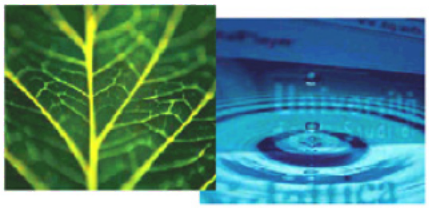
PhD Dissertation



International Doctorate School in Information and Communication Technologies

DISI - University of Trento

Integration of SDI Services: an evaluation of a distributed semantic matching framework.

Lorenzino Vaccari

Advisor:

Prof. Maurizio Marchese

Università degli Studi di Trento

Co-Advisor:

Prof. Fausto Giunchiglia

Università degli Studi di Trento

April 2009

Abstract

Access to geographic information has radically changed in the past decade. Previously, it was a specific task, for which complex desktop Geographic Information Systems (GISs) were built, and geographic data was maintained locally, managed by a restricted number of technicians. With the significant impact of the world-wide-web, an increasing number of different geographic services became available from heterogeneous sources. To support interoperability among different providers and users, GIS agencies have started to adopt Spatial Data Infrastructure (SDI) models.

Usually, each SDI service provider publishes and gathers geographic information based on its background knowledge. Hence, discovering, chaining, and using services require a semantic interoperability level between different providers. This problem is typically referred as the need for "semantic interoperability among autonomous and heterogeneous systems" and it is a challenge for current SDIs, due to their distributed architecture.

This thesis provides a framework to approach the semantic heterogeneity problem in the field of geo-services - services that deal with the generation and management of geographical information - among distributed SDIs. The framework is based on: (i) a peer-to-peer (P2P) view of the semantics of web service coordination, implemented by using the OpenKnowledge system and (ii) the use of a specific semantic matching solution called Structure Preserving Semantic Matching (SPSM). SPSM is a basic module of OpenKnowledge as it enables web service discovery and integration by using semantic matching between invocations of web services and web service descriptions.

We applied the OpenKnowledge system on a realistic emergency response scenario and selected SDI services. We modeled an emergency response scenario, i.e., a potential flooding event in the area of Trento. The scenario is based on the past experience and actual emergency plans as collected from interviews with personnel of the involved institutions and from related documents. Within this emergency response scenario a detailed implementation of selected SDI services is presented, namely the gazetteer, map and download services.

The SPSM solution has been assessed on a set of GIS ESRI ArcWeb services. Two kinds of experiments have been conducted: the first experiment includes matching of original web service signatures with synthetically altered ones. In the second experiment a manual classification of the GIS dataset has been compared to the unsupervised one produced by SPSM. The evaluation results demonstrate robustness and good performance of the SPSM approach on a large (ca. 700.000) number of matching tasks. In the first experiment a high overall matching relevance quality (F-measure) was obtained (over 55%*). In the second experiment the best F-measure value exceeded* 50% *for the given GIS operations set. SPSM performance is good, since the average execution time per matching task was 43 ms. This suggests that SPSM could be employed to find similar web service implementations at runtime. The aforementioned results suggest the practical real time application of the SPSM approach to: (i) discovering geo-services from specific geographic information catalogs, (ii) composing specific geo-processing services, (iii) supporting coordination of geo-sensor networks, and (iv) supporting geo-data discovering and integration.*

Keywords

Semantic heterogeneity, Spatial Data Infrastructure integration, geo-web services, ontology matching, ontology matching evaluation.

Wishes

Acknowledgments

I believe that a lot of very important persons have given me support and suggestions in many aspects of my life. This thesis is a result I could have never achieved without their care.

I wish to express my sincere gratitude to my supervisor Maurizio Marchese for his valuable guidance, meticulous supervision, constructive comments, and, above all, constant encouragement. I am grateful to Fausto Giunchiglia who has improved the vision of my research topic and has helped me on how to do *sound* and *complete* research. I also express my gratitude to the members of the external thesis committee, Dave Robertson and Marco Maggini, for their very useful feedback and the efforts they have invested in reviewing my thesis.

I thank my fellows and friends who have contributed to this work. I especially would like to express my appreciation to Pavel Shvaiko without whom this study could not have been completed. I would like to make special mention of Alexander Ivanyukovich, Juan Pane, Veronica Rizzi, Gaia Trecarichi and Michele Vescovi who have given me a lot of hints during my PhD studies.

Then, I would like to express my estimation to all the people involved in the EU OpenKnowledge project and especially to Paolo Besana, Fiona McNeill, and Dave Dupplaw for many productive discussions we had during the progress of the project.

I am also grateful to the Autonomous Province of Trento who partially

funded my PhD research activity, and particularly to Fabio Scalet and Paola Matonti who gave me the possibility to pursue both my professional and my research activities.

I would like to express appreciation for all my friends who, even though I often dedicate my free time to academic studies and research activities, have preserved their highly valuable friendship. I thank my parents, Teresa and Ivano, my sister Mara, my brothers Stefano and Loris, my nephew Nicolas, my niece Julia, my sisters in law Katya and Alessandra, my mother in law Mirella and my father in law Gