several challenges about the damage estimation process which were elaborated below. Disaster conditions need fast and accurate decisions, in fact, it is impossible for humans to examine all the distributed information and to understand the content and then discover and combine into the disaster management. If machines can be used to discover inference and to manage data automatically, disaster can be managed more efficiently than ever. Ontologies are used for ensuring semantic interoperability and DL rules are used for damage estimation. Thus, only a human can be inferenced from the distributed and heterogeneous data sources which can also be done by computer fast and accurately with DL rules and ontologies. In a literature review, there exist several ontology-based disaster management studies

(Klien et al., 2006, Xu et al., 2008 & Zhang et al., 2014) However, there has been found no DL-based disaster ontology which can estimate earthquake damage.

1.2 Research Questions

Several approaches are reviewed about semantic web-based disaster management. But there are still problems which which are needed to be solved. We are looking for a solution for a couple of problems within the scope of thesis. The study is focused on three points which are coupled with each other.

- How can geo-ontologies be utilized in earthquake domain for disaster management in urban areas?
- How spatial data from proprietary format from the distributed data sources to be integrated?
- How can damage assessment be estimated for early post-disaster by using DL languages?

1.3 Contributions

These thesis contributions defined the aims of semantic geographical information based disaster management as hereinbelow. In close relation to the research questions, various theories of semantic annotation of spatial data from the distributed data sources are analyzed. As a result of reviews, usually web services are used as data source. But annotation of data sources which are isolated and heterogeneous approach has not been discussed yet enough. For this purpose, we propose a new model for annotating these isolated and heterogeneous data sources.

Another contribution of the study is to develop earthquake ontology which aims a new approach in the disaster management field due to the fact that there is no earthquake ontology for disaster management in domain. Finally, early post-disaster damage assessment can be done by this study with DL and ontology technologies. Earthquake ontology and DL rules come together and make inferences about the

damage assessment. This approach is a new method for damage assessment by using semantic web technologies.

1.4 Structure of the Thesis

Related work and background in the field of disaster management and semantic web are discussed in Chapter 2. This includes issues related to disaster management such as interoperability problem between institutions, damage assessment without using strong ground motion network. System description is presented in Chapter 3 which presents the framework model and a disaster ontology model. Chapter 4 presents the implementation of this disaster ontology and earthquake damage assessment by using DL rules and finally querying the disaster ontology and publishing results for disaster managers. Conclusions and discussion and future work issues were mentioned respectively in Chapter 5.

CHAPTER TWO

BACKGROUND AND RELATED WORK

This chapter further elaborates on the motivations of this work. First of all, foundational issues related to interoperability problems are addressed on disaster management that is important for the thesis from theoretical point of view. In this context traditional disaster management studies are explained in section 2.1. After then, approaches and technologies will be examined related to the problems above. The main motivation for this work is identified in research questions; imperfection in semantic annotation of spatial data from the distributed data sources and lack of disaster ontology in earthquake domain for earthquake risky urban areas and DL-based damage assessment. Several major research areas are relevant to the problem addressed by the thesis. This chapter provides a high-level overview of the relevant case studies.

After an overview of the work related to the core of the thesis, Section 3 discusses ontology design and spatial data annotation. Implementation of specific application is discussed in section 4. Finally, thesis contribution to literature is argued in the last section.

2.1 Disaster Management and Foundational Issues

Disasters are events that cause substantial destruction and loss of lives due to natural or other reasons. The Federal Emergency Management Agency (FEMA) classifies disasters as natural (earthquake, flood, hurricane and fire, etc.) or technological (terrorism, nuclear power plant emergencies, hazardous materials, etc.) disaster (Hristidis, Chen, Li & Deng, 2010). Common disaster type is a natural disaster in the world. Developed and developing countries are always affected by the natural disaster. Scientist's assested world is stepping into a new era in which the formation and the destruction of the natural disaster will increase swiftly because of the deterioration of environmental conditions (Kremier & Arnold, 2000).

Especially natural disaster and earthquakes have been gradually increasing in the last 30 years. Natural disasters are examined with two titles. These are geological and meteorological based disasters. Meteorogical based disaster occurs in consequence of atmospheric events. These happen while atmospheric events such as (temperature, rain, pressure and wind) exceed a certain limit. The most common meteorological disastes are climate change, forest fire, drought, mist, hail, streak of lightning, blizzard, storm, frost, snowslide and flood. Geological based disasters are directly come from crust of the earth or depth of surface. The most common geological disasters are earthquake, landslide, rock fall and mudflow. Millions of people are affected by the natural disasters every year. The annual cost of the natural disasters is estimated at 50 billion $ for the world economy. According to the statistics of the United Nations in 1986, natural disasters affected 1.4 billion people between 1900 and 1986 in the world and caused 42 million losses of dead. Natural disasters caused losses of 24.699 people in 7 countries only in the year of 2011 (Table 2.1). These statistics are analyzed in detail, 820 loss events are happened and 9% of these events are geological based. %62 of death caused by natural disaster is again based on geological events. These numbers show us that geological events caused a huge amount of human losses (Coppola, 2011).

Table 2.1 The most casualties caused by the natural disaster in 2011.

Disaster Type	**Country**	**Number of Loses**
Earthquake/Tsunami, March	Japan	19 846
Tropical storm 'Sendong', December	Philippines	1 430
Flood, January	Brazil	900
Flood, August-December	Thailand	813
Earthquake, October	**Turkey**	**604**
Flood, August-November	Pakistan	509
Storm, April	United States	350
Flood, August-November	Cambodia	247
	Total	**24 699**

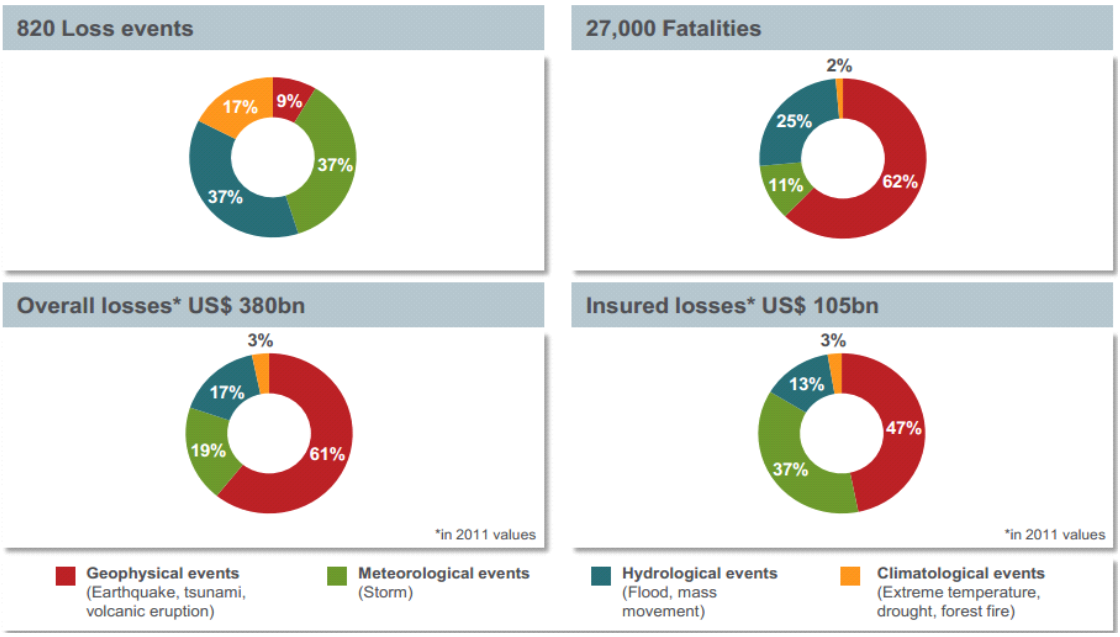


Figure 2.1 Natural catastrophes worldwide 2011, Resource: Münchener Rückversicherungs-Gesellschaft, Geo Risks Research, NatCatSERVICE – As at January 2012.

The effects of the natural disasters expose different results in the ratio of development levels of countries (Figure 2.1). Developing countries – especially small developing countries- are significantly more exposed than developed countries to the natural disaster risks. A natural disaster is more frequent and of higher intensity in developing countries and their economic cost is several times larger than in developed countries. These high economic costs are consequence of the weaker building construction that is found in poorer countries (Perry, 2009). Although the economic losses from natural disaster occur to be covered by insurance rate is %40 in developed countries, this ratio is %3 in developing countries (Gurenko & Zelenko, 2009). Based on these facts, inferences can be drawn about a nation's disaster risk by considering its development status. Public health expert EricNoji (1999) identified four primary reasons why the poor in general are often most at risk:

1. They are least able to afford housing that can withstand seismic activity.
2. They often live along coasts where hurricanes, storms urges, or earthquake-generated tsunamis strike or live in flood plains subject to inundation.

3. They are forced by economic circumstances to live in substandard housing built on unstable slopes that are susceptible to landslides or are built next to hazardous industrial sites.
4. They are not educated as to the appropriate life-saving behaviors or actions that they can take when a disaster occurs. (Coppola, 2011).

Table 2.2 Differences in disaster impact between rich and poor countries (Perry, 2009).

Rich Countries	***Poor Countries***
Tend to suffer higher economic losses, but have mechanisms in place to absorb these costs *Employ mechanisms that reduce loss of life such as early warning systems, enforced building codes and zoning* *Have immediate emergency and medical cares that increase survivability and contain the spread of disease transfer much of personal, private and public risk to insurance and reinsurance providers.*	*Have less at risk in terms of financial value, but maintain little or no buffer to absorb even low financial impacts* *Economic reverberations can be significant, and social development ultimately suffers* *Lack of the resources is necessary to take advantage of advanced technologies, and have little ability to enforce building codes and zoning even if these mechanisms do exist* *Sustain massive primary and secondary casualties* *Generally do not participate in insurance mechanisms. Divert funds from development programs to emergency relief and recovery.*

More people live in cities rather than in rural areas. Many of these metropolitan areas are threatened by floods, storms, earthquakes and other natural hazards. The world's sprawling cities are centers of economic activity and growth. But when natural disaster hits a densely populated area, the effects can be catastrophic. Cities which are affected by the natural disasters are not only suffering physically but also are considerably affected socially and economically (Table 2.2). Region problem discussed in the form of a new equilibrated different composition and at a higher level within the same integrity which are social, economic and physical integrity and general structure, deteriorated significantly. Essential administrative works need to be done such as providing adequate housing for everyone, providing resources and planning mechanisms in order to make human settlements safer, healthier and livable against increasing vulnerability towards disaster. Communities are required to make

a point of mitigation studies against disasters due to direct losses from natural disasters.

Besides that, prior to occurrence of disaster, preventive measures can be taken by using previous experiences to reduce loss of lives and to manage disaster efficiently. Disasters may be natural on the origin, but it is the way in which societies have developed that causes them to become disasters (Wisner & Walker, 2005). Disaster management (DM) is the management of the risks and consequences of a disaster (Othman, 2013). Further, disaster management can be divided into the following four phases: (a) Preparedness; (b) Mitigation; (c) Response; and (d) Recovery (Figure 2.2) disaster cycle). Disaster management has multilateral and interdisciplinary structure. There are several private, national and international institutions existing in multilayered structure of disaster management.

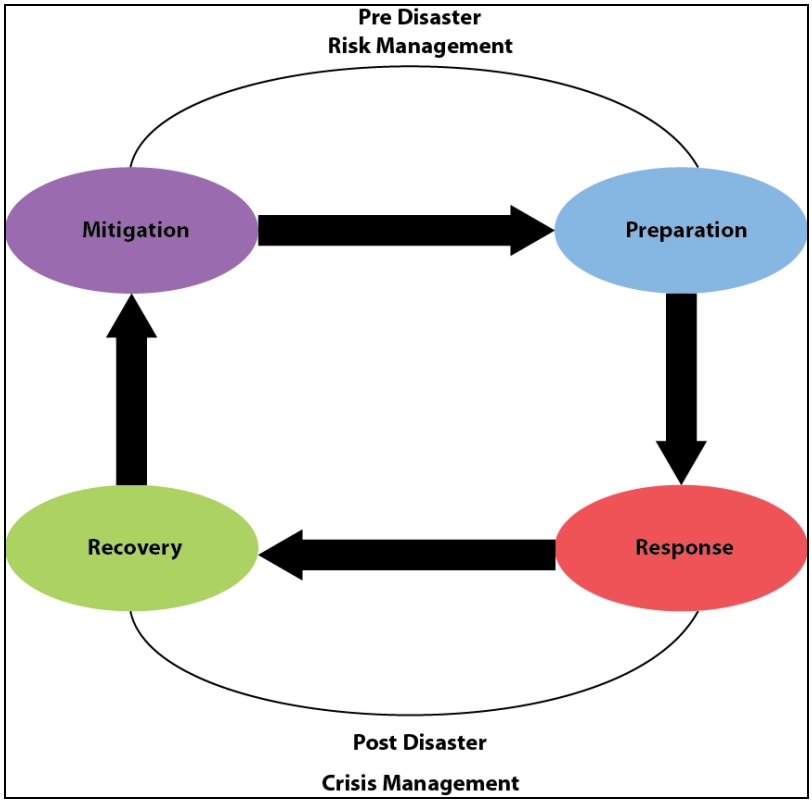


Figure 2.2 Disaster management cycle

2.1.1 Mitigation

Mitigation includes precautions and activity to consider for preventing disaster trouble. These activities are long-time studies which require wide variety of disciplines in accordance with a specific target in many institutions and organizations. Mitigation phase starts with improvement phase activities and continues until a new disaster happens. Activities in this phase show wide application field such as country, region and settlement scale.

2.1.2 Preparedness

The goals of disaster preparedness are to know what to do in the aftermath of a disaster, to know how to do it, and be equipped with the right tools to do it effectively. Also this phase comprises determining principles and configuring application related to the definition of recruitment and performing distribution services for requirements of health, sheltering, daily consumption after the disaster. In this phase making enforcements periodically, assigning personnel significant task, making contingency plans at city and district level are required. These activities reduce losses of lives and destructive effects of the disaster. Contingency plans are required to make at governmental level on preparedness stage. Each government will be required to perform a range of tasks and functions in the lead up to its aftermath. Because post-disaster is not available proper time for planning. Governments are required to work with several institutions for the identified studies in contingency plans together. Government and other institutions already know how to act in details after the disaster. National plans, which may also list various government agencies that have been assigned responsibilities in line with their regular missions, and may describe the tasks and functions these agencies, are expected to perform in the event of a disaster. Governments wait to share information from institution for efficient decision making after the disaster. Which type of data will be shared as information is explained in detail on contingency plans. Damage assessment estimation is very crucial on the preparedness stage. For example, rapid response system takes part immediately after the earthquake and provides assessment of the distribution of

ground shaker intensity. Although strong ground motion network already exists in several cities of the world such as Tokyo, Yokohama, İstanbul strong ground motion network is not available for other high risk cities due to its high cost.

2.1.3 Response

Response function of disaster management aimed to reduce injuries, loss of life and the damage to property after the disaster. Response is the most complex phase amongst other four phases, as it is conducted during periods of very high stress, in a highly time-constrained environment, and with limited information. Response involves the rapid resumption of critical infrastructure (such as opening transportation routes, restoring communications and electricity, and ensuring food and clean water distribution) to allow recovery to take place and to reduce loss of life. Response phase includes 3 stages in its own right.

2.1.3.1 Prehazard

Disaster is to be expected or is inevitable in this phase.

2.1.3.2 Disaster Effects Ongoing

This phase is between the first effect of the disaster and the end of the disaster impact. Streak of lightning and earthquake can be recognized within seconds. Besides that, some disasters such as flood, storm and fire are continued day after day

2.1.3.3 Disaster Effects have Ceased

This phase is the end of the response stage. The impact of the disaster is reduced and losses are revealed properly in this phase. First, responders are establishing dead and injured people and securing disaster area. Once the disaster response begins, the first priority is to save lives. Activities such as assessing the disaster, treating remaining hazard effects, providing water and food, shelter, fatality management,

sanitation, security can be made on response stage. Response comprising coordination between institutions does not only provide shelter, search and rescue, first aid needs at disaster. Several institutions are working together for particular purpose in disaster domain. For this reason, response stage is the most important stage for providing interoperability between institutions. Several problems are determined about the coordination which is efficient allocation of resources, detection of closed roads due to the wreckage, decide to use which first meeting place, site selection for tent cities, rescue team to the hospital when past earthquakes such as Sakarya 1999 and Van 2010 are analyzed that are required very fast response on response stage. Due to these problems, lack of efficient management causes several losses of lives, not the disaster itself. Assessing the disaster, treating remaining disaster effects, providing water and food, shelter fatality management, sanitation, security should be made for responsing disaster during the response stage. Disaster assessment should be made as much as quickly for efficient management of disaster. First the extraction of the data from institutions and analysis of this data with available data for answering questions such as what, where and how about disaster should be required for the assessment of the disaster. Data extracting and processing phase are called as disaster assessment. Data assessments are grouped under two headings which are case assessment and needs assessment.

- Location of area which were affected by the disaster
- Number of dead and injured
- Types of injured and illness
- Description of the medical, health, nutritional, water, and sanitation situation
- Infrastructures and critical facilities situation
- Residence and commercial structures situation are determined while case assessment in other words, loss assessment;

Identifying needs of disaster victims and where the sources come for providing to the victims are determined with need assessment which is another disaster assessment tool. Management of disaster is aimed with analysis of the collected data using accurate reporting methods above. To increase its effectiveness, standardized

formats, terminology, and collection systems have emerged throughout the disaster management world that, while not uniform across all agencies, offers greater ease of use by many recipients (Coppola, 2011). Planning, data collection, data analyzing, prediction, monitoring phases are performed for the assessment of the disaster. Distributed data sources can be used for these phases. But this phase can not be carried out efficiently because of lack of the provided interoperability between institutions.

2.1.4 Recovery

Reconstruction of facilities is fulfilled to eliminate the effects of the disaster in this phase. Reconstrunction facilities which methods and then when and how should be determined advance same as the other stages. In relation to the other disaster management functions, it is by far the most costly. Recovery is also the least studied and the least organized of all of the disaster management functions. Several activities occur such as provision of temporary housing or long term shelter, assessment of damages and needs, demolition of the damaged structures, clearance, removal, and disposal of debris, inspection of the damaged structures, new construction. This stage needs to work with the distributed institutions together same as on the response stage above. For example, site selection is very critical process in the provision of temporary housing activity. Site selection should be made with the combination of different data sources such as population, basement type, zoning status come together. Planning and coordination have a place in this phase.

Disaster management phases have complicated and multi-disciplinary structure above. Each step in itself can be made activities in sequence toward disaster management. For example, kind of difficulties will arise on response and recovery stages if mitigation phase is not planned well enough. So disaster management is a process that should be considered in a holistic form. Several problems can be defined and resolved in spatially in this phase. For example, disaster manager encountered spatial based problem such as determining of the shortest route after disaster for the purpose of disaster assessment. Spatial data management comes into prominence at

disaster management. GIS is used for management of these spatial data. For this reason more efficient and effective decisions are provided by using decision making techniques and GIS together during the disaster. GIS functions those data collections (survey data, aerial photography, satellite data, etc.), querying, analysing analyzing and publishing are used at disaster management. Disaster managers need to accurate information quickly for making a correct decision during the disaster. For example, if disaster manager reaches spatial data such as infrastructure, boundaries, population density, meeting points, roads, will facilitate effective decision making. Interoperability problems are encountered at all phase in disaster management processes.

Tim Berners Lee proposed semantic web technologies in 2001 for solving interoperability problem. In this concept, data from different data sources are provided to convert machine understandable to use for a common purpose. Besides, with this technology it is aimed to make new inferences by establishing relationships which have no connection but are related to each other in different data sources that relations between data are predefined.

2.2 Semantic Web

In its early years, the Web could be viewed as a set of Web sites which offered a collection of Web documents. People read whatever was out there with the goal of getting information they could use in a variety of ways (Yu, 2011). Recently, the web has become much more interactive with the uprising of social media. Users create their own content and interact with the other users by developing social media. Data on the web are started to increase and diversify day by day through also users share content freely. For example, companies are publishing the structured data via web service standards from their databases. Eventually, information on the web needs to be machine understandable for using efficient purpose. Therefore, computers can understand the Web documents and can automatically accomplish tasks that have been otherwise conducted manually, on a large scale (Yu, 2011).

In this context Semantic Web is proposed in Scientific America article "The Semantic Web" by World Wide Web Consortium (W3C) under the leadership of Tim Berners Lee for providing semantification of information on the web. The Semantic Web is an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation (Lee, Hendler & Lassile, 2001). More metadata (data about data) is required in order to make machine- uınderstandable information on the web. Computers should know not only the meaning of data, at the same time should know various but also whether data can be the same at different sources on the web (Komesli, 2007). Resource Description Language (RDF) technologies based on XML are used for this reason. RDF provides statements about object with URI and preparatations of dictionaries associated by URI. Statements can be made by defining relations about data with RDF but more expressivity about data and reasoning can be required. Ontologies play a critical role for the semantic web, and it is necessary to understand ontology in order to fully appreciate the idea of the Semantic Web (Yu, 2011).

2.3 Ontology

The most popular definition of ontology is "ontology is aformalization of a conceptualization". Actually ontology is the very first science which is a particular theory about the nature of being or the kinds of existents (Daconta, 2003). In computer science there are different definitions of ontology. Ontology formally defines a common set of terms that are used to describe and represent a domain. A domain is simply a specific subject area such as disaster, geography, urban area. Ontology includes kinds of concepts such as classes, instances, properties, constraints and rules. Concepts are explained in details for describing the domain with ontologies. Thus, vocabulary appears about the domain and it provides a common and shared understanding about certain key concepts in the domain. In addition to this, ontologies provide a way to reuse domain knowledge and to make the domain assumptions explicit. Ontology makes its meaning available for the use of information technology. In this work, ontologies are used for providing

interoperability between different data sources on the purpose of conceptualization and reasoning.

2.4 Geo-ontologies and Case Studies

2.4.1 Geo-ontologies

According to Guarino; every information system has its own ontology, since is ascribed meaning to the symbols used according to practical view of the world (Guarino, 1998). The subject of ontology also is an important field of geographical information science. Geo–ontology gives a detailed research on connotations and hierarchy relationships of concepts of geographic entities in geographic information domain (Galton, 2003). Representing geographic entities–either constructed features or natural variation on the surface of the earth–is a complex task. A geo-ontology is different from other ontologies, because topology and part-whole relations play a major role in the geographic domain (Figure 2.3). Geographic objects can be connected or contiguous, scattered or separated, closed or open (Fonseca & Llano, 2006). Geo-ontology defines spatial relationships that indicate attribute value of geographic concepts. Those relationships are very important for geo-ontology, which are applied to spatial analysis, query and reasoning.

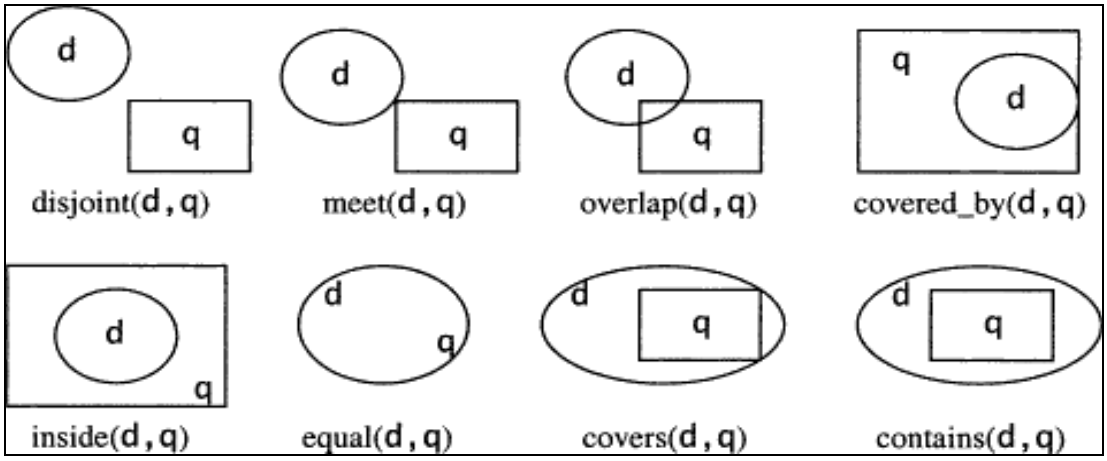


Figure 2.3 Topological relations

Several methods have tried to make geographic ontologies until now. Geo-ontologies are comprised along OWL language like other domain ontologies, in addition to OWL language, GML is used for comprising geo-ontologies, too. Abdelmoty (2005) and friends created geo-ontologies both using OWL and GML languages and differences are come in sight in this work.

First basic requirements of geo-ontologies are determined;

- Representation of geographically features and their associate.
- Representation of spatial and non-spatial properties of geographical features.
- Representation of conventional as well as spatial relationships between geographic features.
- Representation of specialization and generalization feature hierarchies.

Three different ontologies which are feature ontology, feature type ontology and spatial relation ontology are used while designing geo-ontologies with GML. In case, OWL language is used while creating single ontology. Spatial relations such as *PartOf*, *Contains*, *Overlap*, *AdjacentTo*, *Intersects*, *Near* are determined by both language. After suitability of GML and OWL languages for designing geo-ontologies are discussed.

- GML is capable of representing geographic features using a rich built-in vocabulary. OWL is a general purpose language, no specific structures are pre-defined to represent geographic features and they are modelled using user-defined classes.
- GML's predefined vocabulary provides better meaning to datatype properties than OWL's XML schema datatypes.
- In GML properties are used as a means of representing relationships between geographic features. XLinks are normally used to link objects by representing properties and relationships. OWL uses a similar concept to define relationships but also provides better semantics to properties.
- Restrictions, related to the cardinality of defined properties may be defined in both GML and OWL

- Objects as point, polygon predefined in GML but these objects should be defined again in OWL. For example, polygon should have at least 4 points. Conditions should be determined for making inference such as one object is inside the other object is in OWL language (Abdelmoty, 2005).

While modeling geo-ontologies, the problem of which concepts will be used, should be determined in advance. According to Henriksson, Kauppinen & Hyvönen (2008), geo-ontologies should be determined whether by spatial, area studies, man-land or earth science tradition. If study area is from earth sciences, concepts are possibly *coastalLandForm*, *DrainageBasin*. If concepts are spatial tradition, designed having regard to *GeometricObject*, *Coordinates*, *CRS* concepts. Geo-ontologies are used for spatial decision making with spatial data sources like disaster management domain. For this reason, the most suitable geo-ontology as upper-level ontology for the purpose of disaster management is examined with geo-ontology theory and application in literature in section 2.4.2. Disaster ontology theory and applications are examined for modeling disaster domain and integrated with geo-ontology at 2.4.3 section

2.4.2 Case Studies

Spatial data should be machine-readable with the increasing usage of spatial data on the web above. In this way, agent-based applications working with spatial data provide more meaningful results by using this content. For this reason, geo-ontologies have been designed using RDF and OWL standards until now. Several studies are started with 'Integrated geographical information systems' by Abel paper (Abel, 1998). At the same time Bishr made the first studies on interoperability problems on semantification of spatial data (Bishr, 1997, 1998). Semantic web technologies and ontology were used in GIS in 1999 by Egenhofer and Fonseca named ontology-driven GIS (ODGIS) (Egenhofer & Fonseca, 1999). In this study ontologies are used in planning process for highway construction which is intended to pass through two different cities. Interoperability problem is solved by using different data sources together by using ontologies. Urban and geometry ontologies

are designed, road and parcel objects defined in urban ontology and spatial objects are defined with geometry ontology. Semantic search and objects browsing functions are accomplished in this study. This study is more experimental about using ontology in GIS. Mennis (2002) develops a new kind of semantic GIS data model that is better able to represent users' conceptual models of geographic domains than the conventional vector and raster data models.

Much of these works in ontology-driven GIS has focused on developing ontologies of geographic domains for the purposes of data sharing and standardization (Fonseca, 2000), and has not been extended within the context of advanced GIS functions such as knowledge discovery. Basics of conceptualization are laying the foundations with this study. Semantic GIS data modeling has preliminary characteristics study. Users want to find information related to location on the web. The most common way to make this search, spatial expression as *near*, *inside*, *far* and *place name* information using together. Geo-ontologies are used for defining location name and spatial relation. In this context Smart (2003) made a study on spatial search engine with geo-ontologies. Query disambiguations, query term expansion, relevance ranking and web resource annotation are provided by using geo-ontologies. Web resources are annotated with annotation notion which increases visibility and usage of spatial resources.

The search engine needs to match the geographical context of the query with that of the available resources and the most relevant resources. DAML-OIL languages are used for designing geo-ontology. Spatial data usage aimed on the web pages by using more efficient search engine by annotating resources on the web. For example, distance from airport and hotel, route and road name information are annotated and provided to use in semantic search engines. The semantic web supports a future of intelligent agents' infering knowledge and thereby supporting spatial decisions. Hiramatsu & Reitsma (2004) aimed semantification of spatial information on the web, too. They designed geo-ontology which is designed with OWL and geographical features defined as class and relation. Relations on geo-ontology are designed as topological and directional relation. Topological relations are based on

OpenGIS Simple Feature Specification based dimensionally extended 9-intersection model. This ontology includes the following relations: *equals*, *disjoint*, *intersects*, *touches*, *crosses*, *within*, *contains* and *overlaps*. Directional relations are defined with the 8-sector model to express that cardinal directions *North*, *NorthEast*, *East*, *SouthEast*, *South*, *SouthWest*, *West*, and *NorthWest*. Instance repository and relation calculator web services are developed within the study. Instance repository was developed for storing spatial instance which has URI, name and coordinate information. Relation calculator was developed for measuring spatial relations between spatial instance and non-spatial instances (Figure 2.4). This studies' missing part is spatial reasoning with analyzing all of the study.

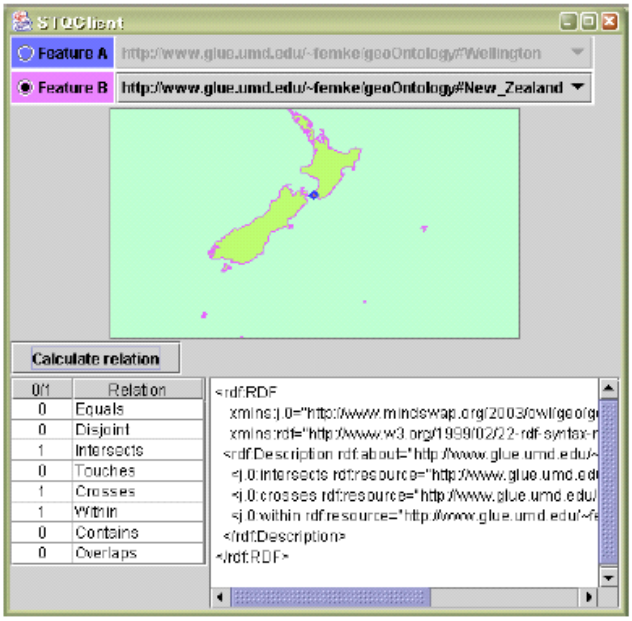


Figure 2.4 A prototype client of the web services (Hiramatsu & Reitsma, 2004).

The other ontology based study is “The semantic Web for Earth and Environmental Terminology (SWEET)” working on representation of ontology based environmental terminology and spatial objects on earth. This study is a conceptualization based study designed with OWL language. Keywords are used from NASA Global Change Master Directory (GCMD) during ontology design. In addition to this CF Standard Names list keyword was added into ontology. SWEET was designed as upper-level ontology for other environmental ontologies. They adopted the Postgresql object-oriented DBMS to store the names and parent-child

relations of their ontology elements (Raskin, 2005). This study was published as an application named on search of Earth science data resources. The challenge of this study is adding semantic acquisition, automatic ontology population and automatic query classification tools into framework.

Klien, Lutz & Kuhn (2005) solved semantic heterogeneity problem due to synonyms and homonyms keyword during keyword based search. This study presents a practical case study to what extent ontology-based service discovery can solve these semantic heterogeneity problems. Semantic translator for enhanced retrieval as a service broker is used for this challenge. Besides, scenario was developed the foundation of geo-information for assessment of potential storm damage on forest. The purpose of this study is the lack of interoperability between institutions. All institutions publish their own catalog and these catalogs are written with their own vocabulary. This shows up semantic heterogeneity. This study works on finding keyword based available spatial web service catalog. There is no spatial query or reasoning mechanism. The other study on handled by Nikolaos, Kostas, Michail & Nikolaos (2006) solving semantic heteregenity between GIS web services above. Spatial data were explained better with semantic Geoportals incorporating the spatial data in semantic and benefiting from them during the search process. Heterogeneity couldn't solve although it uses the metadata with CSD6M and ISO19115 standards for data standardization on geoportals. More expressivity metadata are created with RDF and RDFS languages which are used for for solving heterogeneity on spatial data and are queried with RQL language. For example, metadata of data convert RDF into three different data sources and associate with common ontologies and then querying. There ise no spatial query or reasoning mechanism on this study. Usually studies have been on semantic search, browsing, integration, analysis and discovering topics until this time.

Arpinar (2006) made a study on spatial, thematic and temporal reasoning. This study purposes on specification of spatiotemporal thematic ontologies and populating such ontologies with high quality knowledge. SWETO and spatial based SWETO-GS ontologies are created in this concept. These ontologies make geospatial analytics

and are able to automatically and semi-automatically extract metadata from semantic heterogeneous data sources (Figure 2.5).

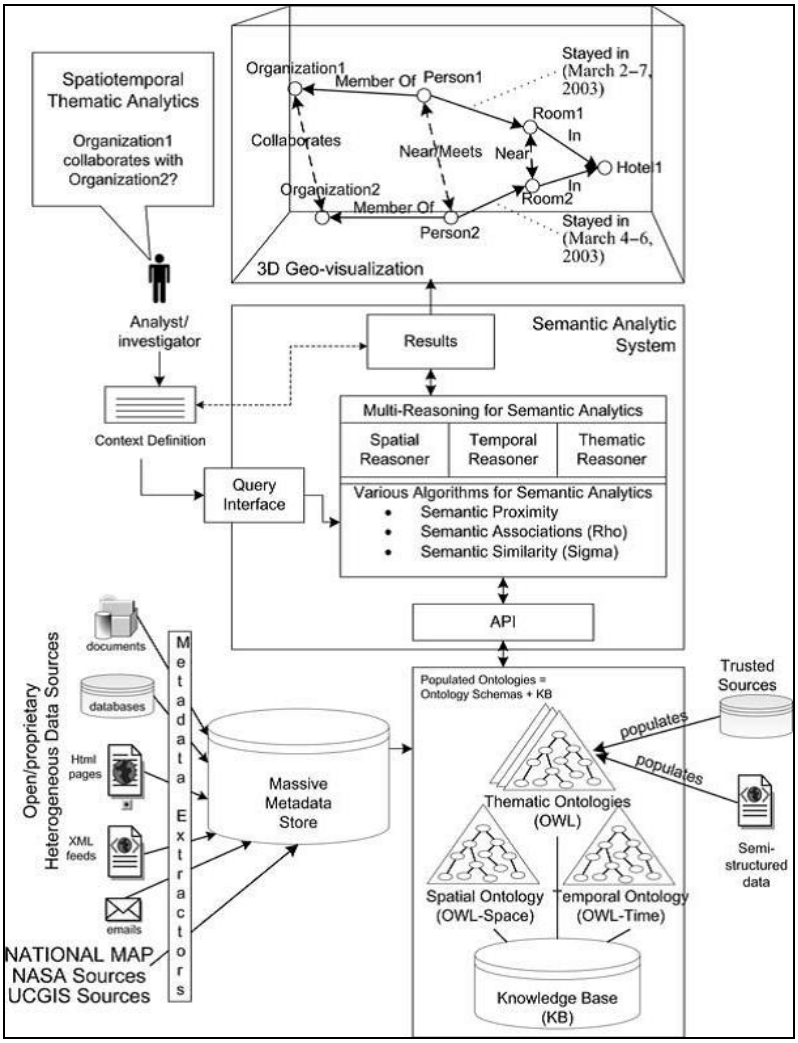


Figure 2.5 Overview of system architecture (Arpınar, 2006).

Besides that analytical processing that exploits these ontologies and associated knowledge base with integral support format they term "spatiotemporal thematic proximity" reasoning. Traditional spatial analysis is used metric measurement, but spatial relations in natural language; for example, "building in the east of road" cannot be queried. Geo-ontologies can make spatial analysis more expressive by using topological relations (*intersects*), cardinal direction (*east*, *west*) and proximity relation (*is close to*). Based on this model, the query "*Which nuclear power plants are close to R" takes the form of: Close:O = CloseTo{o:O,{o}, R, {x1, y1, x2, y2},*

DistanceMethod, C} where O is the object type (nuclear power plant), *CloseTo* is a fuzzy set membership function, and *DistanceMethod* is a distance calibration method (Arpinar, 2006). Study results are applying on national security domain. This approach is accelerating and improving the process in emergency response life cycle.

Several studies were made about reasoning mechanism with the development of geo-ontologies (Smart 2007, Grütter 2007, Wang 2007, & Miron, 2007). All studies' common depth point is spatial reasoning by using topological relations. The use of qualitative spatial reasoning as a tool to support the development of a geo-ontology management system was analyzed in Smart's study. In this study, components of the framework are described and the implementation of the spatial reasoning engine is presented. System comprises the geo-ontology system and geo-location storage system parts, these spatial rules management and the spatial inference engine and the error management system (Figure 2.6). The system has been implemented using OWL, the Jena toolkit and Oracle Spatial.

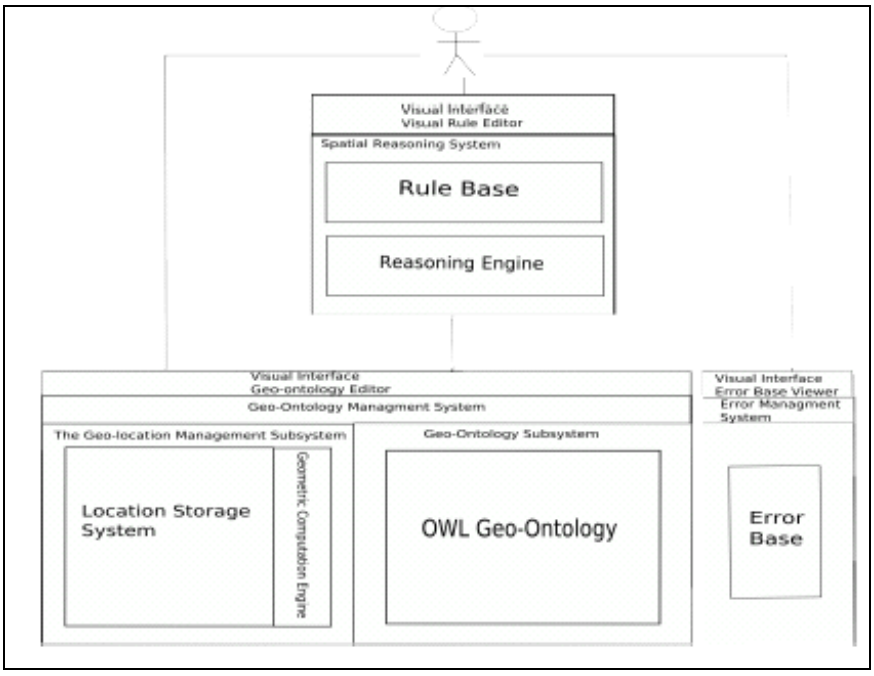


Figure 2.6 Framework for representing and maintaining geo-ontologies (Smart et al., 2007)

Another study on spatial reasoning is which spatiotemporal reasoning on environmental data with the combination of RCC and OWL (Grütter & Messmer,

2007). This study purposes easy-to use semantic search by adding semantic layer on environmental databases.

Database is held environmental inventory in which The core of the Datacenter Nature and Landscape (DNL) of the Swiss Federal Office for the Environment (FOEN) which is operated by the Swiss Federal Institute for Forest, Snow and Landscape Research (WSL). This database keeps many different biotopes its own vocabulary. All biotopes spatially interlace each other. But the main concern is answering the queries such as "endangered butterflies in Birmensdorf and neighboring communes" (Grütter & Messmer, 2007). In this context, it is aimed to integration of semantic layer on database. during conceptualization, RCC representation is used for defining relations between spatial objects. But RCC does not represent enough represent both OWL 1 and OWL 1.1. enough, Description Logic (DL) and RCC are used together for developing system in many such studies (Figure 2.7).

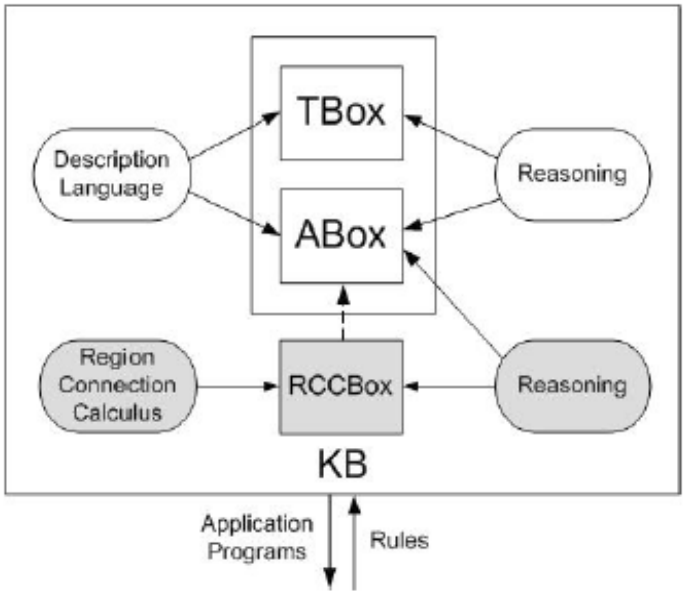


Figure 2.7 RCC and OWL hybrid knowledge representation system (Grütter & Messmer, 2007)

Even though, in principle, all RCC relations can be geometrically computed and as-serted in the ABox of an OWL DL knowledge base, it is, for at least two reasons. First, asserting for all pairs of regions; all relations holding between them easily

results in a very knowledge base thereby bearing on the performance of the system. Second one is inferences implied by the entries in the RCC composition tables which cannot be put in terms of OWL DL axioms (Grütter & Messmer 2007). Geo-ontology design and reasoning studies with OWL are practicing with Wang, Gong & Wu (2007) for spatial reasoning. Geo-ontology reasoning framework is designed with by using mereology, mereotopology and general theory of spatial location. Jena library and RACER reasoner are used while making spatial reasoning. Results are published with ArcIMS on map.

Spatial reasoning purpose automatic discovery of such relations will improve the current capabilities of existing search engine. Miron, Gensel, Villanova-Oliver & Martin (2007) aimed to realize complex management and analysis process by bounding different sources using semantic web. Spatial reasoning was made by designing OntoAs spatial ontology. Web sites are annotated with I-annotate plugin and results are converted into OWL with XSLT and then added into ontology with Protégé program. Topological, orientation and distance relations are used for spatial reasoning. Another challenge proposed by this study is while can not describe distance is numeric or qualitative exactly revealed problems in reasoning terms. For example, fuzzy concepts were founded as qualititative distance relations (very close, close, far and very far). If numeric distance relations, measurement is more clear than the other such as (1 km, 100 meters).

Integration between data sources is increased by developing and becomes widespread of spatial data infrastructures (SDI). SDI responds simple queries, but cannot rule out of data discovery automatically from different data sources. More expressive data and intelligent search engine are needed for accomplishing task. Ontology was used for querying data and expressing spatial data on the SDIs (2006) for the first time. Lutz & Kolas (2007) in rule-based discovery in SDI study more realistic assumption that usually several datasets have to be combined in order to answer the given questions and that discovery mechanism should consider this. The approach is illustrated by an example from the domain of disaster management.

According to scenario, Tsunami happened and SDI searches available airports in 50 km around. Some problems are revealed while disaster manager uses SDI's for managing disaster. For example, disaster managers search available airport for landing, there are different schemas with different vocabulary in two SDI's. Disaster managers (DM) shouldn't reach which type of planes can land at the airport while examining schemas. DM decides to categorize airports according to landing field measurement with its own experience. DM has to examine different schema for making this decision. DM has to join two different schemas together and query from this. This operation is very hard and time loss during disaster. We can categorize potential problems under discovery, interpreting schema and search strategies. For example, some difficulties while obtaining information due to lack of metadata, in other words there is no detailed information about the column names after tables found in SDI's. DL and ontology were used for solving these problems. Domain ontology contains basic terms (the primitives) of a domain which can be combined to describe the semantics of a feature type in separate application ontologies. Rules are determined with SWRL language between local schema and domain ontology by using schema mapping at spatial data sources. Data in the databases or WFS as individuals are attached into ontology with these rules.

Several studies are made for providing web services via SDI for more expressivity semantically. Andrei, Berre, Costa, Duchesne, Fitzner, Gurcar & Zastavni (2008) developed SWING environment and proposed Semantic Web Service (SWS) concept for solving heterogeneity problem on geospatial web services.

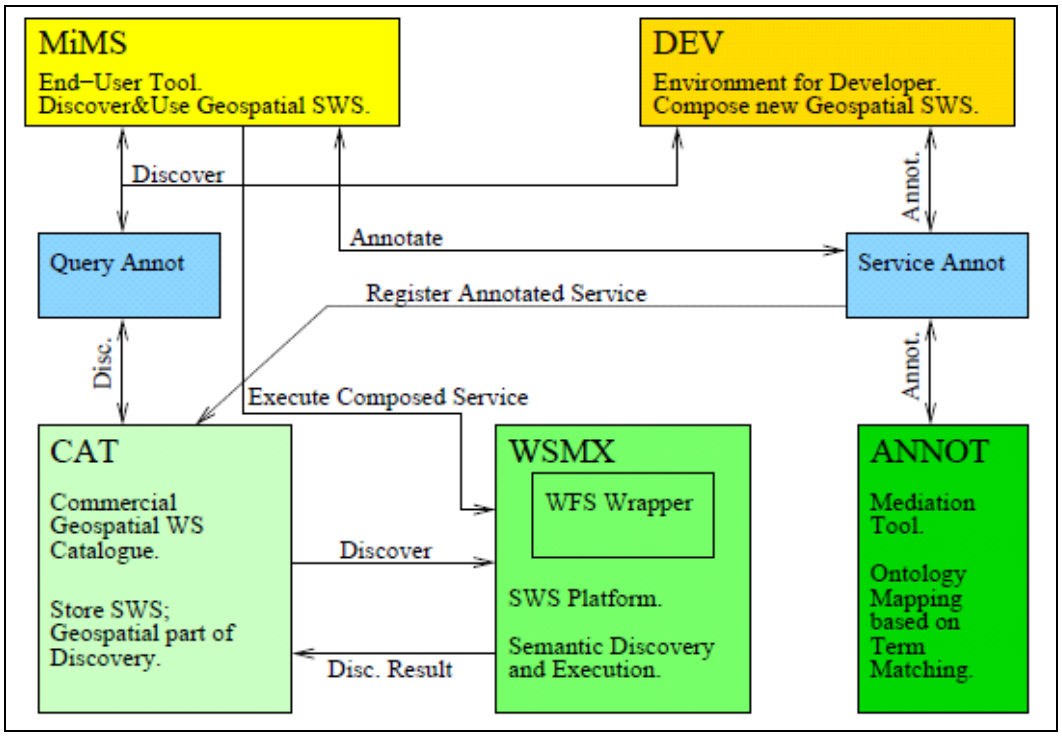


Figure 2.8 SWING environment frameworks (Andrei et al., 2008).

The SWING environment supports the entire life-cycle of geospatial SWS (Figure 2.8); to this end, it integrates: a genuine end-user tool (called MiMS); a tool (DEV) supporting the composition of new geospatial web services from existing ones; a commercial service catalogue (CAT); a state-of-the-art SWS platform (WSMX); and a tool (ANNOT) that helps with the creation of the semantic annotations (Andrei et al., 2008).

Fallahi (2008) developed a new perspective by using GIS services on environmental models. Layer based ontology is proposed for describing spatial services for measurement units. Measurement units are separated as nominal, ordinal, interval and ration derived geo-services on environmental model (Chrisman, 1995). For example, rain flow value as ratio scale and potential of erosion is described as nominal value. This study proposes solving heterogeneity problem during geo-processing operations data from geo-services.

Alam, Khan & Thuraisingham (2008) use Geographic Resource Description Framework (GRDF) language for making spatial data semantic aware and more

expressive is another ontology approach. GRDF uses logical inference and dynamic content aggregation.

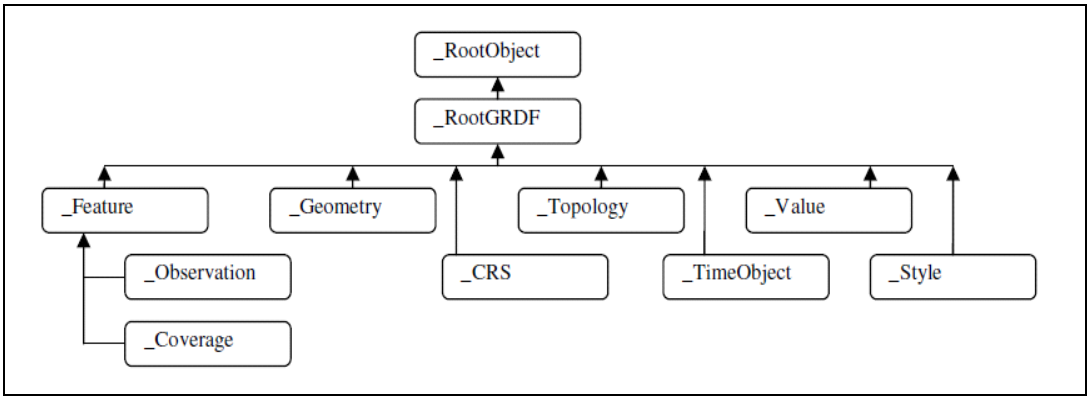


Figure 2.9 GRDF ontology diagram (Alam et al., 2008)

Concepts such as geometry, feature, topology, crs occur on GRDF ontology (Figure 2.9). Spatial objects are conceptualized with relations on GML while using GRDF OWL language.

Zaharia, Vasiliu, Hoffman & Klien (2009) made mineral resource management application for France building firms for using more spatial web services at complex queries. This study proposes calculating aggregate production and consumption of mines using in terms of management. This operation calculates with acquired data from different web services. First discovering services about request and calculating production amount and finally publishing on map.WSMOLX frameworks attached into geospatial web service for making semantic execution of geospatial queries (Figure 2.10).

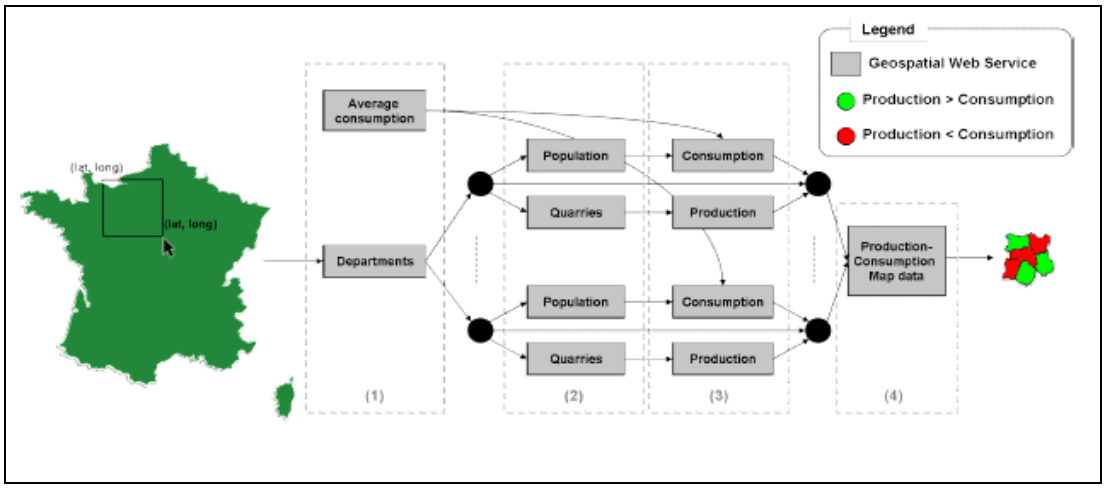


Figure 2.10 Mineral resource management use case description (Zaharia et al., 2009).

The other problem with regards to interoperability between institutions heterogeneous structure arising data sources format are different from each other. Boucher & Zimanyi (2009) realized ontology-based study for solving this heterogeneity structure. This study proposes to produce meaningful results by using information in different format and data model at defense firm. First request sends and calls information in WFS format and then WFS converts into GML with XSLT. GML attached into ontology via XSLT again. Instance from GML is mapped into domain ontology and makes inferences with reasoning mechanism. Instances are converted into GML with XSLT and converted to expected format Feature Manipulation Engine (FME) and published to the client (Figure 2.11).

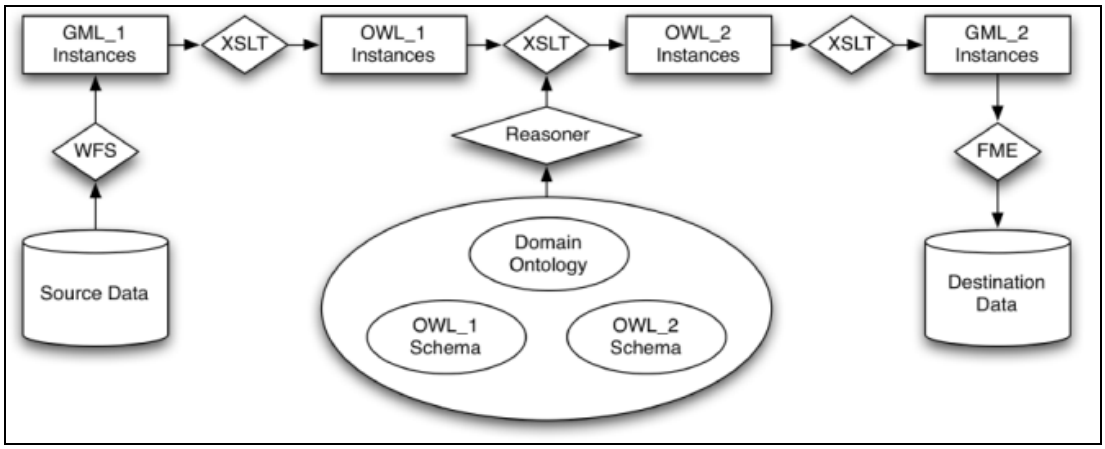


Figure 2.11 Heterogeneous data translation process with XSLT (Boucher & Zimanyi, 2009).

In this context this study was regarded to use FME technologies as the first application. Zhang, Zhao, Li & Osleeb (2010) allow searching and accessing geospatial data and services at the semantic level based on their content instead of keywords in the metadata with similar framework in the above (Zhang et al., 2010). This study provides modeling of WFS and WMS in ontology semantically. Ontology supports automatic discovery, composition, invocation and orchestration of these data services. Spatial data or spatial databases are published with WFS and WMS and translated with WSDL and then WFS and WMS services registered and published using OWL to directory of ontology-based catalog services after OWL process.

Another study presents an approach to expressing and using feature level semantics, which is implemented to support spatial data discovery, with a particular emphasis on its value for legacy datasets. This approach supports better data discovery through the use of formal semantics and the reuse of legacy data without requiring its transformation. Ontology designed with OWL language and GML standards and PostGIS as data store are used in this context. Mapping process between database and ontology is made with D2RQ engine (Table 2.3). Actually D2RQ engine generates virtual, reads only RDF model of the registered dataset and handles Jena API class translating them into database. Besides Jena library uses SPARQL query language for query process.

Table 2.3 Ontology mapping from database structure

Database Field	Ontology Construct	Construct Type
building_id	Building	class
structureName	hasBuildingName	Properties

WFS has limitation to automatic geospatial feature discovery and WFS composition. To overcome these limitations Zhang et al. (2010) proposed a solution for searching, discovering and composing semantically heterogeneous transportation spatial data at feature level from different sources over the web through WFS. In this context DL, DL-based reasoner, inference rules and ontologies were used to support

geospatial feature data interoperability at the semantic level besides they developed automatic geospatial feature discovery and wfs composition algorithms. First WFS descriptions and OWL ontology are mapped with *getCapabilities* method each other. Geo-ontology was developed with RCC 8 extended rules.

WFS feature types are represented as Let Q= (T', P', G', B') be a service query. T= Type name, P= feature Property Name, G=is the geometry, B=Bounding Box are defined for geospatial feature discovery algorithm. WFS items are indexed after defining descriptions. Formal query description score procured with client queries is mapped into ontology on next part. Similarity score between two geometry types of feature sets is presented which matches WFS feature with matching algorithm. Mapping WFS features, properties and geometries to ontology class, this process has to be done manually. For example TransitRoute class is created and occurs *wfs:routes* as a feature, shall make both of manually mapping each other. All WFS features are attached into ontology with this study. Both T-Box and A-Box reasoning can be made with conditions. But T-Box performance is better than A-Box. The challenge is how to improve the efficiency of geo-reasoning and how to handle automatic discover then semi-automatic.

Fonseca & Llano (2011) designed an ontology named Data-Representation Ontology for solving heterogeneity problem as in the other studies. This ontology describes the semantic embedded in geo-data, which cannot be represented in current type of ontologies. They propose a new method for the automatic generation of the data representation ontologies based on pattern classification techniques. Moreover this article shows how the topological relationships can enrich the semantics in the generated ontology and increase the effectiveness of spatial analysis (Fonseca & Llano, 2011).

According to Kalabokidis, Athanasis, Vaitis, Chang & Bran (2011) conventional geo-portals that rely on keywords-typing for searching data of interest are not feasible for interoperability. They proposed OntoFire environment leverages the semantic and spatial relationships between the resources by providing ontology-

based and spatially-based navigation mechanisms. Onto Fire was the first ontology-based geo-portal in the area of wildfire disaster. Their challenge is "How to find useful information in geo-portal environments by browsing using hyperlinks instead of keyword based queries." in this study. Onto-Fire ontology is used as upper-level ontology basic elements of the core metadata set of ISO 19115, such as the dataset title, abstract or the reference date (Figure 2.12).

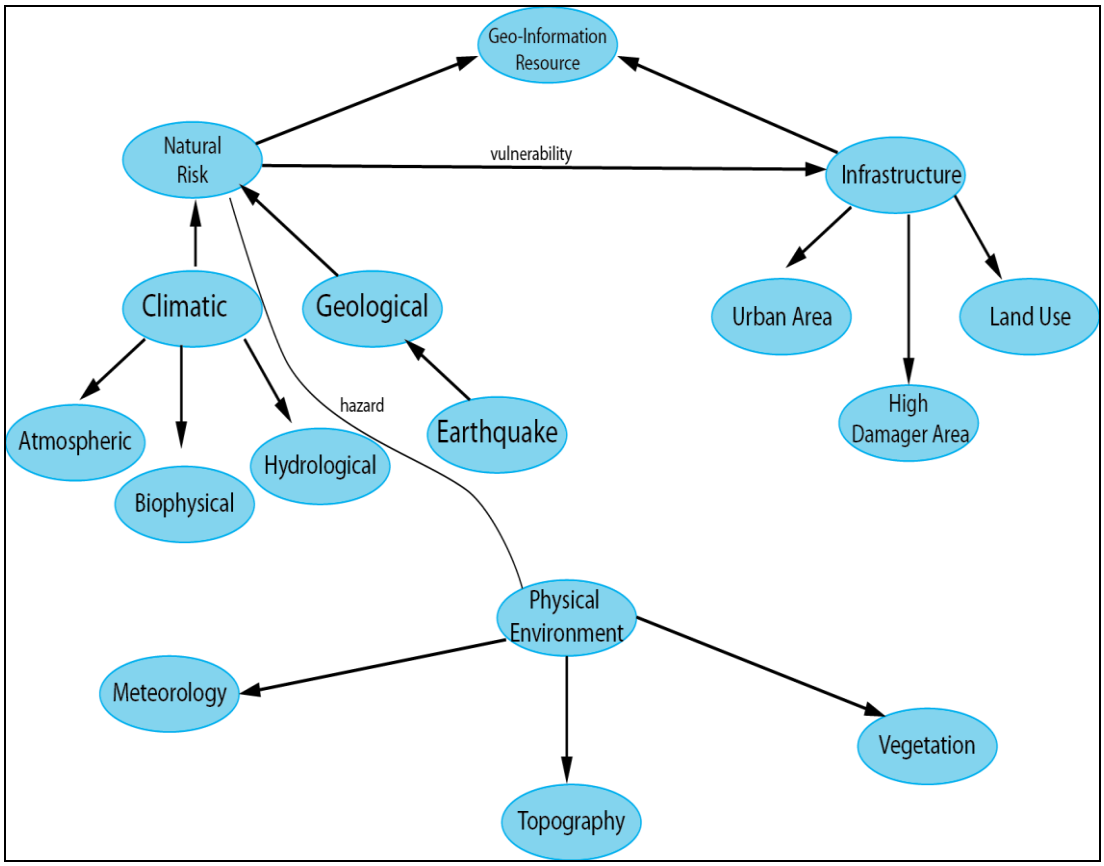


Figure 2.12 OntoFire ontology (Kalabokidiset al., 2011).

OntoFire provides an easy and intuitive way to search also for data "*semantically close*" to the specified class in preferred geographical area. First user specifies spatial and semantic criteria and finds resources classified under a specific category, second chooses resources from the result list and user view or download resources. OntoFire uses PostgreSQL and OpenLayers for viewing resources in map. They used RQL query language for querying. They used RDF language for conceptualization. A future research goal is to use OWL semantic language for higher expressiveness to enrich OntoFire ontology (Kalabokidis et al., 2011).

Several studies are made for more efficient spatial reasoning with developing geo-ontologies (Christodoulou, 2011 & Zheng, 2012). Both spatial reasoning can be made with using Pellet reasoning mechanism in OWL language in these studies. Christodoulou used CHOROS engine supports consistency checking and query answering for spatial information in RCC 8. Therefore can answer SPARQL queries with spatial and non-spatial relations and also present SOWL; a spatial reasoner for both relation calculation implemented in SWRL and runs under PELLET. In this context they are extracting spatial relations from the raw spatial data depends on the application and is not part of the reasoning mechanism. They calculate directions each other according to coordinate's size of two points $P_1(X_1Y_1)$ ve $P_2(X_2Y_2)$ if $X_1<X_2$ => P_2 *northOf* P_1 statement can be made for finding direction relation between point objects (Figure 2.13). At least one SWRL rule should be written for line and polygon objects.

$$Average\ Point = \frac{(\mathrm{Xmax}, \mathrm{Ymax}) + (\mathrm{Xmin}, \mathrm{Ymin})}{2}$$

Figure 2.13 Average point formula for distance calculation (Christodoulou, 2011).

Topological and directional relations are examined in this context but near, far concepts which fuzzier are research further studies. Semantic web technologies are used to increase discovery and use of spatial web services. Tian & Huang (2012) introduce and an approach for incorporating ontology into UDDI results and proposes an advertisement algorithm for representing service metadata contained in OGC capabilities document. Proposed approaches and service oriented architecture strategies, a web-based prototype system have been developed in order to help users to query, access and visualize spatial data (Tian, 2012). Metadata on WFS getting capabilities document with UDDI standard are mapped into ontology. Results are visualized into client by querying SPARQL language. But automating the process augmenting web services with semantic information is still critical challenge.

Auguste provides interoperability with integration of other vocabularies and institutions on study. They propose some alignments between various vocabularies (DBpedia, Geonames, Schema.org, Linked-Geodata, and Foursquare) in order to enable interoperability while interconnecting French geodata with other datasets (Atemezing & Troncy 2012). Upper-level spatial ontologies are examined in this context. Besides GeoSPARQL language is an extension of SPARQL query language used for making spatial queries. In addition to this, it made comparison of most available ontology according to GeoSPARQL query language needs.

Geo-vocabulary	Topological Functions	GeoSPARQL Requirements	Standard Followed
Ordnance Survey Spatial	`easting, northing, touches, within, contains`	Part of Req 4	OpenGIS Simple Feature
Ordnance Survey Topography	`contains, isContainedIn`	Very small part of Req 4	OpenGIS Simple Feature
Place Ontology	`in, overlaps, bounded_by`	Small part of Req 4	N/A
NeoGeo Spatial	All RCC8 relations	Part of Req 3; Req 6	Region Connection Calculus (RCC)
NeoGeo Geometry	—	Req 10 - 14	N/A
FAO Geopolitical	`isInGroup, hasBorderWith`	–	–
OntoMedia Space	`adjacent-below, adjacent-above, orbit-around, is_boundary-of, has-boundary`	–	–

Figure 2.14 Comparison of some geo-vocabularies with respect to the GeoSPARQL requirements.

They made some comparison about triple store which holds spatial data (Figure 2.14). They detailed about triple store such as Parliament, OpenSahara, Virtuoso which achieves about GeoSPARQL query language and functions.

As a result there are several upper-level ontologies in geographic domain like Geo-net ontology in GREASE II (Geographic Reasoning for Search Engines) (Lopez-Pellicer, Silva, Chaves & Rodrigues, 2009). Pre-existing ontologies such as SWEET created by NASA (Raskin, 2005) can be used as an upper- level geo-ontology. NeoGeo Spatial Ontology for describing topological relations between geographic concepts by Juan Martin Salas. GeoSPARQL is an Open Geospatial

Consortium (OGC) standard used for geo-ontology design. GeoSPARQL ontology describes spatial concepts and relationships. There are some reasons for choosing GeoSPARQL ontology for upper-level ontology. GeoSPARQL reuses common geometry serialization formats and relatively straightforward to implement on top of a spatial database and file system. Secondly GeoSPARQL has OWL-based spatial reasoning capabilities. Finally this methodology is implemented and attached appropriate geo-ontology to disaster ontology in earthquake domain.

2.5 Disaster Ontologies and Case Studies

Geo-ontologies are discussed for providing interoperability in Chapter 2.4 above. In this section disaster ontologies will be examined in disaster management domain. Semantic interoperability studies on disaster management have begun since 2005. In this context studies purpose efficient, effective and quick response and interoperability between institutions. Kirsh, Peterson & Lenert (2005) have developed an initial ontology for geospatial and context aware technology with using semantic web technologies. They study how Metropolitan Medical Response System (MMRS) units conceptualize the physical space of a disaster and their organized response. This study aimed to redirect resources to the right places with taking location information continuously by placing wireless devices to field worker. At the same time disaster ontology is created and integrated with devices. Another semantic GIS based disaster management study is made by Tanasescu. Semantic web services are used in this study while the determined web services are not sufficient for providing semantic interoperability. SWS is more fleixible, resuable and universe access than web services. This offers invocation, composition, mediation ve execution of complex service (Tanasescu, 2006). In order to transition to semantic web GIS, they adopt Web Service Modelling Ontology (WSMO) into system.

Spatial data Infrastructures are used often in disaster management as multi-disciplinary field. Xu & Zlatanova (2007) made ontology based disaster management to enhance the discovery and retrieval of geospatial data sources from SDI. Disaster ontology designed for helping to make decisions and manage all phases of disaster,

and domain ontology designed for each data sources for using during disaster. Disaster ontology was mapped with all domain ontologies. Fan & Zlatanova (2010) aims at building and developing an innovative software infrastructure for supporting collaborative work of events handling in emergency situation. Databases in distributed data sources are converting into OWL with RDB2OWL transformation modules. This study aims to solve data sharing between institutions. In case Sootodeh & Kructhen (2008) is developed disaster ontology by converting Unified Modelling Language (UML) into OWL which includes all organization concepts. Disaster ontologies are used at all phases besides not specifically at one phase. Joshi et al. is designed disaster ontology for mitigation phase of disaster management. *CityOnt*, *DisasterOnt* and *ISont* ontology are developed for this purpose. Simulation is made with creating rule by Jess rule engine on flood disaster for testing ontology. This disaster ontology is very important for literature which is the first tested ontology.

The significance of spatial information on disaster management is mentioned above. In this context spatial reasoning is very important requirement on disaster ontologies. Lucas & Werder (2008) aimed to made spatial reasoning for solving semantic gap from messages during disaster. Topological and neighbourhood relations are defined for spatial reasoning. Object center points which defined by Delaunay method rather than the distance to each objects for neighbourhood relations. Iwanaga, Nguyen, Kawamura, Tahara & Ohsuga (2011) applied information retrieval techniques to twitter messages after disaster like information retrieval from messages during disaster on disaster management system above. For example it's very difficult decision to find available meeting place for disaster victim with several twitter messages which contains different concept. Although there is a meeting place in 1 minute walk, will be informed missing information due to not inefficient making information inference. If twitter messages can explain more expressivity, it can reach correct location information of meeting place. In this study they defined tweet proposition with class and properties. After they created disaster ontology and queried with SPARQL languages. Obtained class and properties were attached into ontology in a semi-automatically. Chou, Zahedi & Zhao (2011) made

another study on information inferencing about disaster. They made information inferencing from plain text on web sites about disaster with grounded-theory approach.

Yang, Lin & Chen (2011) created typhoon ontology named SUMO. They described formal and hierarchical domain ontology of typhoon disaster and some typical chain patterns of typhoon disaster were identified based on this ontology. SUDO ontology can be used to support communication and coordination between organizations effectively and ensure the disaster recovery plan to be effective.

Kolas, Hebeler & Dean (2005) examined ontology architecture approach for providing geospatial semantic system. Five different ontology types are used for geospatial semantic system. Each ontology type represents a key role in establishing a rich, dynamic, and flexible geospatial knowledge base. Ontology is created on emergency domain and applied into aircraft emergency scenario for the first time. Application designed on landing a specific aircraft on available landing field at emergency situation. They created a knowledgebase which holds OWL and receives queries in the domain ontology. Data returned from the WFS and translated from GML to OWL, and SWRL rules are applied and then result is then returned to the user. Distinctive features of this work from other studies are that they used WFS data sources for providing interoperability between institutions.

Yusuf, Su'ud, Boursier & Alam (2012) realized information sharing between heterogeneous data sources at another study. Previous efforts in information retrieval and integration have been limited to content-based and keyword-based search. Most of the researchers in this area still focus on semantic based query and integration of proprietary GIS data formats from multiple GIS data sources viewing, creating and manipulating maps. Architecture is created named Multi-format Information Retrieval, Integration and Presentation MIRIP (Figure 2.15). They attached metadata into file formats and database and then mapped into ontology

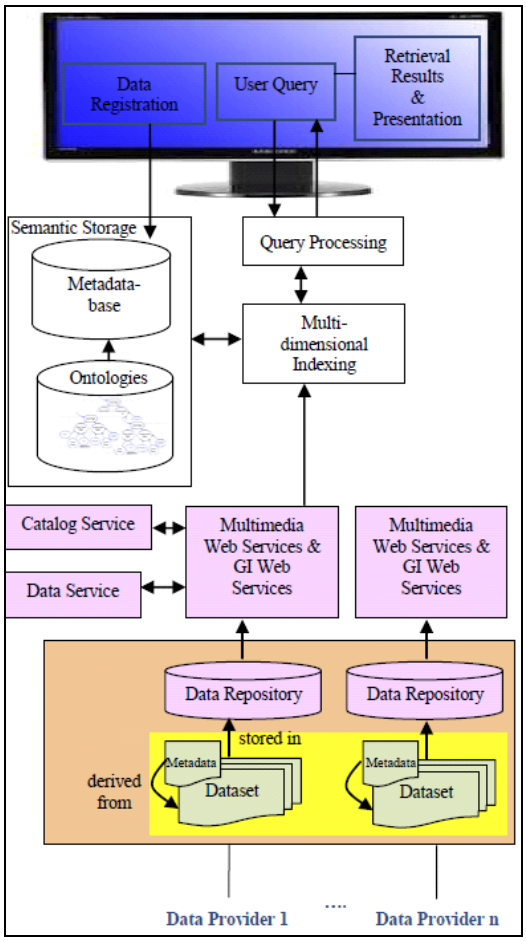


Figure 2.15 The MIRIP (Multi-format Information Retrieval, Integration and Presentation) architecture (Yusuf et al., 2013).

Open Ontology Model (OOM) for ontology building. Hybrid ontology is organized with creating ontology for each data sources. Query engine and reasoning functions are revealed with Jena API and ontology is created with Protégé environment. Application was applied and it allows querying more efficient heterogeneous data sources on disaster management domain. Another study in disaster domain aims at designing of adaptive framework for supporting collaborative work of different actors in public safety and disaster recovery missions. Ontology is designed for in an autonomous and collaborative way. New response types are defined according to the received data from rescue robots and disaster personnel by using SWRL rules.

Statictical methods and experiences are used at traditional disaster studies. But this methods are lack of semantic and intelligent reasoning. Ma & Yu (2013) purpose a new ontology approach which ontologies increase interoperability abilities with providing information sharing. Ontology is designed with analyzing disaster factors and historical development. The hidden influences factor or disaster chain information can be excavated by the reasoner with domain-specific rules. Jena inferences engine is used for keyword parsing and ontology reasoning. Related information is inferenced and visualized into user with keyword search. Recently semantic web technologies are also used for making sensor data into semantic interoperable in disaster domain. Lezcano, Santos, Garcia-Barriocanal (2013) attached sensor data in Emergency Response Management (ERM) messages into semantic sensor network ontology. Thus they provide data transfer between heterogeneous systems. SWRL Rules are integrated to trigger alerts and SWRL language solutions to increase rule-based systems interoperability from the semantic web perspective. Finally Middleton (2013) described their knowledge-based service architecture for multi risk environmental decision support capable of handling geo-distributed heterogeneous real time data sources. They attached metadata into sensor data and provided semantic interoperability which heterogeneous sensor data comes from disaster field

CHAPTER THREE

ONTOLOGY DESIGN

3.1 Description of the Study Area

Turkey is located in a seismically active region where approximately half of the country is declared as first order earthquake zone. 1999 Golcuk and Duzce earthquakes which struck the Marmara region initiated disaster management awareness in the country. All provinces have been taking measures for mitigating hazards of which Izmir lies in a highest earthquake risk region. The city is surrounded by several active faults which produced hazardous earthquakes of magnitude six and higher in the past. Several cities are densely populated under earthquake risk. Besides, they do not have strong ground motion network which is costly and needs advanced technological systems. Deriving damage and loss estimation with lack of strong ground motion network is very challenging task as mentioned in Section 1. For this reason İzmir is chosen as a case study because of the lack of this network and being under earthquake risk.

3.2 Modeling Framework

The objective of this study is to develop a semantic GIS based disaster management system. The methodology of this study is composed of four consecutive steps including disaster ontology design on earthquake domain, spatial data annotation, DL based damage assessment with using reasoning capabilities and finally publishing results on map (Figure 3.1).

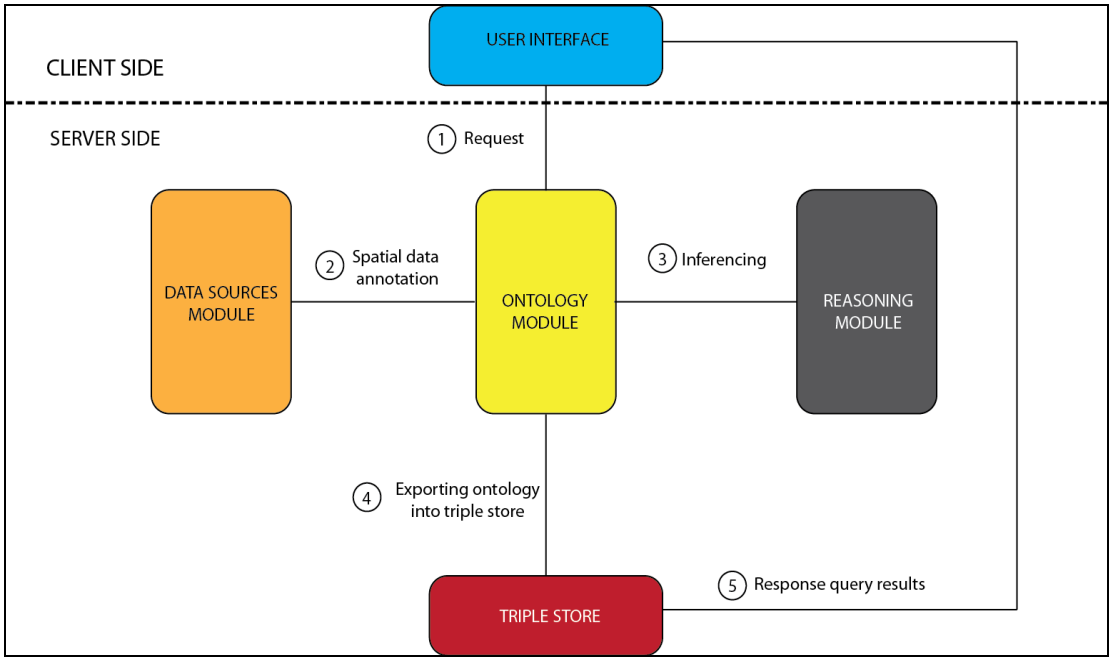


Figure 3.1 Semantic GIS based disaster management system framework

In this thesis, it is aimed to design disaster ontology for conceptualization of disaster management in earthquake domain with the proposed framework. Several methodologies are presented for developing disaster ontologies in literature. These methodologies are investigated in section 3.1. Heteregeneous data sources are attached into ontology after combining disaster ontology and GeoSPARQL ontology which is chosen as upper-level ontology (Challenge 1). Some transformation techniques are used for converting data source into standard GML format and annotated into OWL language. DL-based damage assessment capabilities with reasoning module using creating pre-defined rules after individuals were attached into ontology (Challenge 3). Finally this framework aims to manage disaster efficiently and effectively by using GeoSPARQL and solving interoperability between institutions.

3.3 Disaster Ontology Development Methodologies

Ontologies provide interoperability among information systems and a shared, common understanding of a domain between heterogeneous systems. While building domain ontology, different methodologies are investigated in literature and these

types are changing in upper-level (more general) to bottom-level (more specific); general or upper-level, domain level and representation ontologies. The classification of ontologies based on their granularity and thematic scope into top-level, domain, task and application ontologies introduced by Guarino (1998). Van Heist & Guarino proposed a more flexible and pragmatic classification into local and global ontologies. General or upper-level ontologies define very generic concepts and relations like geo-ontology and Semantic Web for Earth and Environmental Technology (SWEET) ontology which can be used in general terms. Domain ontologies are specific ontologies like pizza ontology in which concepts are in specific topic.

There are several methodologies for ontology development (Öhgren & Sandkuhl, 2005). In the literature there are seven methodologies which (1) Enterprise Ontology, (2) TOVE, (3) Unified Ontology, (4) Methontology, (5) Sugumaran and Storey, (6) Noy and McGuiness, (7) Sure et al. for ontology development (Öhgren et al. 2005). The three most representative methodologies (Pinto H., 2004) are TOVE (Gruninger & Fox, 1995; Gruninger, 1996), Enterprise (Uschold & King, 1995; Ushold, 1996b) and Methontology for building ontologies from scratch. TOVE and Enterprise belong to the first generation and Methontology belong to second generation of ontology building methodologies (Pinto, 2004).

These methodologies are similar each other but still have some advantages and disadvantages. The enterprise ontology was developed to provide interoperability from different institutions. This method for development of was ontologies proposed by Uschold & King (1995) and have "Identify the purpose and scope, build the ontology, evaluate and document" stages. This methodology has advantages and disadvantages as the other methodologies; the use of middle-out approach instead of a bottom-up or top-down approach is advantageous situation, besides maintenance stage is not explicitly referred to in this methodology. TOVE ontology was developed in TOVE project by Gruninger and Fox. The TOVE methodology is based on scenarios and competency questions and proposes "capture motivating scenarios, formulate informal competency questions, specify the terminology in a formal

language, formulate formal competency questions in First Order Logic (FOL), specify axioms and definitions for the terms in the formal language, evaluate competency and completeness" stages. TOVE methodologies stages are not exposed good enough from the other methodologies and maintenance and documentation stages are not explicitly referred as enterprise methodology. TOVE methodology proposes FOL as the knowledge representation language has to be used (Pinto, 2004). This means that a transformation language have to be used in TOVE methodology for converting natural language to syntactic language for ontology development.

Methodologies propose some stages respectively (1) Specification, (2) Conceptualization, (3) Formalization, (4) Implementation, (5) Maintenance (Figure 3.1) according to usual terminology in ontology engineering (OE). Some stages are not clearly divided into different stages in TOVE and Enterprise methodology. Methontology provides the most accurate descriptions of each activity in ontology life cycle (Brusa, Caliusco & Chiotti, 2008). One of the most important issues that separates Methontology from other methodologies is ontology reuse attend at the ontology development stages. Thus methontology uses object oriented programming for ontology development which is very useful on programming techniques for ontology developers. Earthquake modeling and earthquake ontology creation are targeted for the purpose of disaster management in disaster domain. In accordance with this purpose the above-mentioned methodologies are examined one by one; Methontology methodology is chosen which is capable of providing ontology reuse in other domains, widely used by developers, easy to use and ontology development cycle usage during ontology design. Methontology has continuously been applied in several projects and updated (Fernandez-Lopez, Gomez-Perez & Juristo, 1997). Ontology development life cycle is used for building ontologies from scratch with chosen methodology (Figure 3.2).

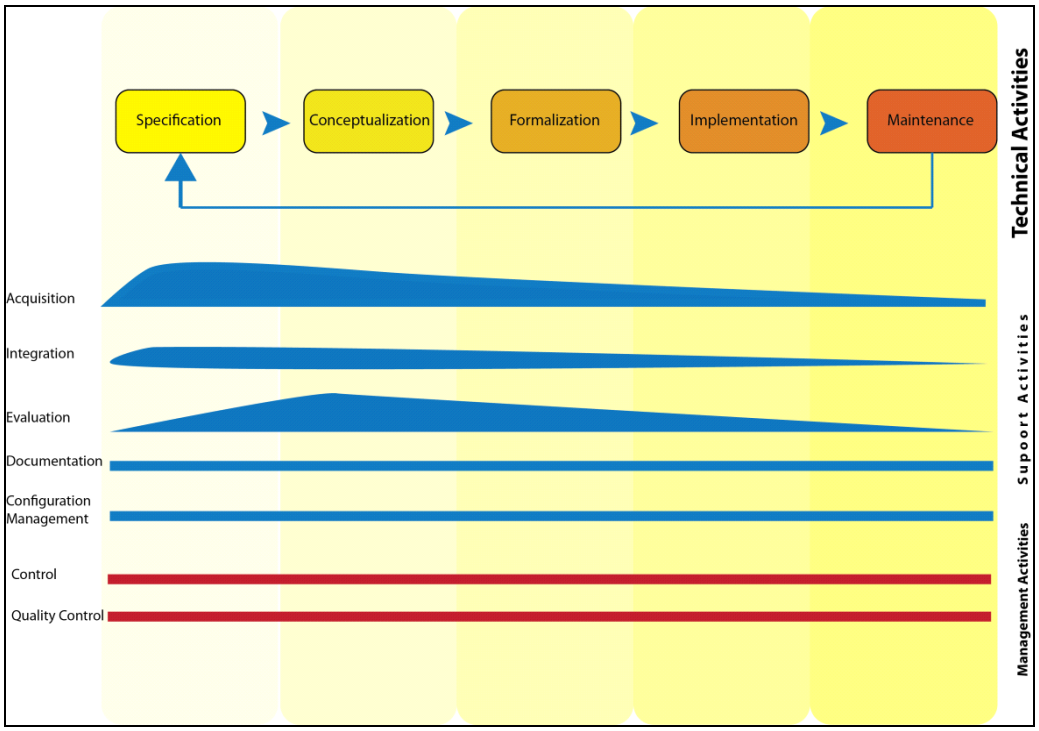


Figure 3.2 Ontology development life cycle

3.4 Disaster Ontology Development

Objective of this section presents the implementation process for disaster ontology. For this purpose we conceptualized disaster and possible relationships as a model in earthquake domain and applied to the development of ontology for Izmir disaster and emergency management center. Specification, Conceptualization and implementation and maintenance phases within ontology development cycle are performed in ontology development phases.

3.4.1 Specification

The purpose of the specification phase identifies definitions and requirements for ontology development. Firstly, this phase summarize as domain description such as defining earthquake and Disaster Management System (DMS) terms, describing disaster management cycle, constituting essential scenario for domain ontology, identifying and specifying ontology aims and goals, and proposing competency

questions. Secondly, glossary and taxonomy is created by revealing all terms about disaster domain. The sub phases of specification phase are determined as (1) describing domain, (2) scenario and competency questions, (3) determining ontology goal and scope.

3.4.1.1 Describing Domain

Ontology development process starts with describing domain. Therefore specifying terms and relations are provided as brainstorming with experts about disaster management in this phase. There is an existing disaster management system (DMS) is called İzmir Disaster and Management Information System (İZAYBİS) but this DMS uses traditional working process and so is not interoperable. İzmir Disaster Management System is prepared for providing a coordination between Working Groups and "provincial disaster and emergency management center" (Figure 3.3) that may occur potential disasters in İzmir and it is intended to construct provincial disaster aid plan with DMS. Existing system has gained a favor for establishment of the proposed structure. There are 16 different institutions getting information about disaster in working groups and these are: Communication, Transportation, Wreckage removal, Research and rescue, Health, Preliminary damage assessment, Security, Military, Distribution, Electricity, Water and sanitation, Natural gas, Temporary housing, Environmental, Press and public relations services. They have well-defined registered protocols with provincial disaster aid plan on how to cooperate with each other in case of a disaster. Data exchange and interoperability cannot be done between working groups owing to some institutional problems. Center generates solution for overcoming these problems. Files are sent to the center via CD device or via email in a different format on yearly periods. These files are converted in suitable format and entered to the system by center employees. This situation doesn't allow syntactic and semantic heterogeneity in DMS.

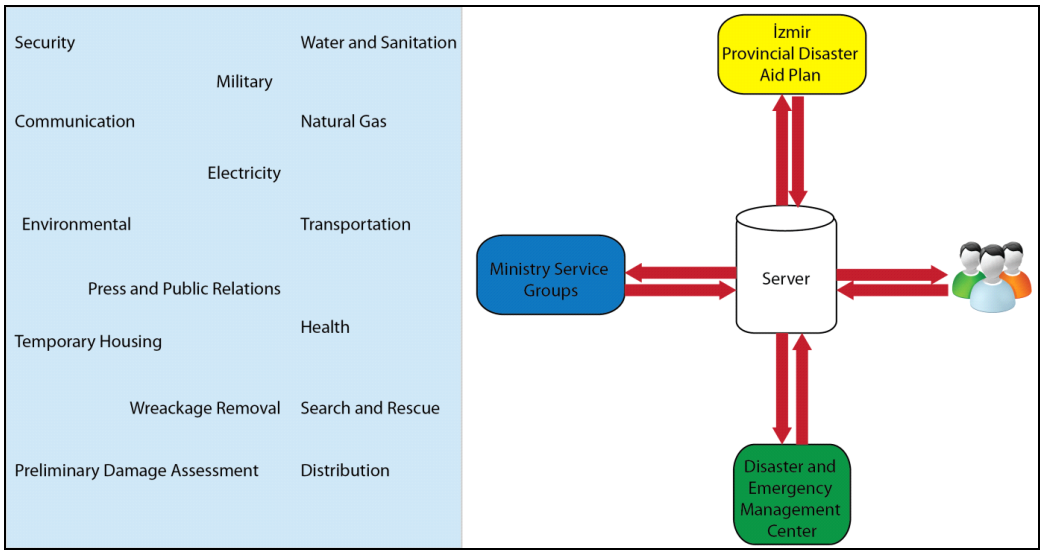


Figure 3.3 İzmir disaster management structure

3.4.1.2 Scenario and Competency Questions

Scenario and competency questions give advice about which concepts, relations and rules are used on ontology design. It is inquired that what kind of questions are required to answer during ontology design with competency questions. It is determined capabilities and limits of ontology by proposing various questions to accomplish this. These kinds of questions are called competency questions (Table 3.1) in ontology design. For example road, bridge and heavy equipment's terms are added to the disaster ontology concepts with questions like "where available roads, bridges and heavy equipment are those after disaster".

Table 3.1 Competency questions

1 How long is the impact area of the disaster? How many dead and injured people are affected from disaster? How many buildings are severe, moderate and light damaged from disaster?
2 Where are damaged roads and bridges, Where is available construction equipment?
3 Where are the closest health and first aid service group personal and how many health and first aid people are available?
4 Which roads are secured for delivering logistic supplies to meeting places and tent cities?
5Which tent city is established and where?

DMS Scenario (Table 3.2) is prepared after competency questions determination for system capabilities. Disaster ontology is designed in accordance with this scenario. In this scenario, phases are carried out step by step, and concepts are determined into ontology. Scenario reveals brainstorming studies with DMC experts and they elaborate with sample event on some big earthquake situation. All operations comprise on earthquake event and respond to event with working service groups.

Table 3.2 Disaster management scenario

1 An event occurred (6.5 magnitude, 2 km depth and 3 minutes duration).
2 Determined event types (earthquake).
3 Event is described as a disaster if event threshold is above some value (VI degree).
4 Message is sent to the working groups.
5 Natural Gas is closed by Natural Gas Group.
6 Preliminary damage group is gone to the hazard area.
7 Damage is calculated.
8 Security 1 (police) and Security 2 (military police) has sent to the hazard place for securing area.
9 Wreckage removal service group is sent to the hazard area for removing wreckages from roads and opened close roads.
10 First aid and health and search and rescue service groups are sent to the area for first intervention and rescuing injured people.
11 Health services led to injured people to hospitals.
12 Management center is determined meeting places for affected people from disaster
13 Management center is decided to deliver logistic supplies from available logistic support center and which roads are used this operation.

3.4.1.3 Identification of Ontology Goals and Scope

The purpose of the ontology is conceptualizing all terms about disaster domain and using necessary information in which attended different institutions towards

same goal during disaster with disaster ontology development. At first, terms are identified with domain experts in disaster domain and comprised and then terms are classified in the glossary (Table 3.3), so taxonomy is created by glossary terms which upper version of glossary with classification of terms.

Table 3.3 Glossary of terms (Summarized)

Event	Road	Intensity	Vehicle	City
Impact	Bridge	Content	Magnitude	Light
Ahead Distribution Center	Ambulance	Clothing	Depth	Moderate
Disaster	Aid	Food	Injured	Severe
Damage	Tent	Furniture	Employee	Earthquake
Logistic Support Control Center	Building	Structural	Death	Neighbourhood
Meeting Place	Ministry Group	Police Office	Bodily	Location
Security Control Point	District	Tent City	Hospital	Environment

In order to describe spatial characteristics of geographic features, Open Geospatial Consortium (OGC) published a standard GeoSPARQL ontology in 2012. It supports representing and querying geospatial data on the Semantic Web. GeoSPARQL defines a vocabulary for representing geospatial data in RDF, and it defines an extension to the SPARQL query language for processing geospatial data (OGC, 2012). The GeoSPARQL comprises six different as core, topology vocabulary, geometry, geometry topology, RDF entailment and query rewrite components (Figure 3.4). For example, a system based purely on qualitative spatial reasoning may support only the core and topological vocabulary components

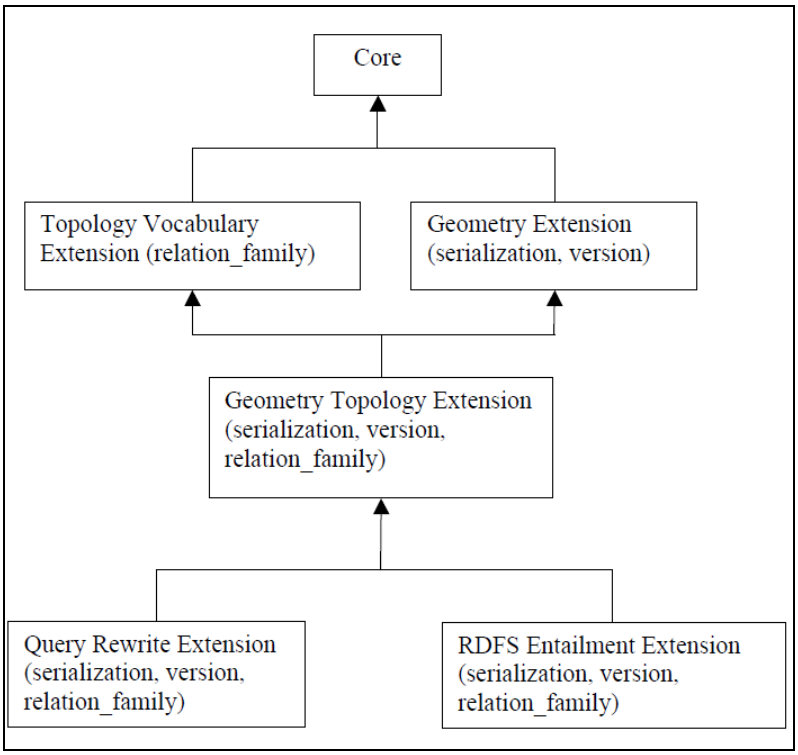


Figure 3.4 GeoSPARQL ontology components (GeoSPARQL, 2011)

GeoSPARQL has defined two main classes *Feature* and *Geometry* under *SpatialObject*. *SpatialObject* class represents everything that can have a spatial representation. *Feature* class represents objects features. Geometry class represents spatial representation. GeoSPARQL has a strong class hierarchy where all the geometric classes are derived from geometry class. Point, line and polygon objects which are used to represent the real geographic features in the world are represented by the primitives' geometries. In addition there are more complex geometries like polyhedral surfaces under geometry class (Figure 3.5).

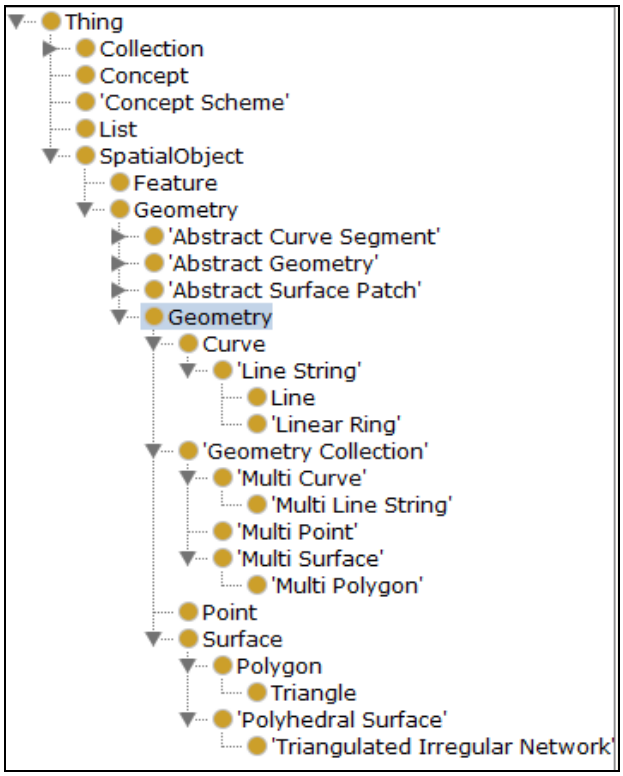


Figure 3.5 Basic concepts of geometry class

Finally, *'geo:hasGeometry'* property defines spatial features in ontology. *'geo:hasPointGeometry'* property determines geometric objects for features. Data properties were used for indicating string values like '25' as integer and as:gml or as:wkt type. GML type means coordinates in a gml format or wkt (well known text) means coordinates in text format (Figure 3.6). Data properties are used as GML for representation of spatial objects relationship because the data source is based on GML.

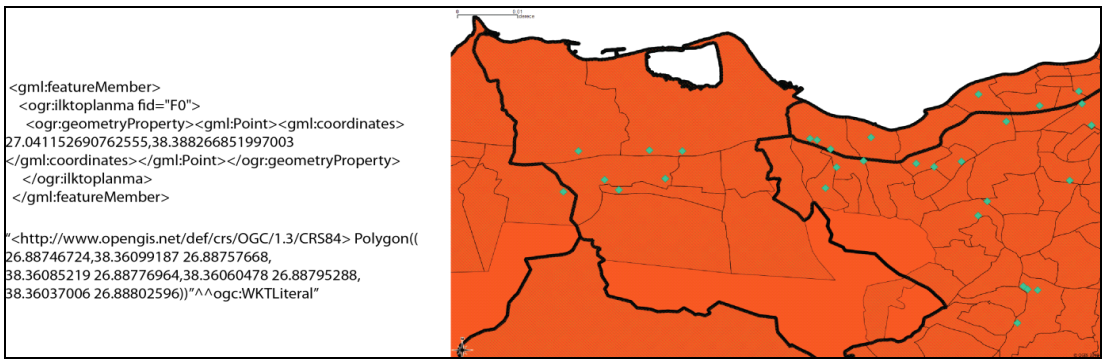


Figure 3.6 Spatial objects in WKT and GML types

GeoSPARQL ontology supports simple feature, egenhofer and RCC8 topological relations. Simple features relations are *sfEquals*, *sfDisjoint*, *sfIntersects*, *sfTouches*, *sfCrosses*, *sfWithin*, *sfContains* and *sfOverlaps* used in SPARQL graph patterns. Egenhofer topological relations shall allow the properties as *ehEquals*, *ehDisjoint*, *ehMeet*, *ehOverlap*, *ehCovers*, *ehCoveredBY*, *ehInside*, *ehContains* in triples. At the same time GeoSPARQL ontology supports RCC8 relation family *rcc8eq*, *rcc8dc*, *rcc8ec*, *rcc8po*, *rcc8tppi*, *rcc8tpp*, *rcc8ntpp*,*rcc8ntppi* properties. Actually all of three relation families define the same property in different format (Figure 3.7) and both of three are held in GeoSPARQL ontology.

Simple Features	**RCC8**	**Egenhofer**
equals	equals	equal
disjoint	disconnected	disjoint
intersects	¬ disconnected	¬ disjoint
touches	externally connected	meet
within	non-tangential proper part + tangential proper part	inside + coveredBy
contains	non-tangential proper part inverse + tangential proper part inverse	contains + covers
overlaps	partially overlapping	overlap

Figure 3.7 Equivalent simple features, RCC8 and Egenhofer relations.

3.4.2 Conceptualization

Class creation is provided with the specified terms in glossary which were created under specification phase and determination of the class hierarchy with taxonomy. Relationships are determined between classes with the creation of the properties. Rules and restrictions are designed as natural language with competency questions and scenario descriptions.

3.4.2.1 Defining Classes and Hierarchy

Relevant terms in ontology are specified while defining classes. Class hierarchy is stated with relationships between classes each other. For this purpose basic terms are defined like *'disaster'* if required these terms can be specialized as *'earthquake'* or generalized as *'event'*. Bottom-up hierarchy so 'general term to special term' is adapted to conceptualization while designating class hierarchy (Uschold, 1996). Earthquake and disaster terms are used for identifying class from glossary and we created taking into account the class hierarchy (Figure 3.8).

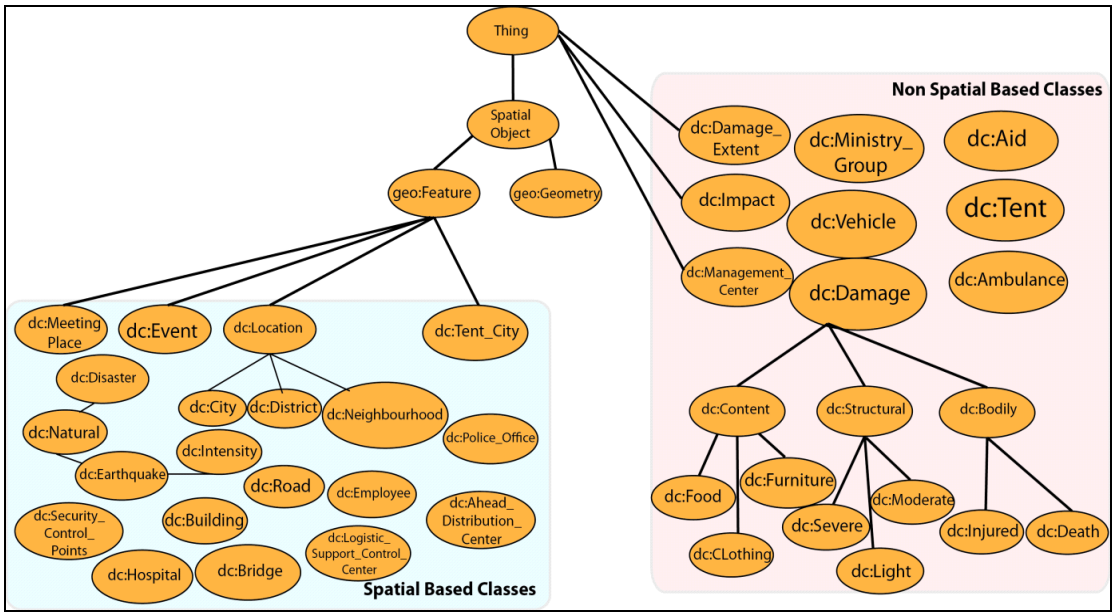


Figure 3.8 Disaster ontology class hierarchies

Earthquake phenomenon is described with conceptualization of earthquake epicenter, hypocenter, magnitude, depth and intensity, acceleration terms during ontology design based on earthquake domain. Earthquake parameter is analyzed from glossary of terms on earthquake which was determined by the United States Geological Survey (USGS, 2012) and interviewed with the expert geologists from our country. Classes and properties are created with these parameters. Earthquake data from Kandilli Observatory are collected in the '*dc:earthquake*' class. Earthquake which *'magnitude > 6.0'* is determined with specific restrictions are

contained as disaster in '*dc:disaster*' class. Disaster damage collects as '*dc:damage*' class for determining damage assessment efficiently. Soil types as '*dc:soilType*' and earthquake ground acceleration value as '*dc:acceleration*' determine at the model for the calculation of damage. We used *geo:feature* and *geo:geometry* class as a sub class *geo:SpatialObject* from upper-level geo-ontology classes and spatial based classes are attached into *geo:feature* class for being spatial enabled (Figure 3.8). Non-spatial classes are attached as subclass of '*Thing'* class in ontology.

3.4.2.2 Defining Property and Class Relations

OWL properties represent relationships. There are two main types of properties, object and datatype properties. Object properties are relationships between two individuals. Datatype properties are relationships between individuals and literal. There are several object properties as functional, inverse functional, transitive, symmetric, antisymmetric, reflexive, irreflexive properties. If a property is functional, there can be at most one individual that is related to the individual via the property. Functional properties are also known as single valued properties. Inverse functional properties are that the inverse property of functional property. For a given individual, there can be at most one individual related to that individual via the property. Transitive Property relates A to individual to individual B and also individual B to individual C, and then we infer that individual A is related to individual C. Symmetric properties means that both individuals represent symmetric properties. Antisymmetric properties are inversed of symmetric properties. Reflexive properties when the property must relate individual itself. Irreflexive properties are inversed of reflexive properties. In brief Figure 3.9 shows all kind of object properties in OWL. These properties are used for making better reasoning mechanism.

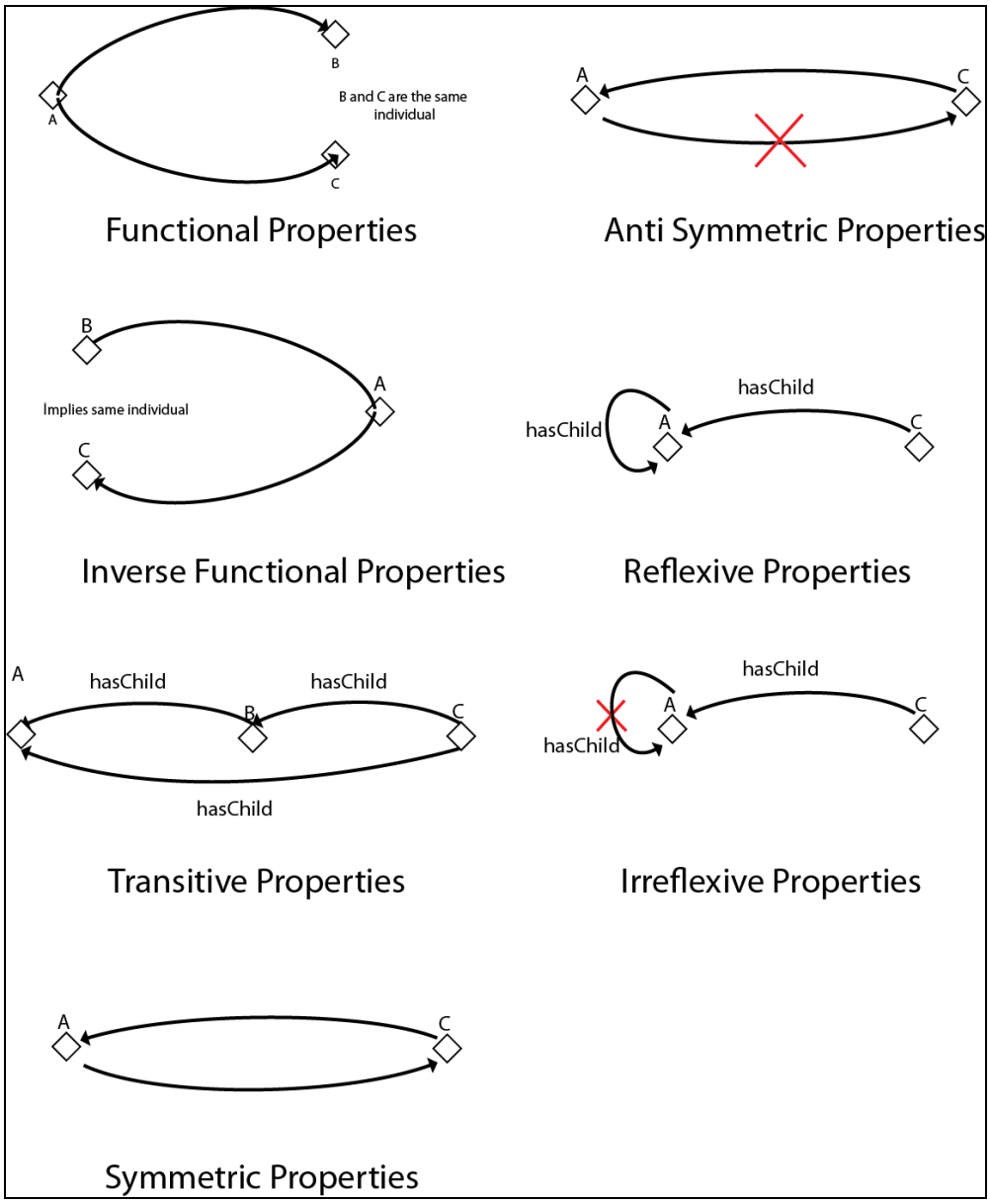


Figure 3.9 Types of object properties in OWL

Specifying 'domain and range' for classes are stated with class relationships after the creation of classes and class hierarchy determination. According to the scenario; '*dc:earthquake*' class is related to '*dc:hasMagnitude*', '*dc:hasDepth*', '*dc:coordinates*' data properties. '*dc:defined*' is used for defining earthquake as a disaster. 'dc:cause' property is defined between '*dc:damage*' and '*dc:acceleration*' classes for calculating disaster damages. '*dc:patientSentTo*' property is described between '*dc:firstaid*' and '*dc:hospital*' classes for people who suffered from disaster for orientating to the hospitals.

Properties are delinated in two ways in this phase; first geo-ontology properties are used for spatial properties. Non-spatial properties are specifying with negotiating experts in disaster management center (Figure 3.10). Domain and ranges were described after the creation of properties. Domain and ranges are specified as relationship between classes. For example '*dc:hasCity'* property is determined as a property between '*dc:Earthquake*' and '*dc:City*' (Figure 3.11).

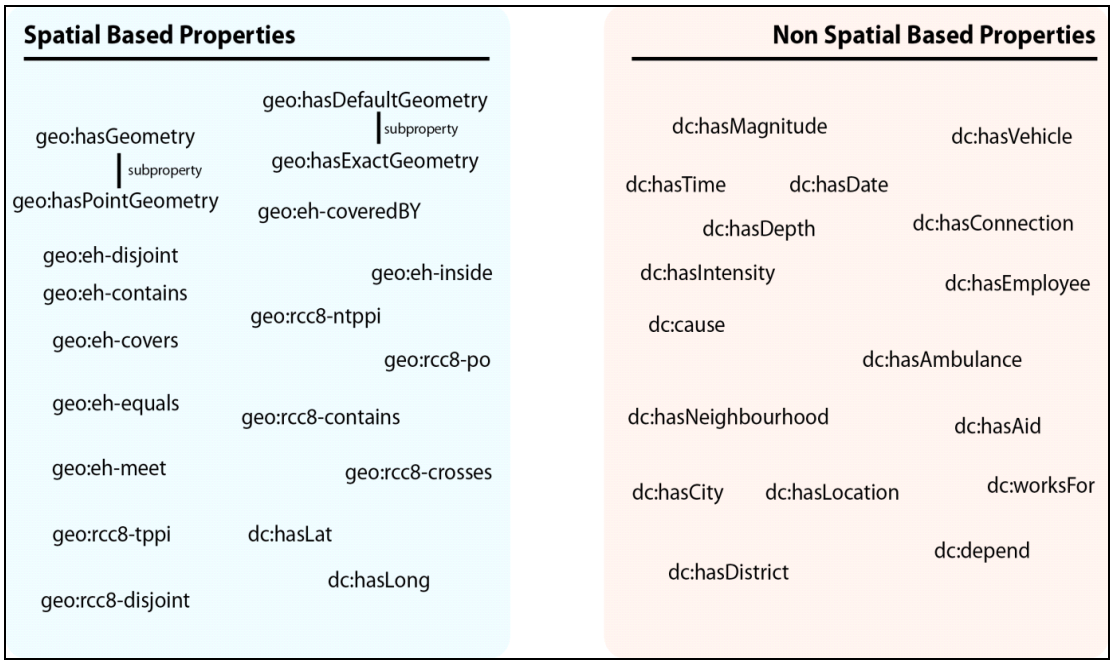


Figure 3.10 Definition of spatial and non-spatial based properties

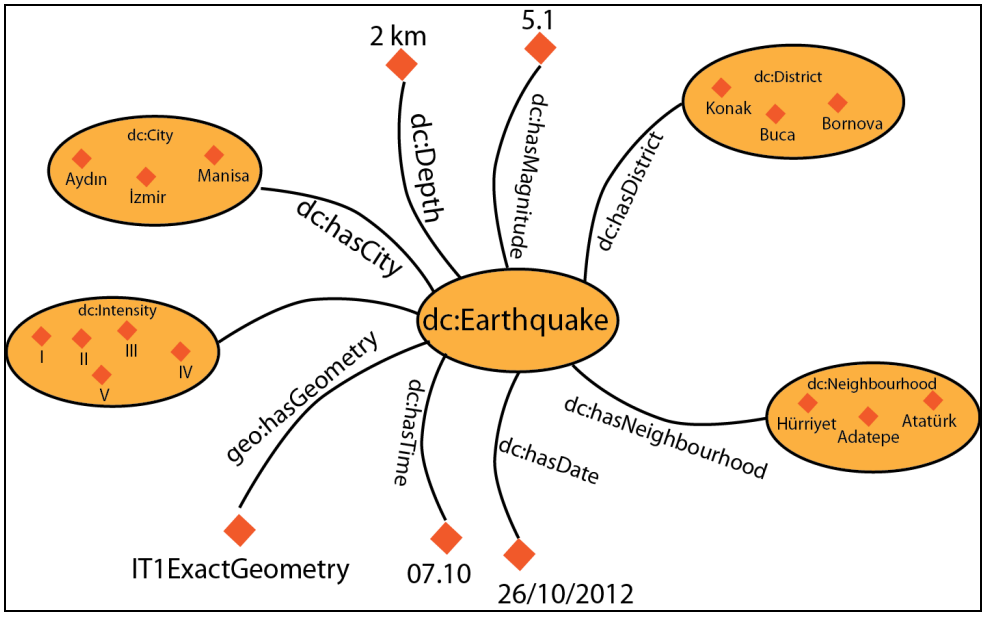


Figure 3.11 Disaster ontology class, property and 'Domain and Ranges' view

3.4.2.3 Representing Rules and Restrictions

In ontologies, properties describe relationships between individuals and data values or between two individuals. The key idea is that a class of individuals is described by the relationships that these individuals participate at the same time we can define such classes by using restrictions (Horridge M., 2009). A restriction describes a class of individuals based on the relationships that members of the class participate in.

Basics of restrictions on ontology stay on description logic (DL) rules. DL rules have been developed into one of the main knowledge representation formalisms. The maturity of the field is also reflected by the adoption of description logics as prior specification paradigm for ontological descriptions of the standardization of the OWL by the W3C Consortium. Besides that DL is used in artificial intelligence for formal reasoning on the concepts of an application domain. DL is a family of formal knowledge representation languages. It is more expressive than propositional logic but has more efficient decision problems than first-order logic. There are many varieties of DL and there is an informal naming convention (Table 3.4). DL types diversify $\mathcal{ALC}$ (Attributive Language with complement) is the simplext one most

complex $\mathcal{SROIQ}^{(\mathcal{D})}$ which means (Attributive language with complement + reflexivity and irreflexivity + object value restriction + inverse properties + qualified cardinality restrictions) through. $\mathcal{ALC}$ was proposed by Manfred Schmidt-Schauß and Gert Smolka in 1991 and is the basis of many more expressive DL. For example OWL-DL is based on $\mathcal{SHOIN}^{(\mathcal{D})}$ (Attributive Language with complement + subproperties + object value restriction + inverse properties + cardinality restrictions). Besides that OWL 2 provides the expressiveness of $\mathcal{SROIQ}^{(\mathcal{D})}$DL and OWL-Lite is $\mathcal{SHIF}^{(\mathcal{D})}$ (Attributive Language with complement + reflexivity and irreflexivity + inverse properties) DL. Most of todays main stream DL's are, in fact sublanguage of $\mathcal{SROIQ}^{(\mathcal{D})}$.

Table 3.4 Naming conventions of DL

$\mathcal{AL}$	Attributive language. This is the base language which allows: Atomic negation ,Concept intersection, Universal restrictions, Limited existential quantification
$\mathcal{S}$	An abbreviation for $\mathcal{ALC}$ with transitive roles.
$\mathcal{H}$	Role hierarchy (subproperties - rdfs:subPropertyOf).
$\mathcal{R}$	Limited complex role inclusion axioms; reflexivity and irreflexivity; role disjointness.
$\mathcal{O}$	Nominals. (Enumerated classes of object value restrictions - owl:oneOf, owl:hasValue).
$\mathcal{I}$	Inverse properties.
$\mathcal{N}$	Cardinality restrictions (owl:cardinality, owl:maxCardinality).
$\mathcal{Q}$	Qualified cardinality restrictions (available in OWL 2, cardinality restrictions that have fillers other than owl:Thing).
$(\mathcal{D})$	Use of datatype properties, data values or data types.

DL is based on three disjoint sets of primal elements; class, property and individuals. $\mathcal{SROIQ}^{(\mathcal{D})}$ knowledge bases convention is converted respectively TBox, RBox and ABox. The TBox contains intensional knowledge in the form of a terminology and is built through declarations that describe general properties of concepts. For example earthquake can be defined as a disaster by writing this declaration with TBox:

$$Earthquake \sqsubseteq Geological \sqcap Disaster$$

RBox captures interdependencies between the properties of the considered knowledge base. RBox is the union of a finite set of property characteristics together with a property hierarchy. For example

$$geo{:}hasGeometry \sqsubseteq geo{:}point$$

The ABox contains extensional knowledge also called assertional knowledge that is specific to the individuals of the domain of discourse (Nardi & Brachman 2007). For example states that the individual İzmir is a city

$$has City(İzmir)\ isACity?$$

Description logic is used for modelling our application domain. First Nardi & Brachman (2007) proposed for specifying knowledge base with natural language-style syntax, can be more appealing to the developer. For example our axioms are described like $\exists hasCity.İzmir$ and can be written (only city İzmir). The concepts of the domain are specified and characterized their relationships to other concepts and to specific individuals mentioned in the above. After concept definition restrictions are described which are a characteristic feature of DL.

OWL restrictions fall into three main categories as quantifier, cardinality and hasValue restrictions. Quantifier restrictions are categorized into existential and universal restrictions. Existential restrictions describe classes of individuals that participate in at least one relationship along a specified property to individuals that are members of a specified class and ***"some"*** keyword and (∃) symbol is used for indicating existential restrictions. Universal restrictions describe classes of individuals that for a given property only have relationships along this property to individuals that are members of a specified class and ***"only"*** keyword and (∀) is used for universal restrictions (Horridge, 2009). Existential restrictions are the most common type of restrictions in OWL ontologies. Existential restrictions mean the entire individual that have at least one class member.

All of the classes that we have created so far have only used necessary conditions to describe them with restrictions above. But necessary conditions alone cannot say that "If something fulfills these conditions then it must be a member of this class". To make this possible we defined necessary and sufficient conditions together with

Protege convert to defined class functions. Thus this means that not only are the conditions necessary for membership of the class, they are also sufficient to determine that any individual that satisfies them must be a member of that class. Classes that have at least one set of necessary and sufficient conditions are known as defined class, others defined as primitive classes. The reasoner can only automatically classify only defined classes with at least one set of necessary and sufficient conditions. Taking into account the scenario description, rules, restrictions and necessary and sufficient conditions were identified and represented in natural language (Brusa G. et al., 2008). These rules were formally represented in triple view. For example an earthquake inferred as a disaster with rule. In this way, the next step is decided according to the scenario.

Rule 1

Natural Language Rule:

If Earthquake magnitude is higher than 6.5 and time interval is higher than 1 minute earthquake is defined as Disaster

DL Rule:

$Earthquake \sqcap (> 6.5\ hasMagnitude \sqcap >1\ hasTime) \sqsubseteq Disaster$

SWRL Rule:

$Earthquake(?e) \wedge hasMagnitude(?e,?mag) \wedge hasTime(?e,?time) \wedge swrlb:greaterThan(?mag,6.5) \wedge swrlb:greaterthan(?time,1) \rightarrow Disaster(?e)$

Figure 3.12 Disaster definition rule

After specifying earthquake as a disaster with rule 1 (Figure 3.12), the next rule is determined neighborhood damage failure. Accordingly this, neighborhood is defined as severe, moderate and light damage with earthquake acceleration value and basement type. Acceleration value should be calculated for each neighborhood.

Rule 2

Natural Language Rule:

If earthquake acceleration greater than 12 and soil type "Soft" neighbourhood damage is severe

DL Rule:

$Neighbourhood \sqcap (hasSoilType\ "Rock") \sqcap (>12\ hasAcceleration)$

SWRL Rule:

$Neighbourhood(?n) \wedge hasSoilType(?s,"Rock") \wedge hasAcceleration(?n,?a) \wedge swrlb:greaterThan(?a,12) \rightarrow Light(?n)$

Figure 3.13 Neighborhood damage ratio rule

Damage assessment can be made on neighbourhood basis on city with these rules. For example Neighbourhood affects ‘Severe’ damage when it has big earthquake acceleration, according to alluvion basement type and distance from earthquake center. Some neighbourhood is affected ‘Light’ damage from earthquake which has Rock basement type and small acceleration value (Figure 3.13).

Rule 3

Natural Language Rule:

If neighbourhood damage is severe bridges can be break down

If neighbourhood damage is moderate bridges can be break down

If neighbourhood damage is light bridges can’t break down

DL Rule:

Severe $\sqsubseteq$ ∃ hasDestroy Bridge

Moderate $\sqsubseteq$ ∃ hasDestroy Bridge

Light $\sqsubseteq$ ∃ ¬hasDestroy Bridge

SWRL Rule:

Severe(?s) $\wedge$ *Bridge(?b)* $\wedge$ *hasDestroy(?s,?b)*

Moderate(?s) $\wedge$ *Bridge(?b)* $\wedge$ *hasDestroy(?s,?b)*

Light(?s) $\wedge$ *Bridge(?b)* $\wedge$ ¬*hasDestroy(?s,?b)*

Figure 3.14 Bridge situations in damaged area rule

After determining neighborhood damage conditions, most critical information is road and bridge situation which path to disaster location. To forecasting the bridge situations, if neighborhood damage is severe, all bridges can be destroyed in boundaries of the neighborhood, rule can be defined in ontology (Figure 3.14). In addition to this to forecasting roads situation, if neighborhood damage is severe and building height is more than 15 meters on the road rule can be used. Destroyed bridge and closed roads can be determined via both rules (Figure 3.15). Thus disaster manager foresees about which route and type of vehicle can be chosen for lead a disaster location.

Rule 4

Natural Language Rule:

If neighbourhood damage is severe and Building height is more than 5 roads can be closed?

If neighbourhood damage is severe and Building height is less than 5 roads can be closed?

DL Rule:

Severe Π Building Π (hasFloor>5) ⊑ ∃ hasClosed Road

Severe Π Building Π (hasFloor>5) ⊑ ∃ ¬hasClosed Road

SWRL Rule:

Road(?r) *Λ* Severe(?s) *Λ Building(?b) Λ hasFloor(?b,?f) Λ swrlb:greaterThan(?f,5)* → *ClosedRoad(?r)*

Figure 3.15 Road situations in damaged area rule

All inferences are not necessary for response at early post-earthquake. After the earthquake, distant logistics of resources are a challenge in itself. For example, tent cities are required to build for homeless people after earthquake (Figure 3.17). But nevermore there are not enough tents in cities, so tents have to be ordered immediately from other near cities. For this reason estimated tent number and the locations of tent city can be defined according to neighborhood population and neighborhood damage assessment. So rule 5 can be identified for beginning to preparations in this regard after early post-earthquake. This provides for homeless people ensured settle into tent city as soon as possible.

Rule 5

Natural Language Rule:

If neighbourhood damage is severe and population greater than 5000 buildt tentcity ?

If neighbourhood damage is severe and population greater than 2000 build tentcity ?

If neighbourhood damage is light and population greater than 5000 not build tentcity ?

DL Rule:

Severe Π Population Π (hasPop>5000) ⊑ ∃ tent_City

Severe Π Population Π (hasPop>5000) ⊑ ∃ tent_City

Light Π Population Π (hasPop>5000) ⊑ ∃ ¬ tent_City

Figure 3.16 Tent city requirements for damaged area rule

These $\mathcal{SROIQ}^{(\mathcal{D})}$DL rules determined restrictions in ontology for making decision efficiently with strong reasoning from ontology. Other rules are shown below;

- If neighbourhood damage is severe assign vehicle, police and soldier to neighnourhood,
- If neighbourhood damage and road are closed, assign vehicle to the neighbourhood,
- If homeless people are more than 500 pcs in the damaged neighbourhood, assign food and water distribution to the neighbourhood,
- If homeless people are more than 100 pcs in the damaged neighbourhood, build tent-city to the neighbourhood.

3.5 Creating Ontology with Protégé

In order to implement the ontology, Protégé software is chosen because it is extensible and provides plug-and-play environment that makes it a flexible base for rapid prototyping and application development (Knublauch, Horridge, Musen, Rector, Stevens, Drummond & Wang, 2005). Protégé 3.4.8 version is used which support OWL 2.0 and SPARQL support for querying ontology. GeoSPARQL ontology as an upper-level is imported into disaster ontology from URI with protégé and protégé is provided spatial abilities from geo-ontology class and property (Figure 3.17).

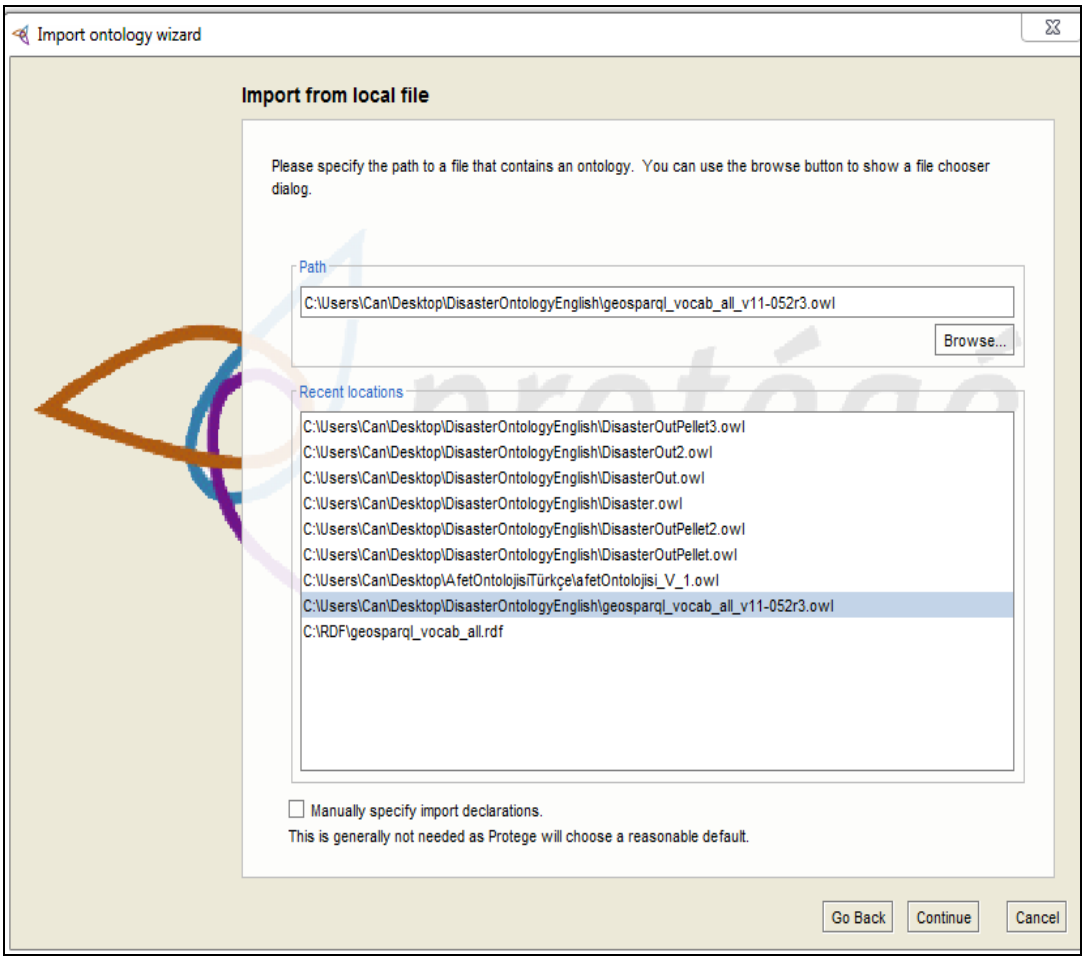


Figure 3.17 GeoSPARQL ontology import

Classes and properties which were designed above are created after importing GeoSPARQL ontology. If classes and properties have spatial feature, they are created under spatial object at GeoSPARQL ontology due to the fact that all classes and properties are separated spatial or non-spatial feature at conceptualization phase.

While ontology creation with Protégé classes and properties are added into ontology respectively. First disjoint relation is determined between class and properties. Flood is subclass of natural disaster and at the same time flood disjoint with earthquake and landslide is defined in Protége is equal to axioms in DL language as indicated $Flood \sqcap (Earthquake \sqcap LandSlide) \sqsubseteq \bot$ (Figure 3.18). After object properties are defined according to characteristics. For example object property is defined as transitive property if it has a transitive relation between classes.

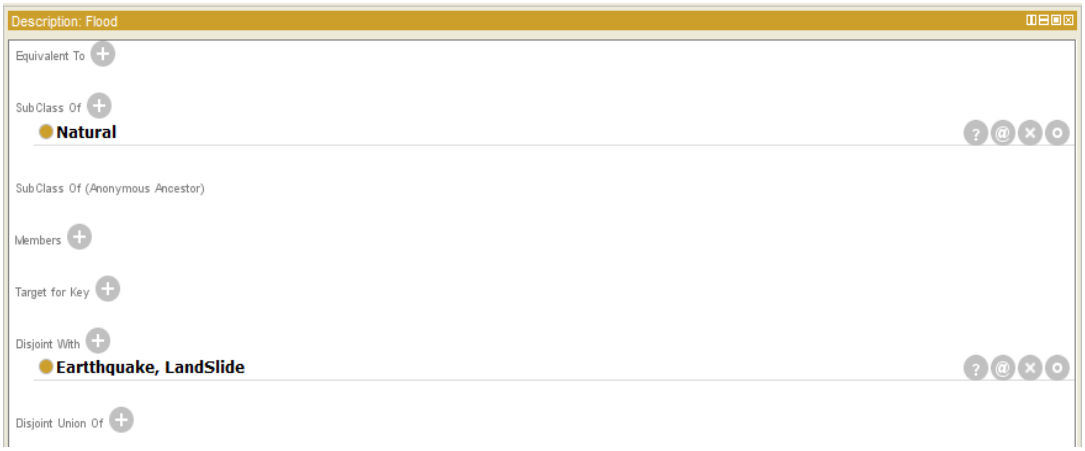


Figure 3.18 Disjoint class of the ontology

Class and property names given above are added into class and property tab and defined domain and ranges to the classes. Thus typed relations are defined between classes. For example *Aid* class is domain of *hasAid* property and *Ahead_Distribution_Center* or *Logistic_Support_Control_Center* is range of *hasAid* property (Figure 3.19).

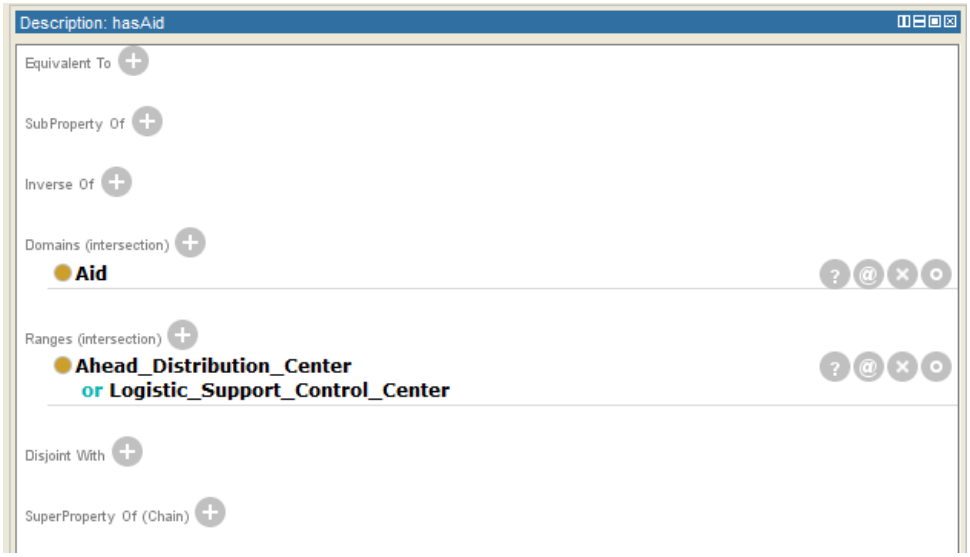


Figure 3.19 Domain and ranges in the ontology

Rules and restrictions described as natural language and description logic below are added in a formal way with Protege to the ontology for reasoning mechanism. Example; *"If Earthquake magnitude is higher than 6.5 and time interval is higher*

than 1 minute earthquake is defined as Disaster" natural language rule above is created for inferencing with ontology. Data property named as *hasMagnitude* is created for making restrictions as in the example. *Earthquake* and *Disaster* classes are created after restrictions. If individual is created as an *Earthquake*, this *Earthquake* inferred as *Disaster* while "*earthquake magnitude is bigger than 6 and time interval is greater than 2 minute, define earthquake as a disaster*". These rules come inferable results on future queries (Figure 3.20).

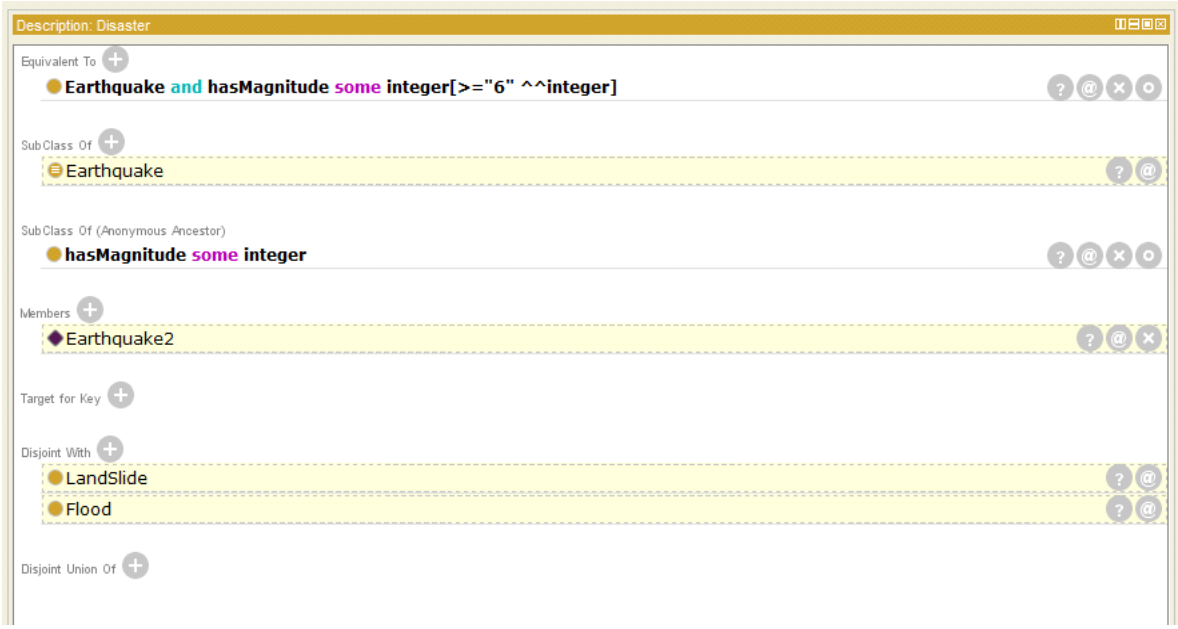


Figure 3.20 Restriction result in Protégé

Rule 2. "*If earthquake acceleration greater than 12 and soil type 'Rock' neighbourhood damage is severe*" rule is used for prediction of damage assessment on *Neighbourhood* class according to acceleration value and basement type value. Several models exist for assessment of damage after earthquake. In this study model based on acceleration value is used associate with distance from earthquake epicenter, magnitude, and depth and basement type parameters. Damage assessment is established with this model for which buildings destroyed or damaged predefined and provides accurate allocation of resources with this model. Horizontal acceleration value formula is calculated with basement type (Rock Soil, Hard Soil etc.), distance between epicenter and district, and earthquake magnitude value and

constant values in the above model (Hu, Liu & Dong 1997). Acceleration value is reduced while distance from fault is increased. Changes in ground conditions highly affect acceleration value. Damage estimation can be calculated with damage ratio on buildings as light, medium and severe damage and wreckage and severe wreckage according to building type (loam, brick or concrete) and acceleration value (Karagöz, 2012). Estimated dead and injured people are calculated with this calculation (Table 3.5). According to these disaster manager can decide to divert equipments accurate place and lead injured people to closest hospital and guiding damaged people to meeting places. Framework can decide fully automatic all stages by using data from different institutions. Acceleration value widely depends on geometric distances between epicenter (Active fault effective distance) and building, and earthquake waves pass on geomechanic size (rock type, basement type) (Table 3.6).

Table 3.5 Horizontal acceleration equations *a:Acceleration, D:Distance between epicenter and district, M: Earthquake Magnitude* (Hu, Liu &Dong, 1997).

Soil Type	**Equation**
Rock Soil	$a=46x10^{0.208M}(D+10)^{-0.686}$
Hard Soil	$a=24.5x10^{0.333M}(D+10)^{-0.924}$
Medium Hard Soil	$a=59x10^{0.261M}(D+10)^{-0.886}$
Soft Soil	$a=12.8x10^{0.432M}(D+10)^{-1.112}$

Table 3.6 Relations between magnitude, ground acceleration and building types

Magnitude	**Ground Acceleration**	**A (Loam)**	**B (Brick)**	**C(Concrete)**
V	12-15	%5 Light Damage	-	-
VI	25-50	%5 Moderate Damage %50 Light Damage	%5 Light Damage	-
VII	50-100	%5 Wreckage %50 Severe Damage	%5 Moderate Damage	%5 Light Damage
VIII	100-200	%5 Severe Wreckage %50 Wreckage	%5 Wreckage %50 Wreckage Damage	%5 Severe Damage %50 Moderate Damage
IX	200-400	%50 Severe Wreckage	%5 Severe Wreckage %50 Wreckage	%5 Wreckage %50 Wreckage Damage
X	400-800	%75 Severe Wreckage	%50 Severe Wreckage	%5 Severe Wreckage %50 Wreckage

In this phase *hasAcceleration* and *hasSoilType* property are created for describing acceleration of earthquake and basement type. For example Adatepe neighbourhood acceleration value determined as 20 and basement type value "Alluvion". *"Neighbourhood and (hasAcceleration some integer [>="15"^^integer]) and (hasSoilType only Alluvion)"* rule is determined at Severe class in ontology for assessing damage of neighbourhood. According to rule and definition above, Adatepe neighbourhood is inference in severe class as seen in the (Figure 3.21). In this context which neighbourhood affected from disaster can determine as prior knowledge without going to the disaster area.

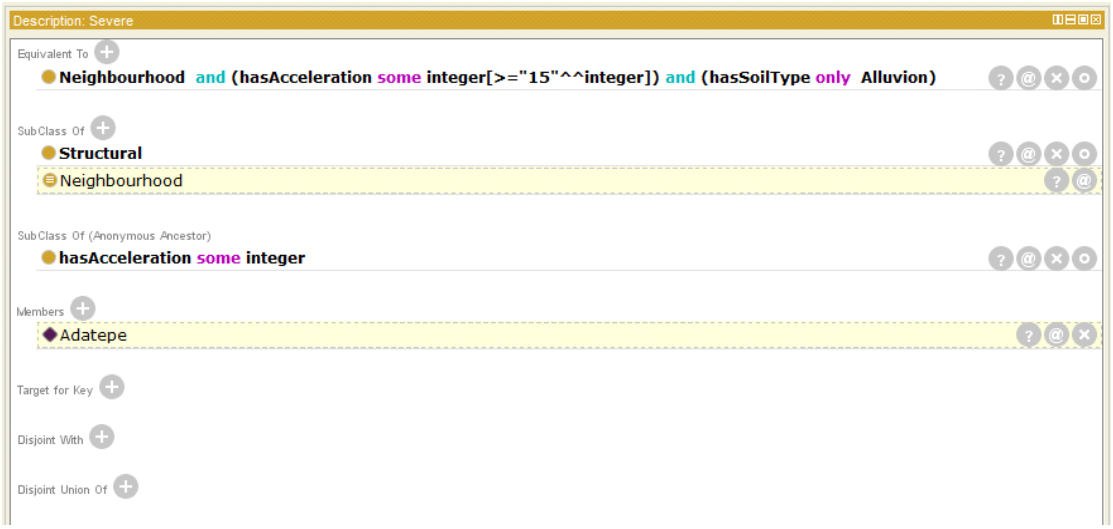


Figure 3.21 Severe neighbourhood restrictions

Broken bridges are defined where neighbourhood damage is severe with "*If neighbourhood damage is severe bridges can be break down*" rule 3. *Bridge* and *BrokenBridge* class are created for bridge status. "*BrokenBridge hasNeighbourhood some severe rule"* is determined to inference bridges named *Bridge1* under *BrokenBridge* class (Figure 3.22). Bridge length, load bearing capacity can be added into rule if exist. For example some bridges could be closed thus other bridges are over capacity with lots of traffic. These bridges load capacity have to be calculated with DL rules. New routes are calculated with this inference information.

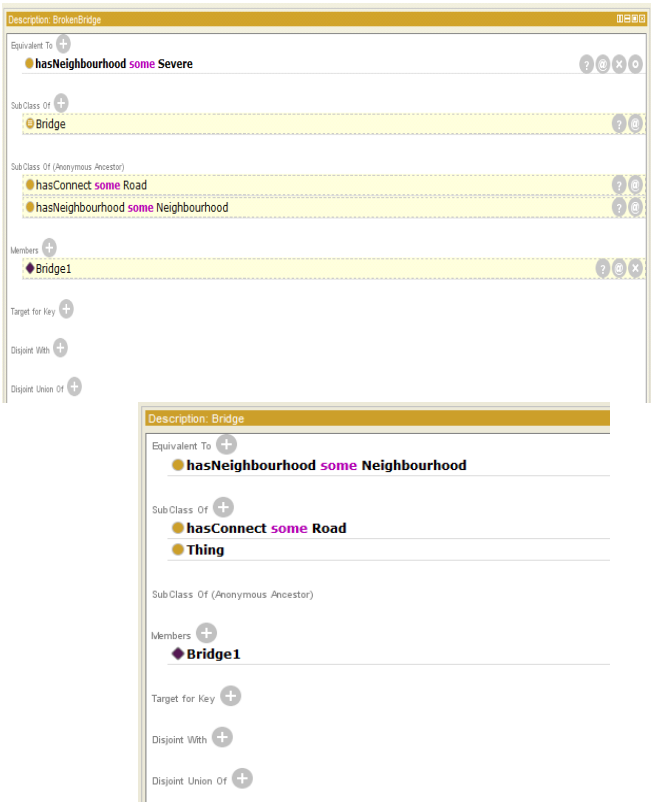


Figure 3.22 Bridge and broken bridge class view

Things are a bit more complicated with assessing closed roads. "*If neighbourhood damage is severe and Building height is more than 5 roads can be closed?"* rule 4 is used for defining closed roads according to building height and damage situation of roads. First we provide to define which buildings can be destroyed and we create *'Building and hasHeight some integer [>="5"^^integer]'* rule belonging to BrokenBuilding class. After building, road and neighbourhood relations are defined each other. Finally closed road class is created under road class and Road1 is inferred under the *closedRoad* class by using *"Road and (hasNeighbourhood some Severe) and (hasConnect some BrokenBuilding)"* rule (Figure 3.23).

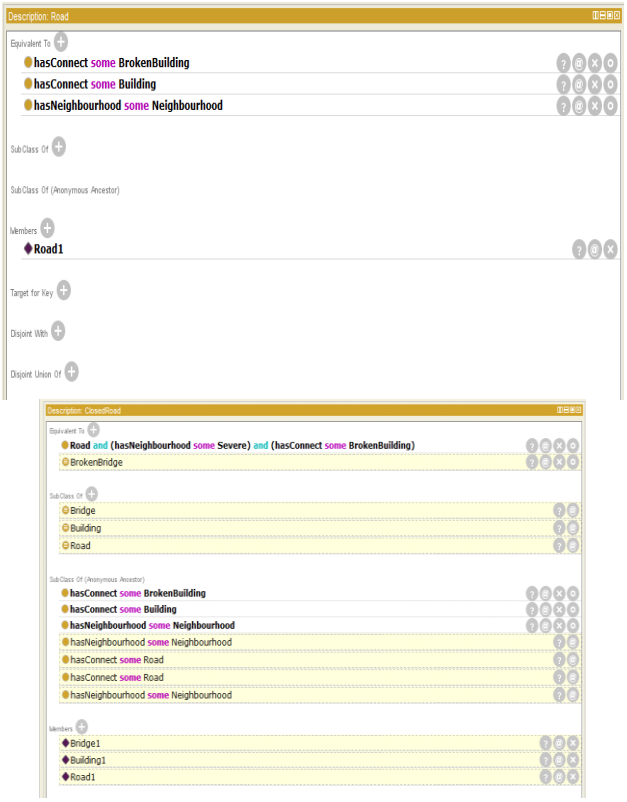


Figure 3.23 Road and closed road class view

TentCity class is defined and *"If neighbourhood damage is severe and population greater than 5000 built tent city?"* rule 5 for this class. Relation between *tentCity* and *neighbourhood* and bound *hasPop* data property is defined on neighbourhood; with this property we get population of neighbourhood. Finally "*Neighbourhood and (hasNeighbourhood some Severe) and (hasPop some integer [>="750"^^integer])* "rule is defined for tent city requirements which if population of neighbourhood is greater than 750 and damage is severe for should be set up tentcity on neighbourhood (Figure 3.24).

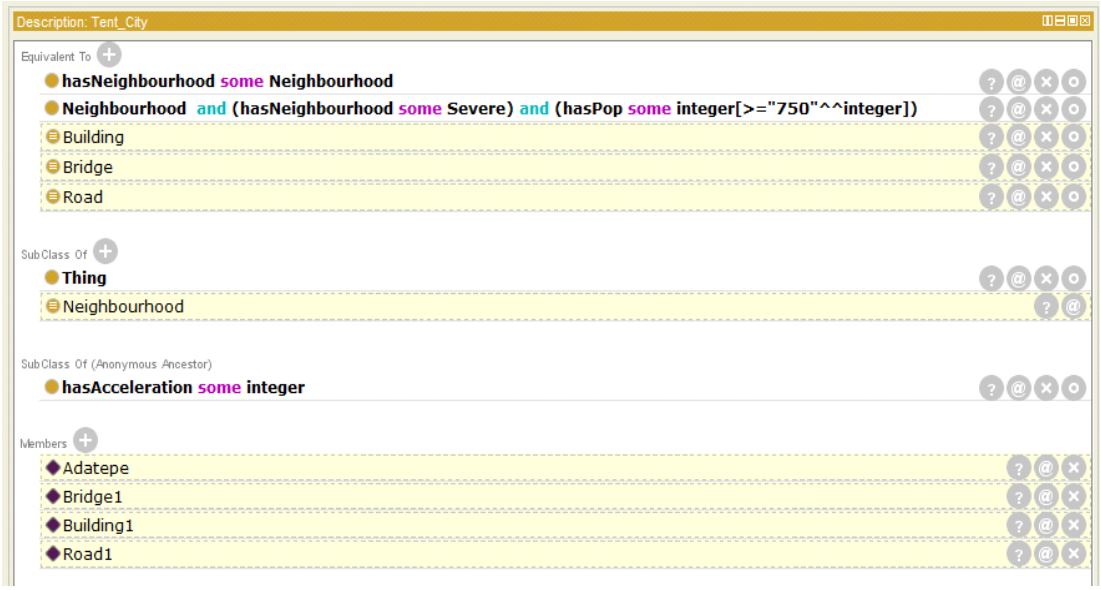


Figure 3.24 Tent city requirement rule for neighborhood class

Finally individuals inserted with java application to the relevant classes. Briefly spatial and non-spatial objects are inserted to the ontology as an XML format named OWL with Protégé.

CHAPTER FOUR
SPATIAL DATA INTEGRATION WITH DISASTER ONTOLOGY

Within addressing the use case elaborated in the motivating overview of Chapter 1, disaster ontology is developed for disaster management domain by applying methods in the Chapter 3. Three challenges are indicated in research questions based on these challenges, respectively spatial data integration into ontology from proprietary file formats, earthquake ontology creation with Protégé and finally earthquake damage assessment are performing with DL rules in this chapter. To accomplish spatial data integration challenge; first FME technology is used for converting data sources into GML. Spatial data annotation is performed with Java programming language on Eclipse platform. Eclipse is an integrated development environment for developer which is easy to develop java application and compatible with Jena Library. Java Server Pages with Eclipse Java platform are used for web site development. Jena is used to develop applications that use the semantic Web information model and languages (McBrian, 2002). Apache Jena is a Java framework for building Semantic Web applications. Jena includes rule based reasoning mechanism and provides programming environment for RDF, RDFS, OWL and SPARQL.

Disaster management needs to access multiple heterogeneous data sources. These data sources may come from distributed data sources. All distributed data sources must be linked and integrated to be processed. Data sources can be heterogeneous in syntactic, schematic and semantic, thus making data interoperation a difficult task (Bishr Y., 1998). Syntactic heterogeneity is caused by the use of different data models. For example different file formats, access protocols etc. for data sources. Schematic heteregonity means structural differences on data model. For example each institution represents and stores same data in different ways. Semantic heterogeneity is caused by different meaning or interpretations of data in various contexts. For example consider two XML sources, one in peer p1 and the other in peer p2, and a global ontology expressed in OWL. As shown in (Figure 4.1), the

global ontology consists of a class publication and two subclasses paper and book. The publication class is mapped to the publication element of the XML source in *"The Semantic web"*, while the book class corresponds to book of the XML source in *"Pinhan"*. An XML query Q1 on p1 involving publication will be rewritten to a target query Q2 on p2 involving include book. The XML fragments inside the dashed-line boxes are integrated and returned as answers.

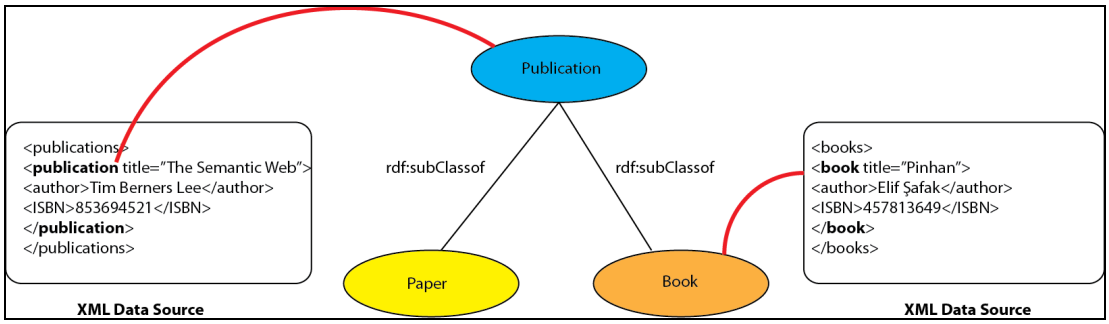


Figure 4.1 Data integration into ontology from XML data sources

However, schematic data heterogeneity may persist, depending on the XML schemas used (e.g., nesting hierarchies). Ontological approaches now offer new solution avenues to semantic interoperability limitation. Common uses of ontologies are data standardization and conceptualization via a formal machine-understandable ontology language. Data integration between data sources can be made by using semantic web technologies. Semantic integration is the process of matching/merging/fusing individual data/schemas in order to provide a global and integrated (unified) view. Several semantic data integration approaches exist in literature. Classification of ontology-based integration approaches fall into following three main categories:

1. **Single ontology approach**. All data sources are directly related to a shared global ontology. All concepts are defined in same ontology which were conceptualized with domain experts. This ontology can also be a combination of several specialized ontologies.

2. **Multiple ontology approach**. Each data source is represented by its own local ontology. Mapping between the ontologies has to be established. There is no need to integrate data in a global ontology. Besides, changes to local ontologies may not affect the mapping.

3. **Hybrid ontology approach**. Data in each source are represented by a local ontology and a shared vocabulary is built for sharing vocabularies among local ontologies. All local ontology is connected to global ontology.

Single and hybrid ontology approaches are used in central data integration systems. These approaches are more suitable for Global as View (GaV) systems which mean every entity in the global schema is associated with local schema. Multiple ontology approach can be most used Local as View (LaV) systems. Local schema can be processed without affecting global schema in the LaV system.

Ontology integration with spatial data sources have been initialled by Hiramatsu and Reitsma by using ESRI ArcExplorer software. Firstly they converted geographic instances into Geographical Markup Language (GML) after then they used Extensible Stylesheet Language (XSLT) to transform into OWL (Hiramatsu et al., 2004) (Raskin et al., 2005).

Other studies on literature, Nikolaos et al. (2005) converted spatial web service into RDF language and then attached into ontology with XSLT language. Di-Maron et al., (2007) attached spatial data into OWL using XSLT with i-annotate plugin from web pages hold spatial data. Lutz & Kolas (2007) realized ontology integration from web services with using schema mapping between spatial data sources. Rules are determined SWRL languages between local schema and domain ontology mapping. Several studies are implemented for mapping spatial data into ontology on literature. But in these studies, data sources are spatial web service which has pro rota metadata information. For this reason integration process developed around converting metadata into ontology.

While investigating our disaster management structure, any data providers in disaster management can publish their data via web services. In this case integration of data sources is examined lack of metadata in proprietary format rather than integration of web services into ontology. In this framework there is a single global ontology and several local data sources. For this reason, data integration approach is single ontology approach in GaV view.

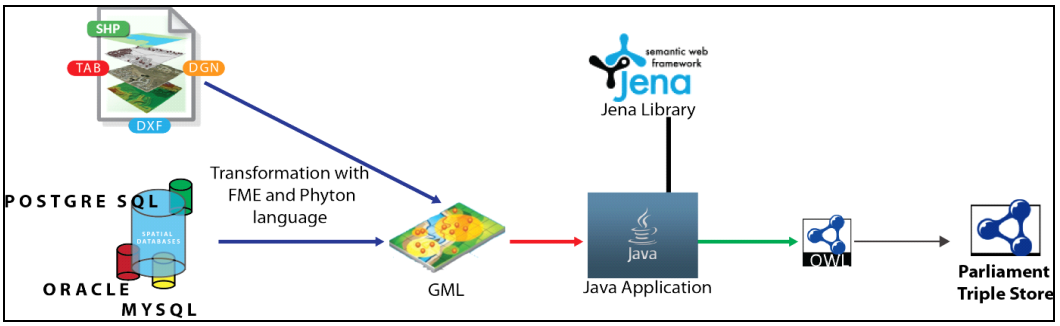


Figure 4.2 Spatial data integration methodology

4.1 Spatial Data Transformation

First data source transformation into same standard format exist different methods for providing syntactic heterogeneity between data sources (Figure 4.2). Feature Manipulation Engine (FME) method techniques and manual programming techniques are used with pre-defined algorithm for this operation. FME has been introduced as a complete interoperability solution and eliminates barriers of spatial data integration and transform and enhances data-sharing capabilities between diverse applications. The other method is transformation via python programming language. Spatial data in different formats are transformed into GML format with using python programming language technique and spatial data attached into ontology with generated java application. But using FME technology is more effective than the other application for several reasons. One of the major reason is that FME has great format choice for transformation data sources and FME has a server version for online transformation. Finally FME server can make transformation process on duty choice. This means there is always transformed data

sources from institutions. This operation is made by using FME technology to retrieve data from spatial databases and files in standard format.

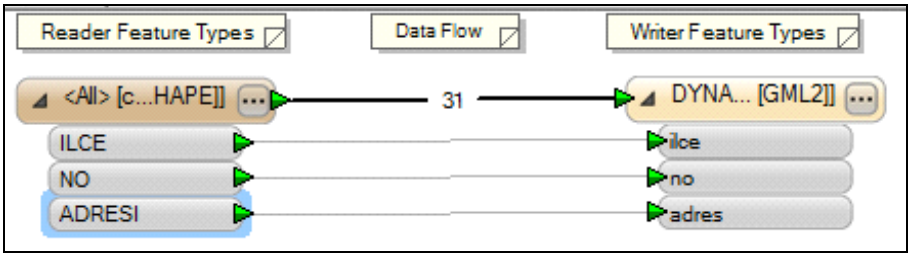


Figure 4.3 FME spatial data transformation method

Therefore it is possible to convert and use maps into common GML language, no matter in which format. For example meeting place shape file is converted into GML format with all column structure and data structure together (Figure 4.3). FME enabling column selection for file provides an opportunity for selecting only column with covered into ontology. Consequently syntactic and schematic heterogeneity is provided by using FME technology between data sources. For this reason it's need to be read GML file for integration with java I/O library in application after converting data sources into GML format. GML language is a standard language which makes it work easier because <*gml:featureMember*> tag is used regardless of data type. Coordinates and attribute data about all objects can be read from gml file by using this standard tag (Figure 4.4).

```
<?xml version="1.0" encoding="utf-8" ?>
<ogr:FeatureCollection
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:schemaLocation="http://ogr.maptools.org/ ilkToplanma.xsd"
        xmlns:ogr="http://ogr.maptools.org/"
        xmlns:gml="http://www.opengis.net/gml">
<gml:boundedBy>
        <gml:Box>
        <gml:coord><gml:X>-0.9727200000000003</gml:X><gml:Y>-0.03527999999999998</gml:Y></gml:coord>
        <gml:coord><gml:X>-0.8383200000000002</gml:X><gml:Y>0.28055999999999999</gml:Y></gml:coord>
        </gml:Box>
</gml:boundedBy>
<gml:featureMember>
        <ogr:City fid="İzmir">
<ogr:geometryProperty><gml:Point><gml:coordinates>-0.97272,0.28056</gml:coordinates></gml:Point></ogr:geometryProperty>
        <ogr:id>1</ogr:id>
        </ogr:ilkToplanma>
</gml:featureMember>
<gml:featureMember>
        <ogr:City fid="Aydın">
<ogr:geometryProperty><gml:Point><gml:coordinates>-0.83832,-0.03528</gml:coordinates></gml:Point></ogr:geometryProperty>
        <ogr:id>2</ogr:id>
        </ogr:ilkToplanma>
</gml:featureMember>
</ogr:FeatureCollection>
```

Figure 4.4 Geographical markup language structure

Thereafter semantic heterogeneity is provided by using ontologies. Spatial data integration into ontology is required after converting data sources into standard GML language. First of all there is spatial and non-spatial data existing in data sources. For example there are neighborhood name and coordinates as polygon in neighborhood GML file. Each of the spatial and non-spatial data should be integrated with ontology. First GML file is read then attached these concepts into ontology with Jena library and then exported into triple store in our ontology integration methodology. But there are some problems existing while integrating spatial data into ontology and exporting triple store. Triple store cannot have spatial index because of attaching process is made with Jena library. Spatial data should be inserted with GeoSPARQL statement via Parliament triple store library. For this reason the spatial data integration methodologies are separated into two phase; 1) non-spatial data integration with Jena library 2) spatial data integration with Parliament triple store library.

4.2 Non-spatial Data Integration

After transformation of spatial data into GML, non-spatial data will attach from GML into OWL ontology with java programming capabilities through the Jena library on Eclipse java platform. First, we read GML file with I/O library (Figure 4.5) and then disaster ontology read GML file with Jena library in java application.

```
//Reading Neighbourhood GML //
try
{

File file=new File("C:\\Users\\Can\\Desktop\\DisasterOntologyEnglish\\geoData\\gml\\Mahalle.gml");
DocumentBuilderFactory dbf = DocumentBuilderFactory.newInstance();
DocumentBuilder db=dbf.newDocumentBuilder();
Document doc= db.parse(file);
doc.getDocumentElement().normalize();
nodeLst = doc.getElementsByTagName("gml:featureMember");

for(int s=0;s<nodeLst.getLength();s++)
{
Node fstNode=nodeLst.item(s);
Element fstElmnt = (Element) fstNode;
NodeList fstNmElmntLst = fstElmnt.getElementsByTagName("fme:mahalle_adi");
Element fstNmElmnt = (Element) fstNmElmntLst.item(0);
NodeList fstNm = fstNmElmnt.getChildNodes();
mahalle=((Node) fstNm.item(0)).getNodeValue();
ary.add(mahalle);

}
}
catch(Exception e)
{
e.printStackTrace();
}

////////////////////////////////
```

Figure 4.5 Reading neighborhoods GML in java application

OntModel is created and "http://www.izmirdeafet.com/Disaster#" namespace is defined for preparing manipulating ontology with Jena library (Figure 4.6).

```
//DEFAULT ONTOLOGY MODEL
File girendosya = new File("C:\\Users\\Can\\Desktop\\DisasterOntologyEnglish\\Disaster.owl");
File cikandosya = new File("C:\\Users\\Can\\Desktop\\DisasterOntologyEnglish\\DisasterOut.owl");

String NS="http://www.izmirdeafet.com/Disaster.owl#";
FileInputStream inputdosya= new FileInputStream(girendosya);
FileOutputStream outputdosya= new FileOutputStream(cikandosya);

OntModel model= ModelFactory.createOntologyModel();
model.read(inputdosya,NS);
////////////////////////////////////////////////////////////////////////////////////////
```

Figure 4.6 Reading default ontology model in java application

Now neighbourhood individuals are attached at neihgbourhood name column from neighbourhood GML file to neighbourhood class which was created above after reading GML and ontology file with java application (Figure 4.7).

```
////////////////////////////////Neighbourhood Individual Insert//////////////////////////
for(int s=0;s<nodeLst.getLength();s++)
{
OntClass NeighbourhoodClass=model.getOntClass("http://www.izmirdeafet.com/Disaster.owl#Neighbourhood");
Individual indNeigh= model.createIndividual("http://www.izmirdeafet.com/Disaster.owl#"+ary.get(s),NeighbourhoodClass);
}

//////////////////////////////////////////////////////////////////////////////////////
```

Figure 4.7 Inserting individuals into neighborhood class in ontology

Id, magnitude, timeInterval, depth values are read for inserting as data type properties from GML file. After reading GML file these data are attached into array in java (Figure 4.8).

```
//Earthquake GML Dosyasi Okunuyor//
try
{
File file= new File("C:\\Users\\Can\\Desktop\\DisasterOntologyEnglish\\geoData\\gml\\earthquake.gml");
DocumentBuilderFactory dbf = DocumentBuilderFactory.newInstance();
DocumentBuilder db=dbf.newDocumentBuilder();
Document doc= db.parse(file);
doc.getDocumentElement().normalize();
nodeLstE = doc.getElementsByTagName("gml:featureMember");

for(int s=0;s<nodeLstE.getLength();s++)
{
Node fstNode=nodeLstE.item(s);

Element fstElmnt = (Element) fstNode;

NodeList fstNmElmntLst1 = fstElmnt.getElementsByTagName("fme:id");
NodeList fstNmElmntLst2 = fstElmnt.getElementsByTagName("fme:Magnitude");
NodeList fstNmElmntLst3 = fstElmnt.getElementsByTagName("fme:TimeInterv");
NodeList fstNmElmntLst4 = fstElmnt.getElementsByTagName("fme:Depth");

Element fstNmElmnt1 = (Element) fstNmElmntLst1.item(0);
Element fstNmElmnt2 = (Element) fstNmElmntLst2.item(0);
Element fstNmElmnt3 = (Element) fstNmElmntLst3.item(0);
Element fstNmElmnt4 = (Element) fstNmElmntLst4.item(0);

NodeList fstNm1 = fstNmElmnt1.getChildNodes();
NodeList fstNm2 = fstNmElmnt2.getChildNodes();
NodeList fstNm3 = fstNmElmnt3.getChildNodes();
NodeList fstNm4 = fstNmElmnt4.getChildNodes();

id=((Node) fstNm1.item(0)).getNodeValue();
magnitude=((Node) fstNm2.item(0)).getNodeValue();
time=((Node) fstNm3.item(0)).getNodeValue();
depth=((Node) fstNm4.item(0)).getNodeValue();
aryID.add(id);
aryMagnitude.add(magnitude);
aryTime.add(time);
aryDepth.add(depth);
}
}
catch(Exception e)
{
e.printStackTrace();
}
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
```

Figure 4.8 Reading values for data type properties from GML file

Data type properties of classes are inserted after individual's insertion on related classes. While inserting data type properties first resources are getting as classes and then defined data type property. Triples are creating with class name datatype property and value from GML file. For example *"Earthquake2, hasMagnitude, 6.5"* triple is used for inserting ontology (Figure 4.9).

```
//////////////////////////////Earthquake Individual Ekleme//////////////////////
for(int s=0;s<nodeLstE.getLength();s++)
{
Resource indM= model.getResource("http://www.izmirdeafet.com/Disaster.owl#"+aryID.get(s));
DatatypeProperty hasMagnitude= model.getDatatypeProperty("http://www.izmirdeafet.com/Disaster.owl#hasMagnitude");
DatatypeProperty hasTime= model.getDatatypeProperty("http://www.izmirdeafet.com/Disaster.owl#hasTime");
DatatypeProperty hasDepth = model.getDatatypeProperty("http://www.izmirdeafet.com/Disaster.owl#hasDepth");
model.add(indM,hasMagnitude,aryMagnitude.get(s));
model.add(indM,hasTime,aryTime.get(s));
model.add(indM,hasDepth,aryDepth.get(s));
}
////////////////////////////////////////////////////////////////////////////////
```

Figure 4.9 Inserting data type properties as triples on ontology

After completing this process, disaster ontology is inferred with reasoning mechanism and exported into Parliament triple store. It needs to be achieved inferenced results according to DL rules in ontology by using reasoning mechanism. Reasoning derives facts that are not expressed in ontology or in knowledge base explicitly. All of the restrictions and rules with description logic that were discussed in the sections above were created with the outlook of automatic inferencing from current ontology. Individuals are inferred and provide more efficient decisions for decision makers in the direction of the rules which defined with reasoning.

For example *earthquake2* which has 6.5 magnitude and 2 minutes time interval are defined as disaster and assigned as a new inference individual to this class which was empty before reasoning. Reasoning mechanisms such as FaCT++, HermiT, and Pellet are used for inferencing. Several reasoning mechanisms are revealed in literature for this reason all of the three reasoning mechanisms are examined and one of the mechanism is selected and used in this study. FaCT++ reasoning mechanism is the new generation of the well-known FaCT OWL-DL reasoner which was developed from Manchester University. It supports OWL DL and partially OWL 2 DL (Horrocks 2007). *HermiT* is the first publicly available OWL reasoner which provides more efficient reasoning than the other. HermiT uses direct semantics and passes all OWL 2 conformance tests for direct semantic reasoners. Finally *Pellet* is developed from Clark and Parsia Company. Pellet is an OWL 2 reasoner and provides standard and cutting-edge reasoning services for OWL ontologies. Pellet is chosen for reasoning mechanism because of the fully support OWL 2 DL reasoning capabilities. Inferenced values are required to achieve after selecting reasoning mechanism and integrating GML data into ontology. Several methods are

investigated for reasoning. Each method analyzes and determines specific advantages and disadvantages. Then all methods are tested and the suitable method is chosen for disaster ontology reasoning process.

4.2.1 Reasoning with Using Jena Library Reasoning Mechanism

First reasoning method is using Jena library in java application. Jena OWL reasoners could be described as an instance based reasoners. They work by using rules to propagate the if- and only-if implications of the OWL constructs on instance data. Beyond these basic choices, the complexities of configuring an ontology model are wrapped up in a recipe object called *OntModelSpec*. This specification allows complete control over the configuration choices for the ontology model, including the language profile in use, the reasoner, and the means of handling compound documents. A number of common recipes are pre-declared as constants in *OntModelSpec*, and listed below (Table 4.1).

Table 4.1 Ontology model specification and reasoning levels

OntModelSpec	**Language Profile**	**Storage Model**	**Reasoner**
OWL_MEM	OWL full	in-memory	none
OWL_MEM_TRANS_INF	OWL full	in-memory	transitive class-hierarchy inference
OWL_MEM_RULE_INF	OWL full	in-memory	rule-based reasoner with OWL rules
OWL_MEM_MICRO_RULE_INF	OWL full	in-memory	optimised rule-based reasoner with OWL rules
OWL_MEM_MINI_RULE_INF	OWL full	in-memory	rule-based reasoner with subset of OWL rules

Jena library is used in java application with this time for inferencing earthquakes as a disaster process. Firstly disaster ontology is chosen and created ontology model with "OWL_MEM_MINI_RULE_INF" ontology model specification. And then SPARQL query: *"Select * Where {?Disaster a dis:Disaster}"* is sent to triple store (Figure 4.10). But there is no result in *Disaster* class with this SPARQL query (Figure 4.11). This result shows that OWL-based reasoning process with Jena library cannot get sufficient enough expressivity by examining ontology model specification and reasoned types above.

```
protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException
{
    File girendosya = new File("C:\\Users\\Can\\Desktop\\DisasterOntologyEnglish\\DisasterOut2.owl");
    if(girendosya.exists())
    {
        String NS="http://www.izmirdeafet.com/Disaster.owl#";
        FileInputStream inputdosya= new FileInputStream(girendosya);
        OntModelSpec spec= new OntModelSpec(OntModelSpec.OWL_MEM_MINI_RULE_INF);
        spec.setReasoner(OntModelSpec.OWL_DL_MEM_RULE_INF.getReasoner());
        OntModel model= ModelFactory.createOntologyModel(spec);
        model.read(inputdosya, NS);
        String queryString=
                "PREFIX owl: <http://www.w3.org/2002/07/owl#>"+
                "PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>"+
                "PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>"+
                "PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>"+
                "PREFIX dis:<http://www.izmirdeafet.com/Disaster.owl#>"+
                "SELECT * WHERE{?Disaster a dis:Disaster}";
        com.hp.hpl.jena.query.Query query = QueryFactory.create(queryString);
        QueryExecution qe=QueryExecutionFactory.create(query,model);
        ResultSet results=qe.execSelect();
        ByteArrayOutputStream baos= new ByteArrayOutputStream();
        PrintStream ps= new PrintStream(baos);
        ResultSetFormatter.outputAsJSON(ps, results);
        String s= new String(baos.toByteArray(),"UTF-8");
        System.out.println(s);
        qe.close();
    }
    else
    {
        System.out.println("Dosya Yok!");
    }
```

Figure 4.10 Ontology model with OntModel spec

```
{
  "head": {
    "vars": [ "Disaster" ]
  } ,
  "results": {
    "bindings": [

    ]
  }
}
```

Figure 4.11 Basic ontology model SPARQL result

In this context, Pellet reasoning mechanism is used with Jena library for getting more expressivity from reasoning results. Pellet library is imported with this statement "*import org.mindswap.pellet.jena.PelletReasonerFactory"* and Jena library with this "*import com.hp.hpl.jena.ontology.OntModel"*. These statements are inserted on java application for available usage of Pellet reasoned (Figure 4.12). Therefore Pellet and Jena reasoning model results are compared with each other. Although SPARQL query gives result by using Pellet ontology model but other SPARQL query gives no result by using default ontology model about inferencing disaster class (Figure 4.13). SPARQL query statement is stored in different file with SPARQL extension in query with Pellet ontology model. This makes query reusable for another queries.

```
// PELLET ONTOLOGY MODEL
String  ontology= "file:C:\\Users\\Can\\Desktop\\DisasterOntologyEnglish\\Disaster.owl";
String  queries="file:C:\\Users\\Can\\Desktop\\DisasterOntologyEnglish\\query1.sparql";
String ont = "http://www.izmirdeafet.com/Disaster.owl";
String ns = "http://www.izmirdeafet.com/Disaster.owl#";
OntModel pelletModel = ModelFactory.createOntologyModel( PelletReasonerFactory.THE_SPEC );
pelletModel.read(ont);

//PELLET SPARQL
pelletModel.read( ontology );
Query q = QueryFactory.read( queries );
QueryExecution qe = SparqlDLExecutionFactory.create( q, pelletModel );
ResultSet rs = qe.execSelect();
System.out.println(q.toString());
ResultSetFormatter.out( rs );
System.out.println();
```

Figure 4.12 Pellet reasoner ontology model

```
PREFIX  rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX  dis:  <http://www.izmirdeafet.com/Disaster.owl#>
PREFIX  owl:  <http://www.w3.org/2002/07/owl#>
PREFIX  xsd:  <http://www.w3.org/2001/XMLSchema#>
PREFIX  rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

SELECT  *
WHERE
  { ?Disaster  rdf:type  dis:Disaster .}

-------------------------------------------------------------
| Disaster                                                  |
=============================================================
| <http://www.izmirdeafet.com/Disaster.owl#EarthquakeBig> |
-------------------------------------------------------------
```

Figure 4.13 Pellet reasoner SPARQL query result

As a result of the comparisons, Pellet reasoned is chosen for inferencing mechanism. Furthermore suitability with triple store is examined with this method because ontology stores and queries from triple store. But as is seen this method which can be inference with Jena and Pellet library cannot be queried over triple store. For this reason the other method is examined below which uses reasoning over Parliament triple Store.

4.2.2 Reasoning with Parliament Triple Store Reasoning Mechanism

Reasoning mechanism for inferencing is provided with Parliament triple store itself. Parliament originally designed very simple ontologies and not didn't put much effort towards more advanced inferences like intersections and restrictions. Parliament present rule engine, intersections would not be hard to implement. However restrictions are much more difficult, in part because there are several kinds

of restrictions, and also present rule engine can support them efficiently. Besides Parliament triple store inferencing setting should be changed for making reasoning over this. Finally inferencing over Parliament triple store gives no result for SPARQL query on disaster inference (Figure 4.14).

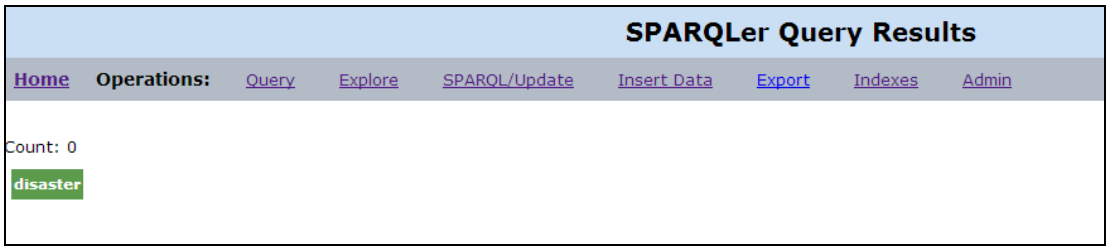


Figure 4.14 Inferenced SPARQL result

4.2.3 Reasoning with Using Reasoner Mechanism on Protégé

Finally reasoning process is made by using Protégé reasoning mechanism. Advantage of this method is providing more expressivity by using Pellet reasoner on Protégé. For example three individuals named as 1, 2, 3 are created under earthquake class (Figure 4.15). Earthquake named as 2 is inferred under the disaster class by using DL-based *Earthquake Π (> 6.5 hasMagnitude Π >1 hasTime)* $\sqsubseteq$ *Disaster rule* (Figure 4.16).

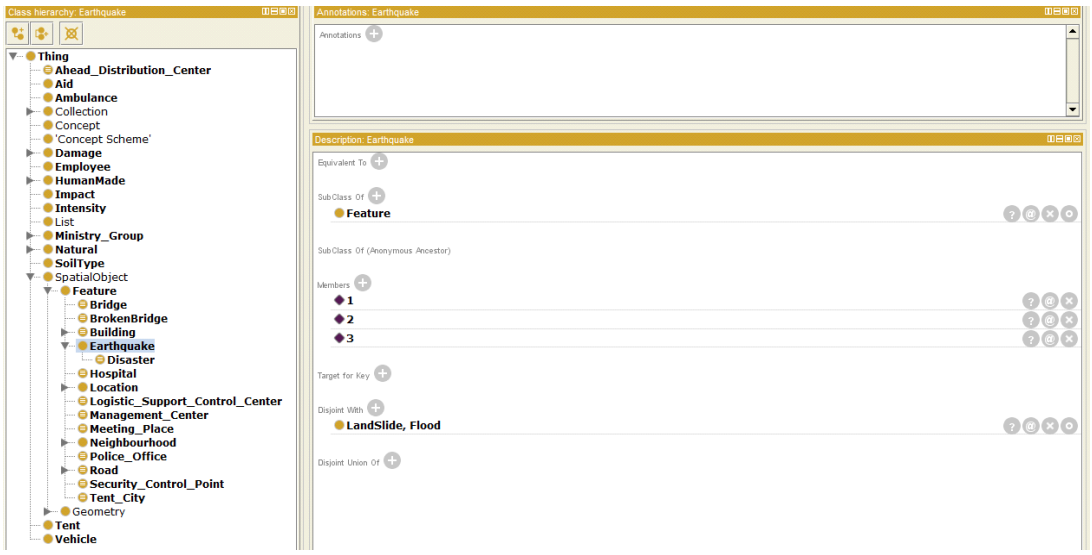


Figure 4.15 Earthquake class individuals view

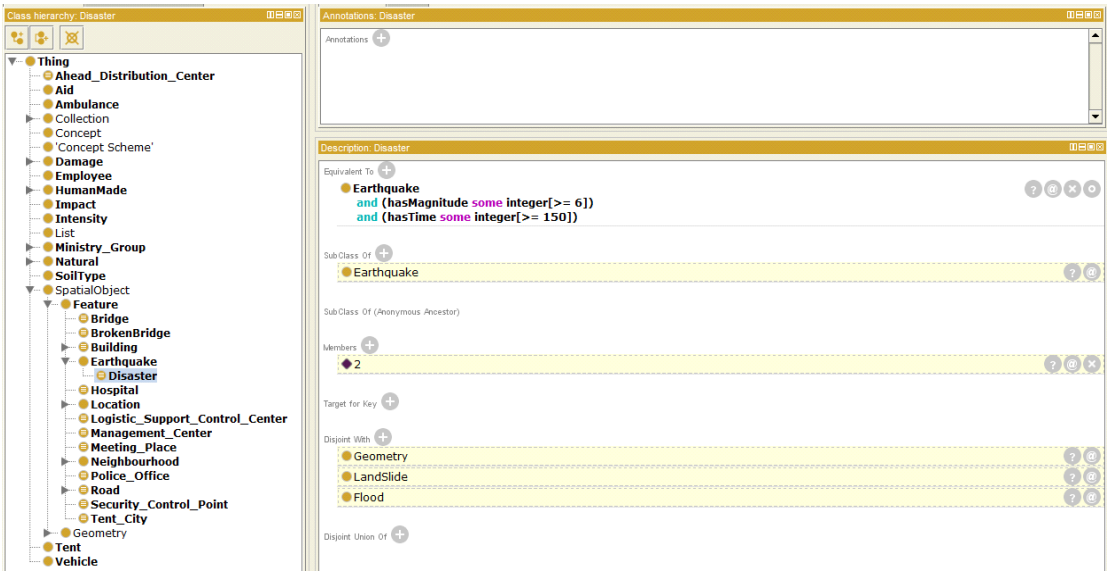


Figure 4.16 Disaster class inference individuals view

Results are inferred with this method in a most expressivity way under favor of Pellet reasoning mechanism. Despite everything this method requires some manual intervention during reasoning. This manual intervention interrupts automatization of all process. This purpose can be investigated in the future works.

Thus far non-spatial data is inserted into ontology and inference new individuals according to DL rules. But any spatial object is not inserted according spatial index problem in Parliament triple store. This also shows that spatial query cannot be done without spatial object. For this reason, spatial objects' different phases are inserted below.

4.3 Spatial Data Integration

Spatial data needs to insert into disaster ontology after completing non-spatial data inserting process. First GML file is read as same above and coordinates are extracted and inserted into ontology as triples with Parliament remote model. SPARQL query is used for INSERT statement (Figure 4.17). Triple consist;

"<http://www.izmirdeafet.com/Disaster.owl#AdatepeGeo>
<http://www.opengis.net/ont/geosparql#asWKT>

<http://www.opengis.net/def/crs/OGC/1.3/CRS84>”27.35 38.56, 27.68 38.66”
<http://www.opengis.net/ont/sf#wktLiteral>”

This means AdatepeGeo stores coordinate asWKT and coordinate system is WGS 84 and inserted into ontology in literal format.

```
   String geo = (String)slide.get("json_geometry");
   String id = (String)slide.get("mahalle_adi");
///PARLIAMENT TRIPLE STORE QUERY//////////////////////////
   String queryString=
           "PREFIX afn: <http://jena.hpl.hp.com/ARQ/function#>"+
           "PREFIX fn: <http://www.w3.org/2005/xpath-functions#>"+
           "PREFIX owl: <http://www.w3.org/2002/07/owl#>"+
           "PREFIX par: <http://parliament.semwebcentral.org/parliament#>"+
           "PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>"+
           "PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>"+
           "PREFIX time: <http://www.w3.org/2006/time#>"+
           "PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>"+
           "INSERT DATA {<http://www.izmirdeafet.com/Disaster.owl#"+id+"Geo> <http://www.opengis.net/ont/geosparql#asWKT> "+
           "'<http://www.opengis.net/def/crs/OGC/1.3/CRS84>"+geo+
           "'^^<http://www.opengis.net/ont/sf#wktLiteral> .}";
   RemoteModel rmParliament= new RemoteModel("http://localhost:38080/parliament/sparql","http://localhost:38080/parliament/bulk");
   rmParliament.updateQuery(queryString);
////////////////////////////////////////////////////////////
```

Figure 4.17 Inserting coordinates value with Parliament remote model as WKT

Inserted neighbourhood spatial objects are shown in Parliament triple store in (Figure 4.18). These triples are spatial indexed properly with Parliament triple store.

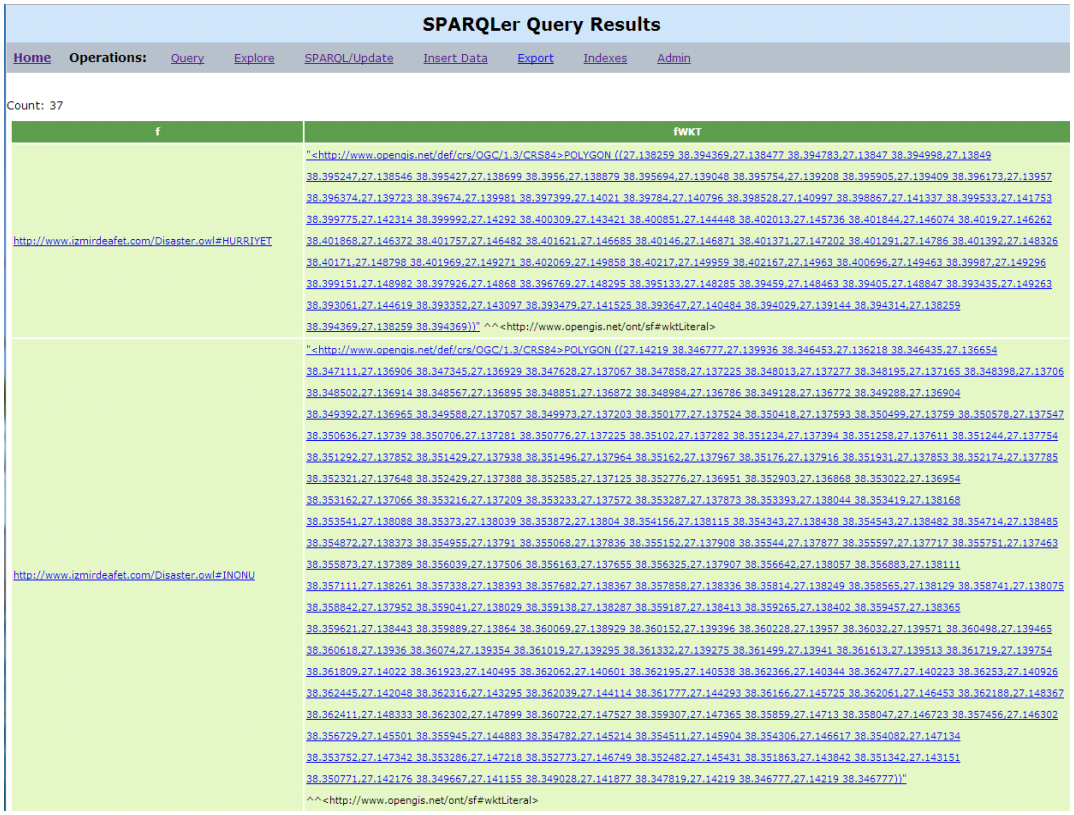


Figure 4.18 Neighborhood as WKT SPARQL result view

Finally spatial objects are added into ontology with spatial capabilities as latitude and longitude information in WKT format. These spatial data come from data sources dynamically with FME conversion. So reasoning and querying will be done with this integration method.

CHAPTER FIVE

EARTHQUAKE DAMAGE ASSESSMENT WITH DL RULES

Earthquake damage estimation within the first few hours of devastating earthquake is very substantial from the point of decision making and planning after disaster. Recently; with the development of earthquake engineering, earthquake damage estimation is determined by using strong ground motion network. However, several cities are densely populated under earthquake risk. Besides, they dont have strong ground motion network which are costly and needs advanced technological systems. This chapter demonstrates how to estimate earthquake damage at early post-disaster by only using data sources such as up-to-date earthquake source parameter, population, building quality, roads and bridge locations. Ontologies are used for ensuring semantic interoperability and DL rules are defined for damage estimation. Finally triple store is used for ontology storing and querying in this chapter.

5.1 Triple Stores

The Semantic Web employs a different data model than a relational database. A relational database stores data in tables while RDF represents data as directed graph of ordered triples of the form. Accordingly, a Semantic Web data store is often called a graph store, knowledge base, or triple store (Emmons, 2012). Directed graph can be stored in relational databases, but only straightforward want to store the graph is to use a single table to store all the triples. Thus query performance of such implementation is usually poor. Several triple stores exist in literature as Parliament triple store, Strabon, UseekM. All triple stores are analyzed and compared below.

5.1.1 Strabon

Development of strabon started in the context of European FP7 project SemsorGrid4Env (Semantic Sensor Grids for Rapid Application Development for Environmental Management). Spatial data usually store in RDF format and queried by SPARQL which extends spatial extension. Strabon presents a

new version of the data model as stRDF and the query language *stSPARQL* (Kyzirakos, 2013). Strabon both supports WKT and GML to represent spatial data as literals. *stSPARQL* is an extension of SPARQL. Strabon is built on top of the well-known RDF store Sesame. Strabon can work both PostgreSQL and proprietary DBMS. But Strabon is very new technologies rather than the other triple store (Figure 5.1).

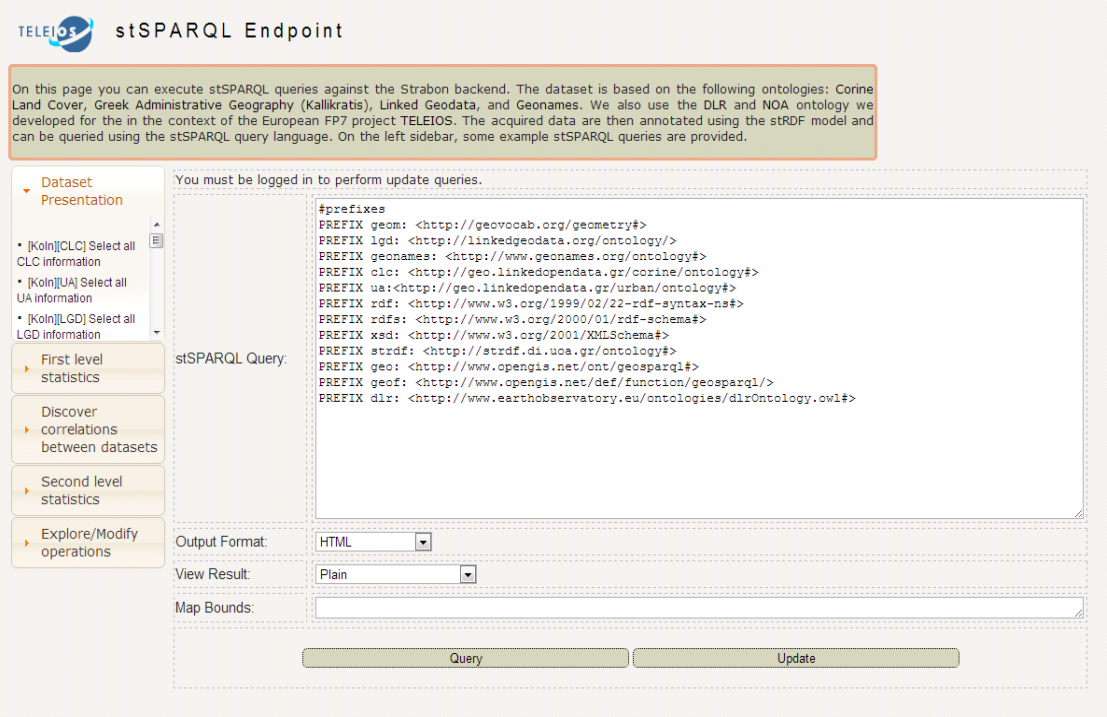


Figure 5.1 Strabon graphical user interface

5.1.2 USeekM

USeekM is library for semantic databases which use the Sesame Java interface. This adds indexing and querying capabilities to ontologies. This supports GeoSPARQL query languages as the other triple store. USeekM provides full text search capabilities like google keyword searching. Geospatial objects are only provided as WKT and this triple store cannot support CRS.

5.1.3 OWLIM

This is a semantic database which implements in Java and delivers through Sesame and Jena. It supports RDFS, OWL 2. OWLIM supports extended SPARQL language and W3C basic geo vocabulary.

5.1.4 Parliament Triple Store

Parliament is implemented a high-performance storage engine that is compatible with RDF and OWL standards from BBN Technologies. Parliament is not a complete database system at the same time Parliament query processor and includes rule engine which applies a set of inference rules to the directed graph of data in order to derive new facts. The core of Parliament is written in C++, but the integration of Jena is Java code. Jena graph for Parliament includes temporal and geospatial indexes. Parliament triple store combines Jena (a SPARQL query processor) and Jetty (a servlet engine) to create a complete knowledge base server application that implements a SPARQL endpoint. Open source based Berkeley database is used for storing triples at Parliament triple store.

Parliament triple store is chosen among several triple stores mentioned before in this study. Firstly Parliament triple store was installed on jetty servlet engine on the web server (Figure 5.2). Then specified ontology was exported into triple store. Joseki Parliament library is imported for SPARQL querying capabilities from triple store by using Jena library in java application.

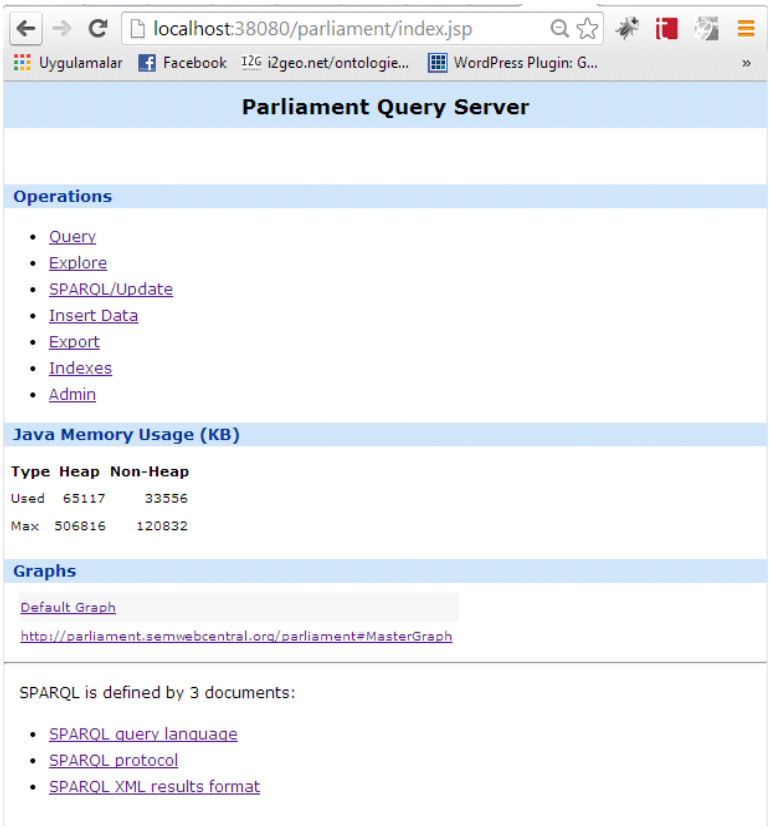


Figure 5.2 Parliament triple store GUI

5.2 Acceleration Calculation Methodology

Several models exist for damage assessment after an earthquake. Hu, Liu and Dong model is used in this study which depends on acceleration value to associate with distance from epicenter, magnitude of earthquake, basement type parameters state above. Horizontal acceleration equations are used for determining damage (Hu, Liu & Dong, 1997). Damage is calculated neighbourhood based by using acceleration formula and then this damage ratio is used in the other SPARQL queries. Acceleration formula needs such as magnitude, distance between epicenter and district region and basement type variable for calculation. Magnitude data is obtained from specific data sources as Kandilli Observatory but such as distance from epicenter, neighbourhood basement type data should be obtained with some calculation and queries. According to this scenario, all neighborhood ground acceleration values are defined for assessment damages with this algorithm (Figure 5.3).

Acceleration Calculation Algorithm

Input: *B* basement types, *M* magnitude of earthquake, *N* neighborhood, D neighborhood distance from epicenter, *cons1*, *cons2*, *cons3* from formula, *A* result

```
1: for i=0 to N size
2:   if B equals 'rock' do
3:       M=M*cons2
4:       D=D+10
5:       A=cons1*Math.pow(10,M)*Math.pow(D,cons3)
6:       end
7: AssignNeighbourhoodResult(N,A)
8: return
```

Figure 5.3 Acceleration calculation algorithm

5.3 GeoSPARQL Query Operations

Some variables are obtained with spatial query and incorporated into acceleration calculation according to algorithm. In this context neighbourhood distance from epicenter and neighbourhood basement type should be defined respectively.

5.3.1 Neighborhood Distances from the Main Epicenter Query

While calculating the distance from epicenter of neighbourhood, *geof:distance()* method is used for spatial query (Figure 5.4). Distance function has the same characteristics with spatial function which is used in GIS software calculate distances between two objects. First SPARQL query; *"Select ?f ?distance WHERE{ dis:2 geo:hasGeometry ?cGeom. ?cGeom geo:asWKT. ?f a dis:Neighborhood; geo:hasGeometry ?fGeom. ?fGeom geo:asWKT ?fWKT. BIND (geof:distance (?cWKT,?fWKT,uom:metre) AS? distance)}"* is defined and sent into triple store with parliament remote model. Earthquake as '*dis:2'* and neighborhood as *'dis:Neighborhood'* are determined as variable in query (Figure 5.4). Determination of these variables is not enough for spatial process, for this reason it is defined with '*geo:asWKT'* format. Therefore *'?cWKT'* and *'?fWKT'* variables exist for distance

functions. Distance type is used as meter with *'uom:metre'* format with '*UOM*' library at http://www.opengis.net/def/uom/OGC/1.0. We convert query results as XML file format and then results are read from XML file and send results to acceleration formula.

```
static void distance() throws IOException
{

//PARLIAMENT TRIPLE STORE QUERY
        String queryString=
                "PREFIX afn: <http://jena.hpl.hp.com/ARQ/function#>"+
                "PREFIX fn: <http://www.w3.org/2005/xpath-functions#>"+
                "PREFIX owl: <http://www.w3.org/2002/07/owl#>"+
                "PREFIX par: <http://parliament.semwebcentral.org/parliament#>"+
                "PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>"+
                "PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>"+
                "PREFIX time: <http://www.w3.org/2006/time#>"+
                "PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>"+
                "PREFIX geo: <http://www.opengis.net/ont/geosparql#>"+
                "PREFIX geof: <http://www.opengis.net/def/function/geosparql/>"+
                "PREFIX sf: <http://www.opengis.net/ont/sf#>"+
                "PREFIX dis:<http://www.izmirdeafet.com/Disaster.owl#>"+
                "PREFIX uom: <http://www.opengis.net/def/uom/OGC/1.0/>"+
                "SELECT ?f ?distance WHERE { dis:2 geo:hasGeometry ?cGeom . ?cGeom geo:asWKT ?cWKT . ?f a dis:Neighbourhood; "+
                "geo:hasGeometry ?fGeom . ?fGeom geo:asWKT ?fWKT . BIND(geof:distance(?cWKT,?fWKT,uom:metre) AS ?distance).} ";
        RemoteModel rmParliament= new RemoteModel("http://localhost:38080/parliament/sparql","http://localhost:38080/parliament/bulk");
        ResultSet results = rmParliament.selectQuery(queryString);
        ByteArrayOutputStream baos = new ByteArrayOutputStream();
        PrintStream ps = new PrintStream(baos);
        ResultSetFormatter.outputAsXML(ps,results);
        String s = new String(baos.toByteArray(), "UTF-8");
        PrintWriter out = new PrintWriter("C:\\Users\\Can\\Desktop\\DisasterOntologyEnglish\\geoData\\gml\\query.xml");
        out.write(s);
        out.close();
        try
        {

        File file=new File("C:\\Users\\Can\\Desktop\\DisasterOntologyEnglish\\geoData\\gml\\query.xml");
        DocumentBuilderFactory dbf = DocumentBuilderFactory.newInstance();
        DocumentBuilder db=dbf.newDocumentBuilder();
        Document doc= db.parse(file);
        doc.getDocumentElement().normalize();
        nodeLstQD = doc.getElementsByTagName("result");

        for(int i=0;i<nodeLstQD.getLength();i++)
        {
            Element question = (Element) nodeLstQD.item(i);
            NodeList optionList = question.getElementsByTagName("uri");
            for (int j = 0; j < optionList.getLength(); ++j)
            {
                Element option = (Element) optionList.item(j);
                querySD = option.getFirstChild().getNodeValue();
                queryD.add(querySD);
            }
            NodeList optionList2 = question.getElementsByTagName("literal");
            for (int j = 0; j < optionList2.getLength(); ++j)
            {
                Element option = (Element) optionList2.item(j);
                querySD = option.getFirstChild().getNodeValue();
                queryD.add(querySD);
            }
        }
        }
        catch(Exception e)
        {
        e.printStackTrace();
        }
```

Figure 5.4 Neighborhood distance from epicenter calculation

The same query is made with Pitney Bowes Mapinfo proprietary GIS software for proving and comparing results with GeoSPARQL query. Results are so closed each other and this also shows that distance function can be used in a standard way in this process (Figure 5.5, 5.6).

Count: 37

f	distance
http://www.izmirdeafet.com/Disaster.owl#HURRIYET	"3373.9707737399654e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#INONU	"5975.1253874611176e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#MUSTAFAKEMAL	"4901.1942391422963e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#CAMLIPINAR	"4413.5542813677834e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#CAMLIKULE	"3585.1897521938795e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#YESILBAGLAR	"4436.698924222251e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#GOKSU	"5298.857904089552e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#SEYHAN	"4644.333995005387e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#FIRAT	"3569.503571494168e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#YENIGUN	"3167.0363718686835e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#YILDIZ	"3302.212297452891e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#BUCAKOOP	"3998.9282377045561e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#KURUCESME	"3291.215993103842e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#ATATURK	"2847.934964993937e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#ADATEPE	"1299.062137617818e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#CUMHURIYET	"1043.4727369040345e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#KARANFIL	"1192.6853649777943e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#DUMLUPINAR	"2306.6090638060195e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#KOZAGAC	"2854.761841413898e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#AKINCILAR	"3778.050650520062e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#EFELER	"3090.607821837593e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#INKILAP	"3578.4721976692576e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#CAGDAS	"741.9707602896771e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#MURATHAN	"530.8337619296647e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#IZKENT	"499.742096662057e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#GAZILER	"1045.6715708692952e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#YAYLACIK	"1546.2458880438467e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#MENDERES	"1959.8108195094314e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#VALIRAHMIBEY	"2392.814395637769e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#BARIS	"2225.6278592981807e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#DICLE	"2272.6919395509476e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#GUVEN	"3299.288710264766e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#YIGITLER	"2722.1000770020596e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#LALELI	"3090.305933367266e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#CAMLIK	"2344.2704701602042e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#UFUK	"1513.9030342337064e0" ^^<http://www.w3.org/2001/XMLSchema#double>
http://www.izmirdeafet.com/Disaster.owl#SIRINKAPI	"428.069492086787e0" ^^<http://www.w3.org/2001/XMLSchema#double>

Figure 5.5 GeoSPARQL results of distance function between epicenter and neighbourhood

DistanceQuery Browser

Origin	Destination	Distance_Result	Distance_Unit
SIRINKAPI	2	428,333131	meters
UFUK	2	1.511,342434	meters
CAMLIK	2	2.339,859376	meters
LALELI	2	3.084,430143	meters
YIGITLER	2	2.717,462516	meters
GUVEN	2	3.294,096032	meters
DICLE	2	2.268,840700	meters
BARIS	2	2.222,788334	meters
VALIRAHMIBEY	2	2.394,957999	meters
MENDERES	2	1.958,078645	meters
YAYLACIK	2	1.546,650232	meters
GAZILER	2	1.047,338058	meters
IZKENT	2	500,384680	meters
MURATHAN	2	531,366272	meters
CAGDAS	2	742,980489	meters
INKILAP	2	3.574,086031	meters
EFELER	2	3.090,211030	meters
AKINCILAR	2	3.774,647874	meters
KOZAGAC	2	2.859,236474	meters
DUMLUPINAR	2	2.309,200328	meters
KARANFIL	2	1.195,183244	meters
CUMHURIYET	2	1.044,319385	meters
ADATEPE	2	1.299,733589	meters
ATATURK	2	2.853,000655	meters
KURUCESME	2	3.298,160148	meters
BUCAKOOP	2	4.007,407927	meters
YILDIZ	2	3.307,792627	meters
YENIGUN	2	3.169,336794	meters
FIRAT	2	3.570,744620	meters
SEYHAN	2	4.642,125135	meters
GOKSU	2	5.299,368685	meters
YESILBAGLAR	2	4.440,262118	meters
CAMLIKULE	2	3.589,905408	meters
CAMLIPINAR	2	4.418,420758	meters
MUSTAFAKEMAL	2	4.906,480312	meters
INONU	2	5.979,398766	meters
HURRIYET	2	3.367,381440	meters

Figure 5.6 MapInfo results of distance function between epicenter and neighbourhood

5.3.1.1 Basement Type Query

Other query is neighborhood basement type in Buca district for using the calculation of earthquake acceleration. For this query GeoSPARQL *'sf:intersect()'* filter function is used for providing neighborhood basement type. This result is saved as XML file like neighbourhood distance from epicenter result and used as variable in ground acceleration formula (Figure 5.7).

```
static void  addSoilNeighbourhoodQuery() throws IOException
{
    //PARLIAMENT TRIPLE STORE QUERY
    String queryString=
            "PREFIX afn: <http://jena.hpl.hp.com/ARQ/function#>"+
            "PREFIX fn: <http://www.w3.org/2005/xpath-functions#>"+
            "PREFIX owl: <http://www.w3.org/2002/07/owl#>"+
            "PREFIX par: <http://parliament.semwebcentral.org/parliament#>"+
            "PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>"+
            "PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>"+
            "PREFIX time: <http://www.w3.org/2006/time#>"+
            "PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>"+
            "PREFIX geo: <http://www.opengis.net/ont/geosparql#>"+
            "PREFIX geof: <http://www.opengis.net/def/function/geosparql/>"+
            "PREFIX sf: <http://www.opengis.net/ont/sf#>"+
            "PREFIX dis:<http://www.izmirdeafet.com/Disaster.owl#>"+
            "SELECT ?f ?s WHERE {?f a dis:Neighbourhood; geo:hasGeometry ?fGeom.?fGeom geo:asWKT ?fWKT.?s a dis:SoilType; "
            + "geo:hasGeometry ?gGeom.?gGeom geo:asWKT ?gWKT.FILTER(geof:sfIntersects(?fWKT,?gWKT))} ";
    RemoteModel rmParliament= new RemoteModel("http://localhost:38080/parliament/sparql","http://localhost:38080/parliament/bulk");
    ResultSet results = rmParliament.selectQuery(queryString);
    ByteArrayOutputStream baos = new ByteArrayOutputStream();
    PrintStream ps = new PrintStream(baos);
    ResultSetFormatter.outputAsXML(ps, results);
    String s = new String(baos.toByteArray(), "UTF-8");
    PrintWriter out = new PrintWriter("C:\\Users\\Can\\Desktop\\DisasterOntologyEnglish\\geoData\\gml\\query2.xml");
    out.write(s);
    out.close();

    try
    {

    File file=new File("C:\\Users\\Can\\Desktop\\DisasterOntologyEnglish\\geoData\\gml\\query2.xml");
    DocumentBuilderFactory dbf = DocumentBuilderFactory.newInstance();
    DocumentBuilder db=dbf.newDocumentBuilder();
    Document doc= db.parse(file);
    doc.getDocumentElement().normalize();
    nodeLstQR = doc.getElementsByTagName("result");
    for(int i=0;i<nodeLstQR.getLength();i++)
    {
        Element question = (Element) nodeLstQR.item(i);
        NodeList optionList = question.getElementsByTagName("uri");
        for (int j = 0; j < optionList.getLength(); ++j)
        {
            Element option = (Element) optionList.item(j);
            querySR = option.getFirstChild().getNodeValue();
            queryR.add(querySR);
        }
    }
```

Figure 5.7 Basement type selection of neighborhood with java application

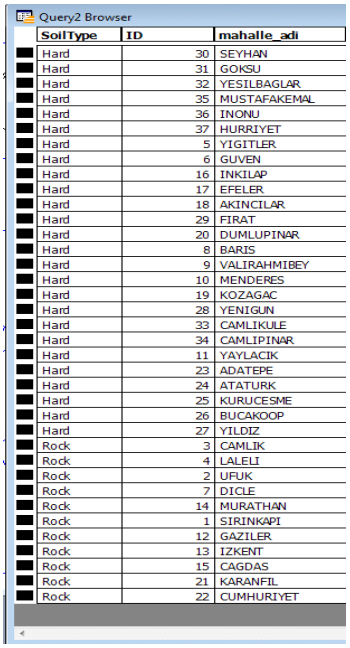
Basement type information for the province was derived from the 1:25,000 scaled digital geologic formation maps of the province provided by General Directorate of Minerals and Exploration (MTA). All formation was assigned one of the four basement types *("Rock", "Hard", "Medium", "Soft")*. Rock formations represent massive volcanic rocks which are the most settlement suitable. The next one *"Hard"*

represents slightly deformed volcanic rocks, which is also settlement suitable. *"Medium"* represents altered volcanic and metamorphic rocks which restricted suitability for settlement and the *"Soft"* represents thick alluvial deposits, which is the least suitable basement type for settlement. All these formations are shown in (Figure 5.8 – 5.9) (Alparslan, 2008).

Count: 37

f	s
http://www.izmirdeafet.com/Disaster.owl#HURRIYET	http://www.izmirdeafet.com/Disaster.owl#Hard
http://www.izmirdeafet.com/Disaster.owl#INONU	http://www.izmirdeafet.com/Disaster.owl#Hard
http://www.izmirdeafet.com/Disaster.owl#MUSTAFAKEMAL	http://www.izmirdeafet.com/Disaster.owl#Hard
http://www.izmirdeafet.com/Disaster.owl#CAMLIPINAR	http://www.izmirdeafet.com/Disaster.owl#Hard
http://www.izmirdeafet.com/Disaster.owl#CAMLIKULE	http://www.izmirdeafet.com/Disaster.owl#Hard
http://www.izmirdeafet.com/Disaster.owl#YESILBAGLAR	http://www.izmirdeafet.com/Disaster.owl#Hard
http://www.izmirdeafet.com/Disaster.owl#GOKSU	http://www.izmirdeafet.com/Disaster.owl#Hard
http://www.izmirdeafet.com/Disaster.owl#SEYHAN	http://www.izmirdeafet.com/Disaster.owl#Hard
http://www.izmirdeafet.com/Disaster.owl#FIRAT	http://www.izmirdeafet.com/Disaster.owl#Hard
http://www.izmirdeafet.com/Disaster.owl#YENIGUN	http://www.izmirdeafet.com/Disaster.owl#Hard
http://www.izmirdeafet.com/Disaster.owl#YILDIZ	http://www.izmirdeafet.com/Disaster.owl#Hard
http://www.izmirdeafet.com/Disaster.owl#BUCAKOOP	http://www.izmirdeafet.com/Disaster.owl#Hard
http://www.izmirdeafet.com/Disaster.owl#KURUCESME	http://www.izmirdeafet.com/Disaster.owl#Hard
http://www.izmirdeafet.com/Disaster.owl#ATATURK	http://www.izmirdeafet.com/Disaster.owl#Hard
http://www.izmirdeafet.com/Disaster.owl#DUMLUPINAR	http://www.izmirdeafet.com/Disaster.owl#Hard
http://www.izmirdeafet.com/Disaster.owl#KOZAGAC	http://www.izmirdeafet.com/Disaster.owl#Hard
http://www.izmirdeafet.com/Disaster.owl#AKINCILAR	http://www.izmirdeafet.com/Disaster.owl#Hard
http://www.izmirdeafet.com/Disaster.owl#EFELER	http://www.izmirdeafet.com/Disaster.owl#Hard
http://www.izmirdeafet.com/Disaster.owl#INKILAP	http://www.izmirdeafet.com/Disaster.owl#Hard
http://www.izmirdeafet.com/Disaster.owl#MENDERES	http://www.izmirdeafet.com/Disaster.owl#Hard
http://www.izmirdeafet.com/Disaster.owl#VALIRAHMIBEY	http://www.izmirdeafet.com/Disaster.owl#Hard
http://www.izmirdeafet.com/Disaster.owl#BARIS	http://www.izmirdeafet.com/Disaster.owl#Hard
http://www.izmirdeafet.com/Disaster.owl#GUVEN	http://www.izmirdeafet.com/Disaster.owl#Hard
http://www.izmirdeafet.com/Disaster.owl#YIGITLER	http://www.izmirdeafet.com/Disaster.owl#Hard
http://www.izmirdeafet.com/Disaster.owl#ADATEPE	http://www.izmirdeafet.com/Disaster.owl#Rock
http://www.izmirdeafet.com/Disaster.owl#CUMHURIYET	http://www.izmirdeafet.com/Disaster.owl#Rock
http://www.izmirdeafet.com/Disaster.owl#KARANFIL	http://www.izmirdeafet.com/Disaster.owl#Rock
http://www.izmirdeafet.com/Disaster.owl#CAGDAS	http://www.izmirdeafet.com/Disaster.owl#Rock
http://www.izmirdeafet.com/Disaster.owl#MURATHAN	http://www.izmirdeafet.com/Disaster.owl#Rock
http://www.izmirdeafet.com/Disaster.owl#IZKENT	http://www.izmirdeafet.com/Disaster.owl#Rock
http://www.izmirdeafet.com/Disaster.owl#GAZILER	http://www.izmirdeafet.com/Disaster.owl#Rock
http://www.izmirdeafet.com/Disaster.owl#YAYLACIK	http://www.izmirdeafet.com/Disaster.owl#Rock
http://www.izmirdeafet.com/Disaster.owl#DICLE	http://www.izmirdeafet.com/Disaster.owl#Rock
http://www.izmirdeafet.com/Disaster.owl#LALELI	http://www.izmirdeafet.com/Disaster.owl#Rock
http://www.izmirdeafet.com/Disaster.owl#CAMLIK	http://www.izmirdeafet.com/Disaster.owl#Rock
http://www.izmirdeafet.com/Disaster.owl#UFUK	http://www.izmirdeafet.com/Disaster.owl#Rock
http://www.izmirdeafet.com/Disaster.owl#SIRINKAPI	http://www.izmirdeafet.com/Disaster.owl#Rock

Figure 5.8 GeoSPARQL results of intersect function between neighbourhood and basement type



Query2 Browser

SoilType	ID	mahalle_adi
Hard	30	SEYHAN
Hard	31	GOKSU
Hard	32	YESILBAGLAR
Hard	35	MUSTAFAKEMAL
Hard	36	INONU
Hard	37	HURRIYET
Hard	5	YIGITLER
Hard	6	GUVEN
Hard	16	INKILAP
Hard	17	EFELER
Hard	18	AKINCILAR
Hard	29	FIRAT
Hard	20	DUMLUPINAR
Hard	8	BARIS
Hard	9	VALIRAHMIBEY
Hard	10	MENDERES
Hard	19	KOZAGAC
Hard	28	YENIGUN
Hard	33	CAMLIKULE
Hard	34	CAMLIPINAR
Hard	11	YAYLACIK
Hard	23	ADATEPE
Hard	24	ATATURK
Hard	25	KURUCESME
Hard	26	BUCAKOOP
Hard	27	YILDIZ
Rock	3	CAMLIK
Rock	4	LALELI
Rock	2	UFUK
Rock	7	DICLE
Rock	14	MURATHAN
Rock	1	SIRINKAPI
Rock	12	GAZILER
Rock	13	IZKENT
Rock	15	CAGDAS
Rock	21	KARANFIL
Rock	22	CUMHURIYET

Figure 5.9 MapInfo results of intersect function between neighbourhood and basement type

5.3.1.2 Ground Acceleration Calculation

Ground acceleration is the measure of how much and how fast the basement is shaken during an earthquake. Most and least affected regions from earthquake can be determined with ground acceleration value (Campbell, 1997). Distance from epicenter and basement type variables above mentioned is acquiring with spatial queries for calculating ground acceleration. According to this scenario, all neighborhood ground acceleration values are defined for assessment damages (Figure 5.10). For example Adatepe neighborhood in Buca district distance from epicenter of the earthquake is ‘1299’ meter and basement type *‘Hard’* is queried with GeoSPARQL. Neighborhood damage estimation was determined as *‘Moderate’* after calculating acceleration formula above.

```
static void Accalc(ArrayList<String> queryR2, ArrayList<String> queryD2) throws IOException
{

String rock="http://www.izmirdeafet.com/Disaster.owl#Rock";
String hard="http://www.izmirdeafet.com/Disaster.owl#Hard";
String medium="http://www.izmirdeafet.com/Disaster.owl#Medium";
String soft="http://www.izmirdeafet.com/Disaster.owl#Soft";

for(int i=0;i<queryD.size();i++)
{
String check=queryD.get(i).substring(0, 1);
try
{
NumberFormat.getInstance().parse(check);
String s=queryD.get(i);
D= Double.parseDouble(s);
neighbourhood=queryR.get(i+1);
    if(queryR.get(i).equals(rock))
    {
        double A; //Sonuc
        double cons1=46;
        double cons2=0.208;
        double cons3=-0.686;
        double M=7;//Değişken
        M= M*cons2;
        D=D+10;
        A=cons1*Math.pow(10,M)*Math.pow(D,cons3);
        SonucRock=neighbourhood+";"+A;
    }
    else if(queryR.get(i).equals(hard))
    {
        double A; //Sonuc
        double cons1=24.5;
        double cons2=0.333;
        double cons3=-0.924;
        double M=7;//Değişken

        M= M*cons2;
        D=D+10;
        A=cons1*Math.pow(10,M)*Math.pow(D,cons3);
        SonucHard=neighbourhood+";"+A;

    }
    else if(queryR.get(i).equals(medium))
    {
        double A; //Sonuc
        double cons1=59;
        double cons2=0.261;
        double cons3=-0.886;
        double M=7;//Değişken

        M= M*cons2;
        D=D+10;
        A=cons1*Math.pow(10,M)*Math.pow(D,cons3);
        SonucMedium=neighbourhood+";"+A;

    }
    else if(queryR.get(i).equals(soft))
    {
        double A; //Sonuc
        double cons1=12.8;
        double cons2=0.432;
        double cons3=-1.112;
        double M=7;//Değişken

        M= M*cons2;
        D=D+10;
        A=cons1*Math.pow(10,M)*Math.pow(D,cons3);
        SonucSoft=neighbourhood+";"+A;
    }

    arySonucRock.add(SonucRock);
    arySonucHard.add(SonucHard);
    arySonucMedium.add(SonucMedium);
    arySonucSoft.add(SonucSoft);
    arySonucRock.remove(null);
    arySonucHard.remove(null);
    arySonucMedium.remove(null);
    arySonucSoft.remove(null);
```

Figure 5.10 Acceleration process in java application

Neighbourhood acceleration values are attached with *"hasAcceleration"* datatype property at triple store after calculating acceleration value with formula (Figure 5.11). Thus new reasoning using some specific rules and criteria can be done with specific inferences on neighbourhood in the future. For example if a neighbourhood

acceleration value over a specific value, neighbourhood damage assessment can be calculated.

```
static void addNeighbourhoodTripleStore() throws IOException
{
    File girendosya = new File("C:\\Users\\Can\\Desktop\\DisasterOntologyEnglish\\DisasterOutPellet3.owl");
    if(girendosya.exists())
    {
        String NS="http://www.izmirdeafet.com/Disaster.owl#";
        FileInputStream inputdosya= new FileInputStream(girendosya);
        Model model=ModelFactory.createDefaultModel();
        model.read(inputdosya,NS);
        RemoteModel rmParliament= new RemoteModel("http://localhost:38080/parliament/sparql","http://localhost:38080/parliament/bulk");
        rmParliament.insertStatements(model);
    }
    else
    {
        System.out.println("Dosya Yok!");
    }

}
```

Figure 5.11 Neighborhood acceleration values are inserted ontology into triple store

In consideration of method above, Pellet reasoning is used in Protégé and then exports ontology into Parliament triple store for full support reasoning and querying. Reasoning is made by using Pellet reasoner for making inferencing with predefined DL rules for neighbourhood damage situation by attaching results obtained from ground acceleration formula into triple store. Inferred ontology by using Pellet reasoner is exported into triple store. GeoSPARQL queries can be made according inference ontologies in the future.

5.4 Querying the Ontology and Results

All neighborhoods in Buca district ground acceleration value are calculated by using acceleration formula in accordance with rules for damage assessment in above section. This information which is queried from triple store needs to be published on maps. In the implemented prototype, JSP (Java Server Pages) was used to publish disaster 3.0 web site (Figure 5.12).

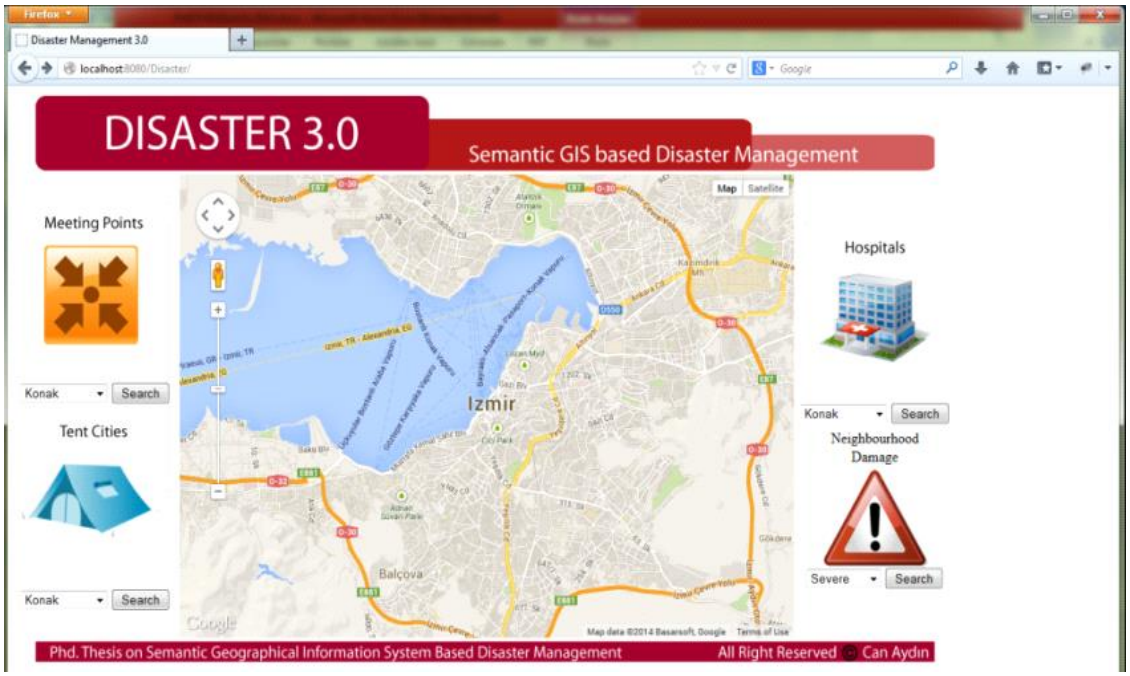


Figure 5.12 Semantic GIS based disaster management web site graphical user interface

Apache Tomcat Server used a container for web application. Based on the proposed methodology, querying chance from multiple sources with query language for manager is provided to make efficient decision making. SPARQL protocol (SPARQL) is used for query language and SPARQL can be used to express queries across diverse data sources, whether the data stored natively as RDF or OWL (W3C, 2008). Parliament triple store works on Joseki server for getting query results. Graphical user interface (GUI) components are comprised such as closed roads, broken bridges and neighborhood according to damage ratio queries. Results of the queries are published with Google Map API. While publishing maps, first request is sent to the server about required information such as available meeting places in damaged area. After sending request to java class, http servlet sends SPARQL query to triple store. This query statement "*SELECT ?damage ?geo WHERE {?damage a dis:meetingPoint; geo:hasGeometry ?fgeom. ?fgeom geo:asWKT ?geo}";*" is used for querying available meeting points on damaged area (Figure 5.13). Parliament sends query results with DL reasoning ability as inference result into java class back. The built in reasoning algorithm of Parliament is used to infer features. This response consists meeting point name and spatial objects as coordinates in WKT format.

```
protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException
{
    String meetingPoint= request.getParameter("meetingPoint");
    //PARLIAMENT TRIPLE STORE QUERY
            String queryString=
                    "PREFIX afn: <http://jena.hpl.hp.com/ARQ/function#>"+
                    "PREFIX fn: <http://www.w3.org/2005/xpath-functions#>"+
                    "PREFIX owl: <http://www.w3.org/2002/07/owl#>"+
                    "PREFIX par: <http://parliament.semwebcentral.org/parliament#>"+
                    "PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>"+
                    "PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>"+
                    "PREFIX time: <http://www.w3.org/2006/time#>"+
                    "PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>"+
                    "PREFIX geo: <http://www.opengis.net/ont/geosparql#>"+
                    "PREFIX geof: <http://www.opengis.net/def/function/geosparql/>"+
                    "PREFIX sf: <http://www.opengis.net/ont/sf#>"+
                    "PREFIX dis:<http://www.izmirdeafet.com/Disaster.owl#>"+
                    "PREFIX uom: <http://www.opengis.net/def/uom/OGC/1.0/>"+
                    "SELECT ?damage ?geo WHERE{?damage a dis:"+meetingPoint+";geo:hasGeometry ?fgeom. ?fgeom geo:asWKT ?geo }";
            RemoteModel rmParliament= new RemoteModel("http://localhost:38080/parliament/sparql","http://localhost:38080/parliament/bulk");
            ResultSet results = rmParliament.selectQuery(queryString);
            ByteArrayOutputStream baos = new ByteArrayOutputStream();
            PrintStream ps = new PrintStream(baos);
            ResultSetFormatter.outputAsJSON(ps,results);
            String s = new String(baos.toByteArray(), "UTF-8");
            PrintWriter out = new PrintWriter("C:\\Users\\Can\\Desktop\\DisasterOntologyEnglish\\geoData\\gml\\damageResult.json");
            out.write(s);
            out.close();
            request.setAttribute("damage",s);
            RequestDispatcher dispatcher = request.getRequestDispatcher("success.jsp");
            dispatcher.forward(request, response);
}
```

Figure 5.13 HTTP request for triple store query java class

This request results get output as JSON format in server. This page was forwarded to success page for displaying results while requesting. In success page google maps are created as base map for displaying spatial objects. First results in JSON format are read with Javascript language and send into array. After then array is splitted and coordinates are read from result array. Finally spatial objects (polygon, point) are created with these coordinates and setting on map for publishing (Figure 5.14).

```
function init()
{
var mapOptions = {zoom: 16,center: new google.maps.LatLng(38.394337,27.190753),mapTypeId: google.maps.MapTypeId.ROADMAP};
map = new google.maps.Map(document.getElementById("map-canvas"),mapOptions);
var sr = <%=request.getAttribute("damage") %> ;
var bindings = sr.results.bindings;
var polygon=[];
var polyline= [];
var space=' ';
var latlng;
////////////////////////////////////////////////////////////////////////////////////////////
for(j=0;j<bindings.length;j++)
{
    var binding= bindings[j];
    for(r in binding)
    {
        if(r=="geo")
        {
            gelen= binding[r].value;
            coord= gelen.replace('<http://www.opengis.net/def/crs/OGC/1.3/CRS84>POLYGON ((',' ');
            newcoord= coord.slice(0,-2);
            newcoord2= newcoord.split(',');
           for(i=1;i<newcoord2.length;i++)
               {
               var clat= newcoord2[i].split(space);
               var coordinatesLat=parseFloat(clat[0]);
               coordinatesLat= coordinatesLat.toFixed(6);
               var coordinatesLong=parseFloat(clat[1]);
               latlng= new google.maps.LatLng(coordinatesLong,coordinatesLat,noWrap=true);
               polyline.push(latlng);
               }
               polygon[j]= new google.maps.Polygon({paths: polyline,strokeColor: "#FF0000",strokeOpacity: 0.4,strokeWeight: 2,
               fillColor: "#FF0000",fillOpacity: 0.35,clickable: true});
               polygon[j].setMap(map);
               polyline=[];
        }
    }
}
```

Figure 5.14 Adding spatial objects into Google map

Figure 5.15 illustrates query example using GeoSPARQL for the geometry of severe damage neighborhoods. This result can be described as the first stage of the query chain. Several query results such as meeting place, hospital, security control points, logistic support control center, head distribution center, police office according to damage status constituting in front of disaster managers after defining severe damaged neighborhood.

```
protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException
{
    String damage= request.getParameter("damage");
    //PARLIAMENT TRIPLE STORE QUERY
            String queryString=
                    "PREFIX afn: <http://jena.hpl.hp.com/ARQ/function#>"+
                    "PREFIX fn: <http://www.w3.org/2005/xpath-functions#>"+
                    "PREFIX owl: <http://www.w3.org/2002/07/owl#>"+
                    "PREFIX par: <http://parliament.semwebcentral.org/parliament#>"+
                    "PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>"+
                    "PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>"+
                    "PREFIX time: <http://www.w3.org/2006/time#>"+
                    "PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>"+
                    "PREFIX geo: <http://www.opengis.net/ont/geosparql#>"+
                    "PREFIX geof: <http://www.opengis.net/def/function/geosparql/>"+
                    "PREFIX sf: <http://www.opengis.net/ont/sf#>"+
                    "PREFIX dis:<http://www.izmirdeafet.com/Disaster.owl#>"+
                    "PREFIX uom: <http://www.opengis.net/def/uom/OGC/1.0/>"+
                    "SELECT ?damage ?geo WHERE{?damage a dis:"+damage+";geo:hasGeometry ?fgeom. ?fgeom geo:asWKT ?geo }";
            RemoteModel rmParliament= new RemoteModel("http://localhost:38080/parliament/sparql","http://localhost:38080/parliament/bulk");
            ResultSet results = rmParliament.selectQuery(queryString);
            ByteArrayOutputStream baos = new ByteArrayOutputStream();
            PrintStream ps = new PrintStream(baos);
            ResultSetFormatter.outputAsJSON(ps,results);
            String s = new String(baos.toByteArray(), "UTF-8");
            PrintWriter out = new PrintWriter("C:\\Users\\Can\\Desktop\\DisasterOntologyEnglish\\geoData\\gml\\damageResult.json");
            out.write(s);
            out.close();
            request.setAttribute("damage",s);
            RequestDispatcher dispatcher = request.getRequestDispatcher("success.jsp");
            dispatcher.forward(request, response);
}
```

Figure 5.15 Selecting moderate damage neighborhoods

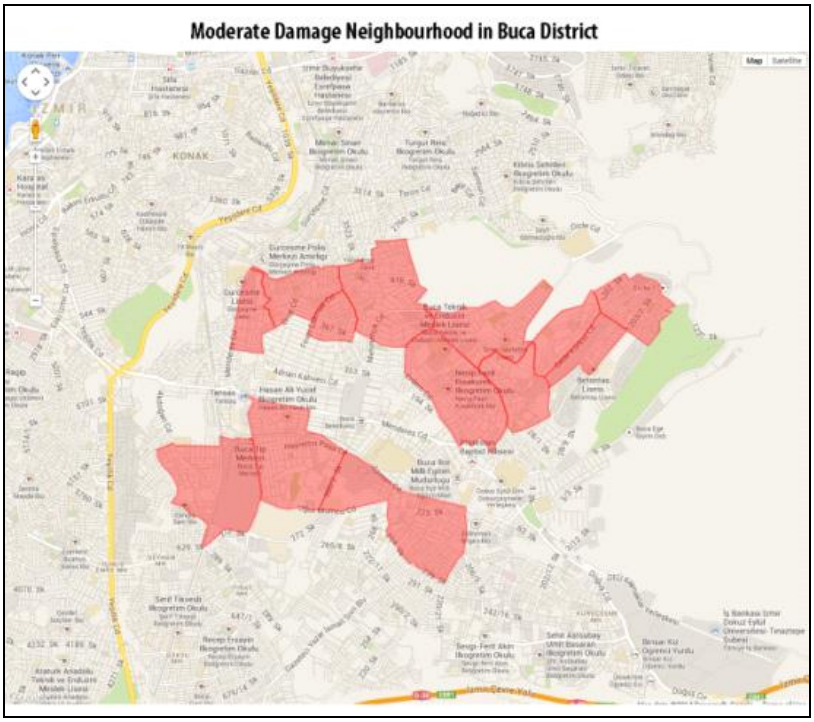


Figure 5.16 Moderate damage neighborhood result

Similar queries can be done on severe damage neighbourhood in Buca district in which it is damaged area from earthquake (Figure 5.16). Severe value as neighborhood situation is sent to the triple store. Severe damage neighborhood map demonstrates to the user where response in priority (Figure 5.17). These

neighborhood populations are under the risk. Decision maker has to forward resources and disaster squads.

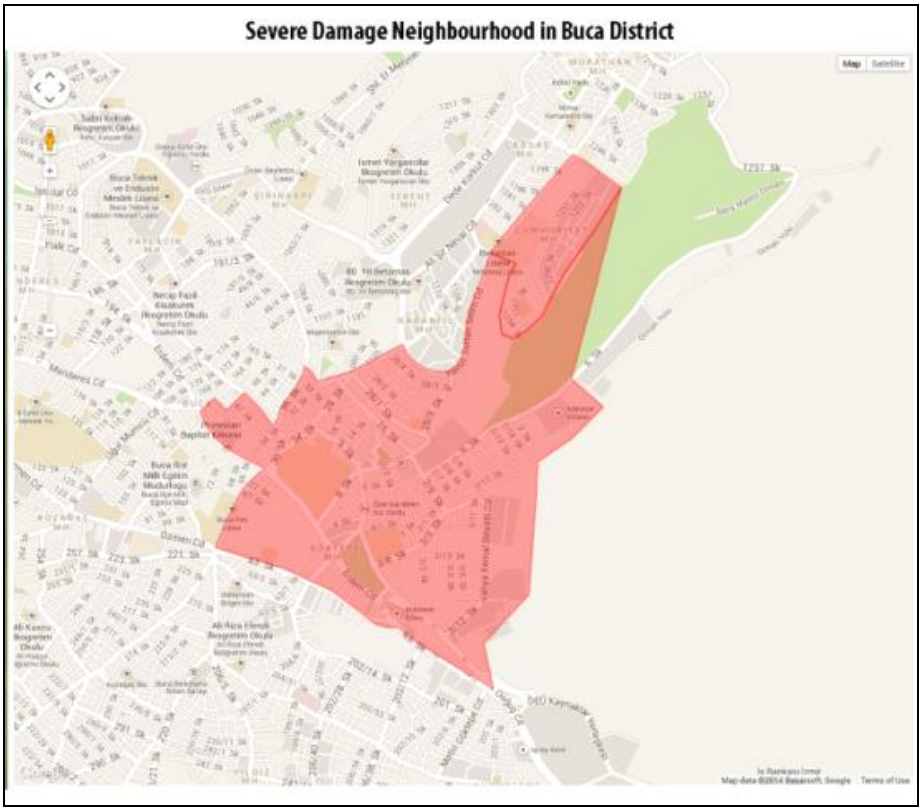


Figure 5.17 Severe damage neighborhood queries

Another query result is closed roads and broken bridges according to damage status. In this way despite distributed rescue and the other types of team in city, shortest path for rescue team can be determined to the disaster area (Figure 5.18 - 5.19).

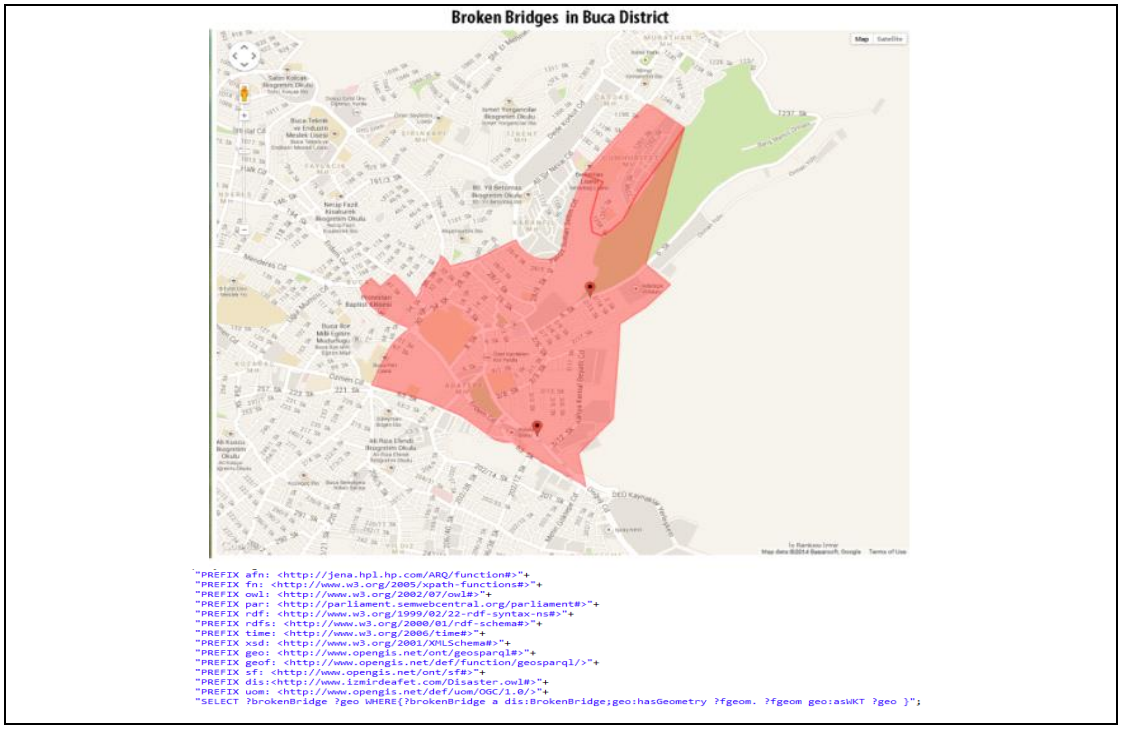


Figure 5.18 Broken bridges according to neighborhood damage

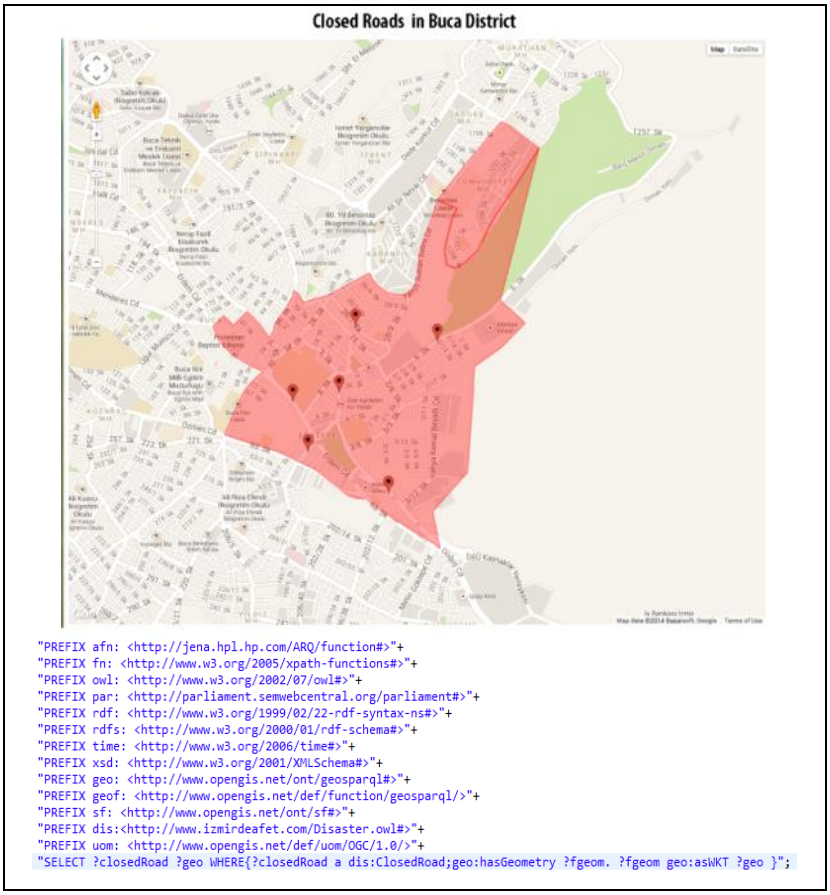


Figure 5.19 Closed roads according to neighborhood damage

The exact process of the scenario is summarized in Chapter 3.5.1.2 scenario and competency questions. Closest and proper meeting places around disaster area for affected people from earthquake are determined after results are obtained according to neighbourhood damage situation. According to scenario disaster manager wants to show that available meeting places within severe damaged neighborhood as basic query (Figure 5.20).

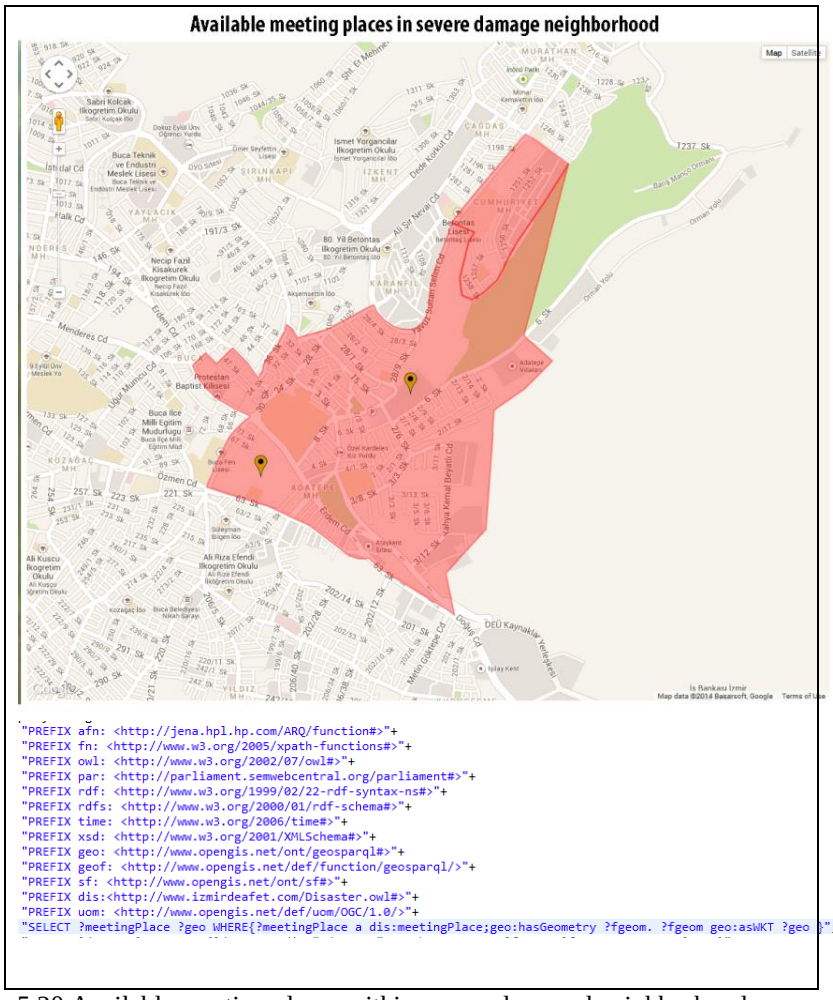


Figure 5.20 Available meeting places within severe damaged neighborhood

The other basic example based on different charge “health services led to injured people to hospitals” on disaster scenario; within function is used to determine nearest hospitals severe neighborhood damage (Figure 5.21). Hospitals and neighborhood location information which comes from different sources such as hospital locations from health institution and severe neighborhood information comes acceleration formula are used for determining this query. For example hospital information is in

MapInfo format. The above algorithm is applied and processed to the hospital information and used for SPARQL query. Therefore, interoperability is provided between the distributed data sources.

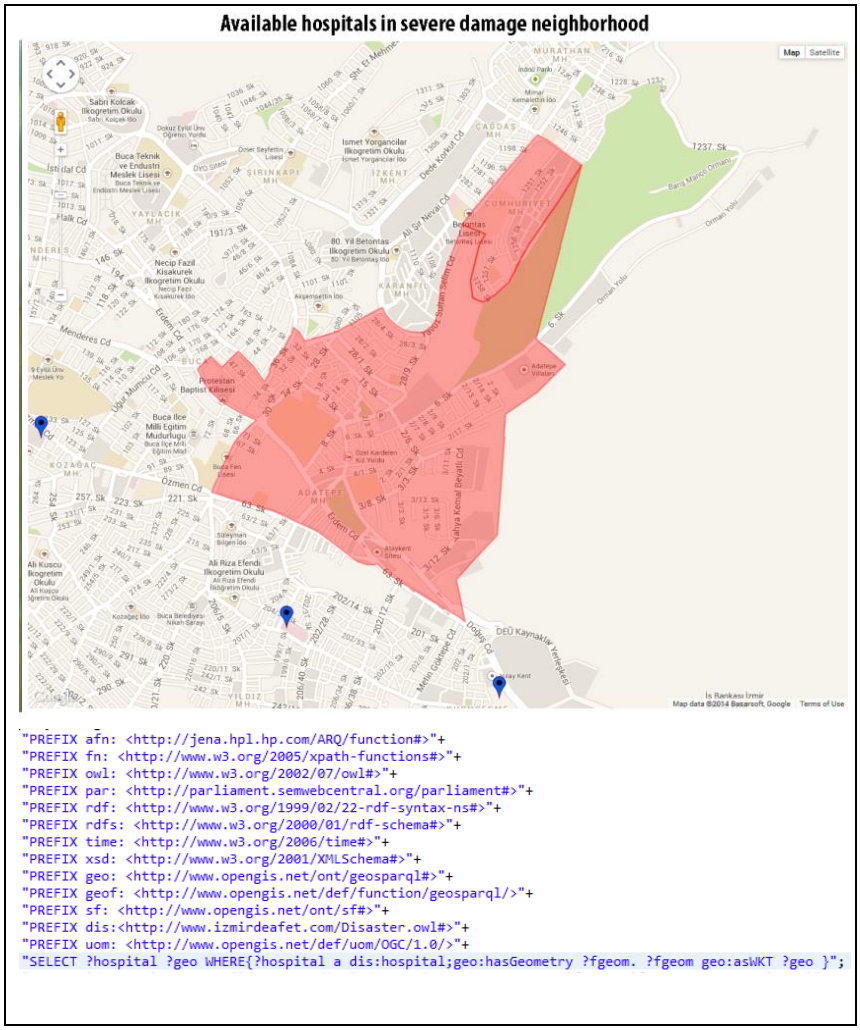


Figure 5.21 Available hospitals within severe damaged neighborhood

According to scenario preventing criminal activities such as burglar, rape available police stations are determined for prevailing crime in disaster area (Figure 5.22). The location information of these police stations is procured crudely related institutions as MapInfo file format. Resource distribution is very crucial event under disaster situations. Several problems appeared during providing and distribution of resources. Disaster manager cannot forecast aid distribution amount and distribution locations because of which region is more affected from disaster cannot be detected previously. For this reason this system can determine the exact distribution location

by using neighborhood population and neighborhood damage ratio using DL rules on disaster ontology (Figure 5.23).

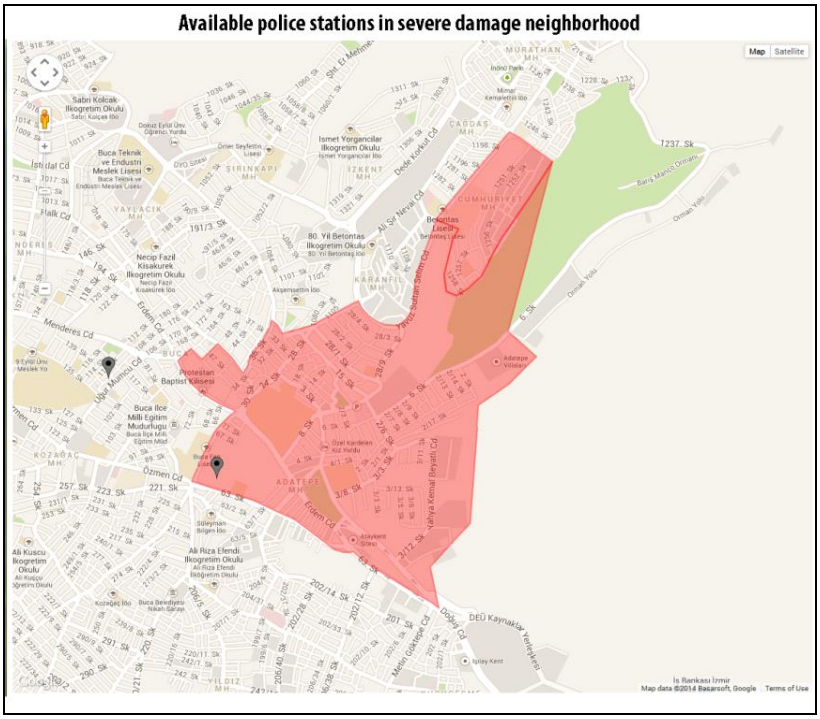


Figure 5.22 Available security points within severe damaged neighborhood

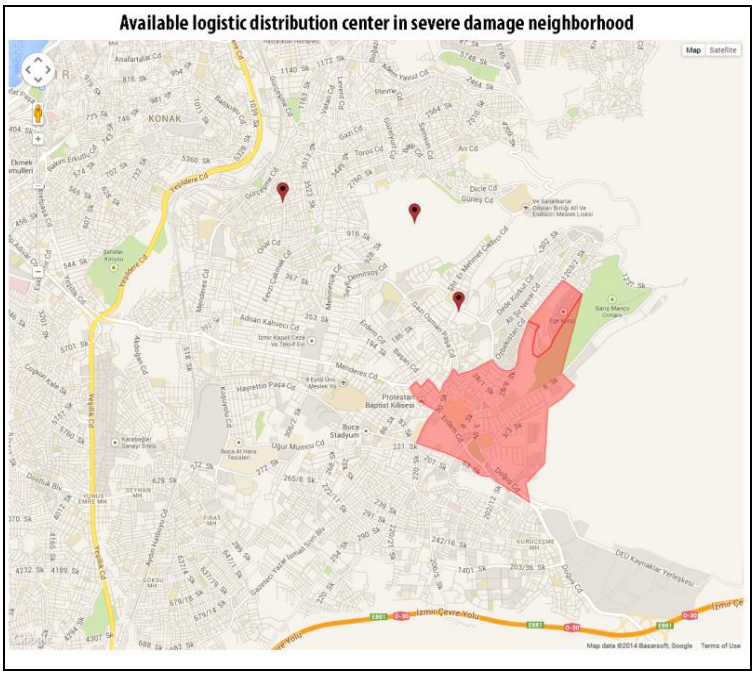


Figure 5.23 Available logistic distribution centers within severe damaged neighborhood

Managers can interfere to disaster more quickly and accurate owing to all map based queries. This system can be used easily by disaster managers due to have user-friendly interface.

CHAPTER SIX
CONCLUSION

After presenting the problem, it is attempted to tackle, to motivate the approach, to describe the proposed solution and to evaluate its deployment, it is time to conclude the thesis in Section 6.1. Section 6.2 recalls the research question and discusses to which extent we managed to provide appropriate answers. A general vision is outlined of future developments based on the ideas elaborated in the thesis in Section 6.3.

The high-level goal of the presented thesis was to provide data interoperability between data sources in disaster management domain. It is done so by proposing a framework for the representation and processing of knowledge about disaster from distributed data sources. The framework is proposed and implemented with semantic web technologies. Disaster ontology designs and integrates spatial data within framework for achieving this aim. Disaster manager can make decisions more efficiently and effectively using reasoning capabilities with DL rules in disaster ontology. Our framework has shown that promising capabilities that unstudied approaches such as designing earthquake ontology, earthquake damage assessment with DL rules and integrate spatial data with geo-ontology in proprietary format. In this way the research can contribute to the disaster management domain. We elaborate on the specific thesis goals and results achieved by recalling the addressed research questions below.

6.1 Answering the Research Questions

6.1.1 How Geo-ontologies can be Utilized in Earthquake Domain for Disaster Management on Urban Areas

Several ontology-based studies are conducted for providing data interoperability on disaster management. But in the literature there is no ontology in earthquake

domain. Availing oneself of this opportunity earthquake ontology is modelled in detail in Chapter 3. Conceptualization, specification and implementation phase are performed with expert on disaster and disaster management. Describing domain and competency questions are created with disaster scenario on specification phases. Structure of disaster management can be explained in detail during while describing domain. What kinds of questions are responded with this system is defined on competency questions. Classes and properties are created and defined terms about earthquake on conceptualization stage. Domains and ranges are defined between properties. DL rules are defined for reasoning and damage assessment during earthquake ontology design. These rules are explained in detail in Chapter 5. At the same time GeoSPARQL ontology is used during ontology design for spatial objects in ontology. All spatial based objects are integrated into ontology under *spatialObject* class of GeoSPARQL ontology. This spatial based ontology provides spatial query capability to the system. Therefore spatial data from distributed data sources can be integrated into disaster ontology in earthquake domain.

6.1.2 How to Integrate Spatial Data from Distributed Data Sources

Mostly semantic integration is used on spatial web services with metadata in literature. Spatial data integration into ontology from distributed proprietary formats without any metadata rather than web service is challenging task as mentioned in Chapter 1. To overcome this problem, all data sources in different format are converted into GML with FME software and then class and property in ontology and GML nodes are matched on java application. Semi-automatic methods in other words human interaction is used during matching between GML nodes and ontology class and property. In fact this matching was carried out automatically, both systems will work more efficiently and system can map data sources into ontology quickly. But automatic integration of data sources is out of the thesis scope. This problem is a separate research question. Thereby spatial data in different format is integrated into disaster ontology with spatial capabilities. For example earthquake location and magnitude value, basement type values which are used for damage assessment can integrate ontology and they are used in acceleration formula. Integration method is

explained in the next research question. Difficulties on spatial data integration into ontology cannot insert spatial and non-spatial data within the same method into ontology. To overcome this problem; first non-spatial data is integrated into ontology by using Jena library in chapter 4.2.1. Then, these non-spatial data coordinate information are inserted on Parliament triple store by using SPARQL query statement and Parliament remote model in chapter 4.2.2. Using two different methods for integration of spatial and non-spatial data into ontology is seen as disadvantage for fully automatic system.

6.1.3 How can Damage Assessment Estimate Early Post-disaster by Using DL Languages

Earthquake damage estimation is very important for decision making and planning after disaster. Damage estimation is determined by using strong ground motion network. Although damage assessment can be made static and heterogeneous data sources with decision tree method, it is very difficult to automate the discoveries of these geospatial features and make reasoning if the semantic disparity exists. There is no way to automatically retrieve the desired information from semantically heterogeneous and distributed data sources with human integration. To expedite automatic damage assessment, DL-based damage assessment is proposed which has integrated heterogeneous data sources, reasoning and querying capabilities. It is focused on intelligent damage assessment with DL rules and creating map interface for decision maker to make decision efficiently. Earthquake horizontal acceleration calculation formula is used for earthquake damage assessment. In this context basement type, distances from epicenter and magnitude information are used for calculation. The proposed methodology rakes together power of ontology and damage assessment methods. The results of the implementation show that it is possible to make damage assessment with DL rule based ontologies where strong ground motion network lack of cities. Distributed data sources turn out to be machine understandable and gain inference capabilities within the DL rules and easily queried these data sources for damage assessment owing to ontologies. The results show that spatial queries from distributed data sources are possible by using GeoSPARQL and

Parliament Triple Store together. By providing interface for accessing damage assessment information, the proposed framework can provide to use easily all levels of disaster managers without prevailing semantic web technologies. In other words disaster managers dont need to understand DL language or ontology for damage assessment with ontologies.

6.2 Future Work

Some challenges are examined and they realized both of three research questions. Some points of these challenges are out of thesis scope and will be investigated in the future. Automatic integration of spatial data sources without manual intervention is very crucial for this field. New data source can be added quickly and minimum error with automatic integration of spatial data sources. Two different methods are used such as spatial and non-spatial data integration for integrating data sources into ontology. First non-spatial data with Jena library and then spatial data with triple store are integrated with ontology. This integration of both data type should be made with the same method for more rapid response from framework. This study mostly focuses on response phase in disaster management, but this method can be applied into preparedness and recovery phases for different purpose. Such as efficient preparation to the disaster by providing semantic heterogeneity between data source on disaster management or recover the effects from disaster which available site selection for new housing and settlement.

REFERENCES

Abdelmoty, A. I., Smart, P. D., Jones, C. B., Fu, G., & Finch, D., (2005). A critical evaluation of ontology languages for geographic information retrieval on the Internet. *Journal of Visual Languages & Computing*, *16*(4), 331-358.

Abel, D. J., Ooi, B. C., Tan, K. L., & Tan, S. H., (2001). Towards integrated geographical information processing. *International Journal of Geographical Information Science*, (12), 353-371.

Akyel, R. (2007). *Disaster management system: challenges on identification and solution problems of Turkish disaster management.* Retrieved February 06, 2014 from http://sosyalbilimler.cu.edu.tr/tez/940/.

Alam, A., Khan, L., & Thuraisingham, B. (2011). Geospatial resource description framework (GRDF) and security constructs. *Computer Standards & Interfaces*, (33-1), 35-41.

Alparslan, E., Ince, F., Erkan, B., Aydöner, C., Özen, H., Dönertaş, A., et al. (2008). A GIS model for settlement suitability regarding disaster mitigation, a case study in Bolu Turkey. *Engineering Geology, 96* (3), 126-140.

Andrei, M., Berre, A., Costa, L., Duchesne, P., Fitzner, D., Gurcar, M. et al., (2008). Swing: An integrated environment for geospatial semantic web services. *In The Semantic Web: Research and Applications* (767-771).

Arpinar, I., Sheth, A., & Ramakrishnan, C., (2006). Geospatial ontology development and semantic analysis, *Handbook of Geographic Information Science*, Wilson, J. P. (Ed.), *10*(4), 551–575.

Atemezing, A. & Troncy, R. (2012). Comparing vocabularies for representing geographical features and their geometry. *Terra Cognita 2012 Foundations, Technologies and Applications of the Geospatial Web*. (3-14).

Babitski, G., Bergweiler, S., Grebner, O., Oberle, D., Paulheim, H., & Probst, F. (2011). SoKNOS – Using semantic technologies in disaster management software. *In The Semanic Web: Research and Applications. (*183-197). Springer Berlin Heidelberg.

Berners, L. T., (2001). The Semantic web - A new form of web content that is meaningful to computers will unleash a revolution of new possibilities. *Scientific American* (284-5), 34.

Bilham, R. (2009). The seismic future of cities, *Bulletin of Earthquake Engineering*, (7-4), 839-887.

Bishr, Y. (1998). Overcoming the semantic and other barriers to GIS interoperability. *International Journal of Geographical Information Science*, (12-4), 299-314.

Boucher, S., & Zimányi, E., (2009). Leveraging OWL for GIS interoperability: rewards and pitfalls. *In Proceedings of the 2009 ACM Symposium on Applied Computing* (1267-1272). ACM.

Brusa, G., Laura Caliusco, M., & Chiotti, O., (2008). Towards ontological engineering: A process for building domain ontology from scratch in public administration. *Expert Systems*, (*25*-5), 484-503.

Campbell, K. W. (1997). Empirical near-source attenuation relationships for horizontal and vertical components of peak ground acceleration, peak ground velocity, and pseudo-absolute acceleration response spectra. *Seismological Research Letters*, (*68*-1), 154-179.

Chaves, M., Silva, M., & Martins, B. (2005). A geographic knowledge base for semantic web applications. *20th Brazilian Symposium on Databases-SBBD*. (05), 40-54.

Chou, C. H., Zahedi, F., & Zhao, H. (2011). Ontology for developing web sites for natural disaster management: methodology and implementation. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, (*41*-1), 50-62.

Chrisman, N.R., (1995). Beyond Stevens: A revised approach to measurement for geographic information, *Autocarto Conference*, (4), 271–280.

Christodoulou, G., Petrakis, E. G., & Batsakis, S. (2012). Qualitative spatial reasoning using topological and directional information in OWL. *In Tools with Artificial Intelligence (ICTAI), 2012 IEEE 24th International Conference on* (1), (596-602). IEEE.

Coppola, D. P. (2011). *Introduction to international disaster management* (Second Edition), Butterworth-Heinemann

Cruz, I. F., & Xiao, H. (2005). The role of ontologies in data integration.*Engineering Intelligent Systems for Electrical Engineering and Communications,13*(4), 245.

Daconta, M. C., Obrst L. J. & Smith, K. T. (2003). *The Semantic Web: A guide to the future of XML, web services, and knowledge management.* John Wiley & Sons.

Daniele, N., & Brachman, R. (2003). An introduction to description logic, *In description logic handbook*, 1-40.

Emmons, I. (2012). *Parliament user guide.* Retrieved November 15, 2013 from http://parliament.semwebcentral.org/ParliamentUserGuide.pdf.

Erdik, M., Şeşetyan, K., Demircioğlu, M. B., Hancılar, U., & Zülfikar, C. (2010). Rapid earthquake loss assessment after damaging earthquakes, soil dynamics and earthquake engineering, (31-2), 247-266.

Erl, T. (2005). *Service-oriented architecture concepts, technology, and design.* Retrieved June 15, 2013 from http://web.uncg.edu/bae/ documents/syllabi/data/ 200705/ISM/ISM/ISM65841.pdf

Fan, Z. & Zlatanova, S. (2010). Exploring ontology potential in emergency response, *In Proceedings of the Gi4DM Conference—geomatics for Disaster Management,* Torino, Italy.

Fan, Z. & Zlatanova, S. (2011). Exploring ontologies for semantic interoperability of data in emergency response. *Applied Geomat,* (3), 109-122.

Fernández-López, M., Gómez-Pérez, A., & Juristo, N. (1997). Methontology: From ontological art towards ontological engineering. *AAAI Technical Report*, 97-06

Fonseca, F. T. & Egenhofer, M. J. (1999). Ontology-driven geographic information systems, *Proceedings of the 7th ACM International Symposium on Advances in Geographic Information Systems*, (14-19).

Fonseca, F. T., Egenhofer, M. J., Davis Jr, C. A., & Borges, K. A. (2000). Ontologies and knowledge sharing in urban GIS. Computers, *Environment and Urban Systems*, *24*(3), 251-272.

Fonseca, F., Câmara, G., & Miguel Monteiro, A. (2006). A framework for measuring the interoperability of geo-ontologies. *Spatial Cognition and Computation*, (6-4), 309-331.

Fonseca, L. R. & Llano, E. G. (2011). Automatic representation of geographical data from a semantic point of view through a new ontology and classification techniques, *Transactions in GIS*, (15-1), 61-85.

Forte, M., de Souza, W. L., & do Prado, A. F. (2008). Using ontologies and web services for content adaptation in ubiquitous computing, *The Journal of Systems and Software* (81-3), 368-381.

Galton, A. (2003). Desiderata for a spatio-temporal geoontology. *Spatial Information Theory*. Foundations of Geographic Information Science. *Lecture Notes in Computer Science*, (2825), 1-12.

Gruninger, M. (1996). Designing and evaluating generic ontologies. In*Proceedings of the 12th European Conference of Artificial Intelligence* (53-65).

Gruninger, M., & Fox, M. S. (1995). Methodology for the design and evaluation of ontologies.

Grütter, R., & Messmer, B. B., (2007). Combining OWL with RCC for spatioterminological reasoning on environmental data, *CEUR Workshop Proceedings* (258).

Grütter, R. & Bauer-Messmer, B. (2007). Towards spatial reasoning in the Semantic Web: A hybrid knowledge representation system architecture. *The European Information Society*. Springer-Verlag Berlin Heidelberg. 349-364.

Guan, J. H., Zhou, S. G., Chen J. P., Chen, X. L., An, Y., Yu, W. et al., (2003). Ontology-based GML schema matching for spatial information integration, *Second International Conference on Machine Learning and Cybernetics*, (5), 2240-2245.

Guarino, N. (Ed.). (1998). Formal ontology in information systems: *Proceedings of the first International Conference (FOIS'98)*, June 6-8, Trento, Italy (46). IOS press.

Gurenko, E., & Zelenko, P. (2007). *Global catastrophe facility for developing countries, Its financial viability and economic benefits: A Quantitative Assessment.* Mimeo. WorldBank.

Henriksson R., Kauppinen T., & Hyvönen, E. (2008). Core geographical concepts: case Finnish geo-ontology. *LOCWEB'08 Proceeding of the First International Workshop on Location and the Web,* (57-60).

Hiramatsu, K., & Reitsma, F. (2004). GeoReferencing the semantic web: ontology based markup of geographically referenced information. *In Joint EuroSDR/EuroGeographics workshop on Ontologies and Schema Translation Services*, (15-16).

Hristidis, V., Chen S. C., Li, T., Luis, S., & Deng, Y. (2010). Survey of data management and analysis in disaster situations, *Journal of Systems and Software* (83), 1701-1714.

Horridge, M., Knublauch, H., Rector, A., Stevens, R., & Wroe, C. (2004). A Practical guide to building OWL ontologies using the Protégé-OWL plugin and CO-ODE tools edition 1.0. *University of Manchester*.

Horrocks, I., Patel-Schneider, P.F., & van Harmelen, F. (2003). From SHIQ and RDF to OWL: The making of a web ontology language, *Journal of Web Semantics* (1), 7–26.

Horrocks, I., Patel-Schneider, P. F., McGuinness, D. L., & Welty, C. A. (2007). OWL: A description logic based ontology language for the semantic web.

Huang, Y. & Deng, G. (2009). Research on representation of geographic spatio-temporal information and spatio-temporal reasoning rules based on geo-ontology and SWRL, *International Conference on Environmental Science and Information Application Technology*, 381-384.

Iwanaga, I. S. M., Nguyen, T. M., Kawamura, T., Nakagawa, H., Tahara, Y., & Ohsuga, A. (2011). Building an earthquake evacuation ontology from twitter. In *Granular Computing (GrC), 2011 IEEE International Conference on* (306-311). IEEE.

Kaiser, C. F., Sattler, D. N., Bellack D. R., & Jennifer, D. (1996). A conservation of resources approach to natural disaster, *Journal of Social Behavior & Personality*, (11), 459.

Kalabokidis, K., Athanasis, N., Vaitis, M., Chang, K. T., & Bran, D. (2011). OntoFire: an ontology-based geo-portal for wildfires. *Natural Hazards & Earth System Sciences*, (11-12).

Karagöz, Ö., Özel, O., & Tan, O., (2012). Analyzing acceleration waveforms and determining source parameters of 23 October 2011 Van earthquake (Mw = 7.2), *IESCA-2012 - International Earth Science Colloquium on the Aegean Region-2012*, 01-05 October, Dokuz Eylül University-İzmir, Abstracts, (322) (Oral Presentation).

Kirsh, D., Peterson, N., & Lenert, L. A. (2004). An ontology of geo-reasoning to support medical response to attacks with weapons of mass destruction. *In AMIA. Annual Symposium proceedings/AMIA Symposium. AMIA Symposium* (2005), (400-404). American Medical Informatics Association.

Klien, E., Lutz, M. & Kuhn, W. (2006). Ontology-based discovery of geographic information services – an application in disaster management. *Computers, Environment and Urban Systems*. (30-1), 102-123.

Knublauch, H., Horridge, M., Musen, M. A., Rector, A. L., Stevens, R., Drummond, N. et al., (2005). *The Protege OWL experience*. In*OWLED*.

Komesli, M. (2007). *Integrating sermantic web and geographical information system with using OWL and GML*. Retrieved January 3, 2013 from https://tez.yok.gov.tr/UlusalTezMerkezi/TezGoster?key=7d53ed97e31a8bd39defac1ccce0a718dc0379c5005f01a86ae5db4d164a0c33b5e461eff52e4814c64260d6d1ffbf5

Kolas, D., Hebeler, J., & Dean, M. (2005). Geospatial semantic web: Architecture of ontologies. In *GeoSpatial Semantics* (183-194). Springer Berlin Heidelberg.

Kreimer, A., & Arnold, M. (2000). World Bank's role in reducing impacts of disasters, *Natural Hazards Review*, (1), 37-42.

Kreimer, A. (2000). *Uganda post-conflict reconstruction country case study series*, World Bank Operations Evaluation Department Publications.

Kruchten, P., Woo, C., Monu, K. & Sootoodeh, M. (2007). A human-centered conceptual model of disasters affecting critical infrastructures, *Proceedings of the 4th International Conference on Information Systems for Crisis Response and Management (ISCRAM2007),* Delft, The Netherlands:327–344.

Kyzirakos, K., Karpathiotakis, M. & Koubarakis, M. (2013). Strabon: A semantic geospatial DBMS, *Lecture Notes in Computer Science*, (7649), 295-311.

Lang, D., Molina-Palacios, S. & Lindholm, C. D. (2008). Towards near-real-time damage estimation using a CSM-based tool for seismic risk assessment. *Journal of Earthquake Engineering*. (12-2), 199-210.

Langegger, A. (2010). *A flexible architecture for virtual information integration based on semantic web concepts.* Retrieved October 10, 2013 from http://www.langegger.at/papers/Langegger-phd-thesis.pdf.

Lee, T. B., Hendler, J., & Lassile, O. (2001). The semantic web, *Scientific American.*

Lezcano, L., Santos, L., & García-Barriocanal, E. (2013). Semantic integration of sensor data and disaster management systems: The Emergency Archetype Approach. *International Journal of Distributed Sensor Networks, 2013*.

Ling, J., Bin, L., Jianya, G., & Min, M. (2006). Georeferencing the semantic web based on geoontology. *In Geoscience and Remote Sensing Symposium*, 2006. IGARSS 2006. *IEEE International Conference* (1545-1548).

Liu, G., Wu, C., Ma, X., Wang, Y. & Tian, F. (2009). Approach for interoperability of multi-source geological hazard data based on ontology and GeoSciML. Geoinformatics, *2009 17th International Conference.* (1-5), IEEE.

Lopez-Pellicer, F. J., Silva, M. J., Chaves, M. S., & Rodrigues, C. (2009). Geographic ontologies production in GREASE-II.

Lutz, M., & Kolas, D. (2007). Rule-based discovery in spatial data infrastructure. *Transactions in GIS*, (11-3), 317-336.

Maue P. & Schade S. (2010). Data integration in the geospatial semantic web, *Journal of Cases on Information Technology*, (11-4), 100-122.

McBride, B. (2002). Jena: A semantic web toolkit. *IEEE Internet Computing*, (*6*-6), (55-59).

Mennis, J. L., (2003). Derivation and implementation of a semantic GIS data model informed by principles of cognition. *Computers, Environment and Urban Systems* (27), 455–479.

Middleton, S. E., Sabeur, Z., Lowe, P., Hammitzsch, M., Tavakoli, S., & Poslad, S. (2013). Multi-disciplinary approaches to intelligently sharing large-volumes of real-time sensor data during natural disasters. *Data Science Journal*, (*12),* 109-113.

Miron, A. D., Gensel, J., Villanova-Oliver, M., & Martin, H. (2007). Towards the geo-spatial querying of the semantic web with ONTOAST. *In Web and Wireless Geographical Information Systems* (121-136). Springer Berlin Heidelberg.

Nardi, D., & Brachman, R. J. (2003). An Introduction to description logics. In *Description logic handbook* (1-40).

Nikolaos A., Kostas K., Michail, V., & Nikolaos, S. (2005). The emerge of semantic geoportals. In on the move to meaningful internet systems 2005: OTM 2005 Workshops, (1127-1136), Springer Berlin Heidelberg.

Noji, E. K. (1999). The public health consequences of disasters. *Public Health and Disaster Relief Conference*, University of California.

Novacek, V. (2010). *EUREEKA! Towards a practical emergent knowledge processing.* Retrieved November 10, 2013 from http://140.203.154.209/~vit/resources/2010/pubs/phd-thesis.pdf.

Othman, S. H. & Beydoun, G. (2013). Model-driven disaster management, *Information & Management* (50), 218-228

Öhgren, A., & Sandkuhl, K. (2005). Towards a methodology for ontology development in small and medium-sized enterprises. In *IADIS AC* (369-376).

Perry, D. & Gurenko, Z. (2007). Beyond lending: How multilateral banks can help Developing countries manage volatility - Chapter 13: Dealing with Natural Disaster Risks. Center for Global Development

Perry, M. & Herring, J. (2012). *OGC GeoSPARQL - A geographic query language for RDF data*, Open Geospatial Consortium.

Pinto, H. S., & Martins, J. P. (2004). Ontologies: How can they be built?. *Knowledge and Information Systems*, *6*(4), 441-464.

Quan-Li X., Kun, Y., Jun, W., Shuang-Yun, P. & Jun-Hua, Y. (2008). A study on ontology-driven geospatial- information retrieval systems in the semantic web, *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. (37), 733-738.

Raskin, R. G., & Pan, J. M. (2005). Knowledge representation in the semantic web for earth and environmental terminology (SWEET). *Computers & Geosciences* (31), 1119–1125.

Raskin, V., Triezenberg, K., Malaia, E., & Krachina, O. (2005). *Ontological semantic support for a specific domain.* CERIAS, Purdue University, USA.

Smart, P. D., Abdelmoty, A. I., El-Geresy, B. A., & Jones, C. B. (2007). A framework for combining rules and geo-ontologies. *In Web reasoning and rule systems* (133-147). Springer Berlin Heidelberg.

Smart, P., Abdelmoty, A. I., & Jones, C. B. (2004). An evaluation of geo-ontology representation languages for supporting web retrieval of geographical information.

Sotoodeh, M., & Kruchten, P. (2008). An ontological approach to conceptual modeling of disaster management. In *Systems Conference, 2008 2nd Annual IEEE* (1-4). IEEE.

SWRL. (2009). Environmental science and information application technology, ESIAT 2009. (2-3), 77-82.

Tanasescu, V., Gugliotta, A., Domingue, J., Gutiérrez Villarías, L., Davies, R., Rowlatt, M., & Richardson, M. (2006). Spatial integration of semantic web services: The e-Merges Approach.

Tanasescu, V., Gugliotta, A., Domingue, J., Davies, R., Gutierrez, L., Villarias-Gutierrez, L., et al. (2006). A semantic web GIS based emergency management system. *Lecture Notes in Computer Science*, *4273* (3), 959-966.

Tecim, V. (2009). Using geographical information technology for disaster management. *Disaster Risk Reduction in İzmir Symposium 2009, İzmir Turkey.*

Tian, Y., & Huang, M. (2012). Enhance discovery and retrieval of geospatial data using SOA and semantic web technologies. *Expert Systems with Applications*, (39-16), 12522-12535.

Truptil, S., Benaben, F., Couget, P., Lauras, M., Chapurlat, V., & Pingaud, H. (2008). Interoperability of information systems in crisis management: Crisis modeling and metamodeling. *Enterprise Interoperability III New Challenges and Industrial Approaches. Springer*. London. (583-594), Springer London.

Uschold, M., & King, M. (1995). *Towards a methodology for building ontologies* (15-30). Artificial Intelligence Applications Institute, University of Edinburgh.

Uschold, M. (1996). Building ontologies: Towards a unified methodology.*Technıcal Report-Unıversıty of Edınburgh Artıfıcıal Intellıgence Applıcatıons Instıtute Aıaı Tr*.

United States Geological Survey , USGS, (2012). *Glossary of terms on earthquake maps*. Retrieved October 11, 2013 from http://earthquake.usgs.gov/earthquake/glossary.php.

van Borkulo, E., Barbosa, V., Dilo, A., Zlatanova, S. & Scholten, H., (2006). Services for emergency response systems in the Netherlands. *In Second Symposium on Gi4DM*, Goa.

Wang, Y., Gong, J., & Wu, X. (2007). Geospatial semantic interoperability based on ontology. *Geo-spatial Information Science*, (10-3), 204-207.

Wisner, B., & Walker, P. (2005). *Beyond Kobe: An interpretative report on the world conference on disaster reduction, Kobe, Japan, 18–22 January 2005.* http://online.northumbria.ac.uk/geography_research/radix/resources/beyond_kobe_for_websites_july2005.pdf

Xu, W., & Zlatanova, S. (2007). Ontologies for disaster management response. In *Geomatics Solutions for Disaster Management* (185-200). Springer Berlin Heidelberg.

Yang, Z., Lin, G., & Chen, J. (2011). The domain ontology of typhoon disasters and its application. In *Geoinformatics, 2011 19th International Conference on* (1-5). IEEE.

Yu, L. (2011). A Web of data: Toward the idea of the semantic web, *A Developer's Guide to the Semantic Web*, Springer-Verlag Berlin Heidelberg, 1-18.

Yusuf, J. C. M., Su'ud, M. M., Boursier, P., & Alam, M. (2012). Extensive overview of an ontology-based architecture for accessing multi-format information for disaster management. In *Information Retrieval & Knowledge Management (CAMP), 2012 International Conference on* (294-299). IEEE.

Zhang, C., Zhao, T., & Li, W. (2010). Automatic search of geospatial features for disaster and emergency management. *International Journal of Applied Earth Observation and Geoinformation*, (12-6), 409-418.

Zhang, C., Zhao, T., Li, W. & Osleeb, J. P. (2010). Towards logic-based geospatial feature discovery and integration using web feature and geospatial semantic web, *International Journal of Geographical Information Science*, (24-6), 903-923.

Zhang, C., Li W., & Zhao T., (2010). Geospatial data sharing based on geospatial semantic web technologies, *Journal of Spatial Science*, (52-2), 35-49.

Zaharia, R., Vasiliu, L., Hoffman, J., & Klien, E. (2008). Semantic execution meets geospatial web services: A pilot application. *Transactions in GIS*, (12-1), 59-73.

Zheng Y. L., He, Q. Y., Qian, P., & Li, Z. (2012). Construction of the ontology-based agricultural knowledge management system. *Journal of Integrative Agriculture*, (11-5), 700-709.

Zlatanova, S., van Oosterom, P. & Verbree, E. (2006). Geo-information supports management of urban disasters. *Open House International*. (31-1), 62-79.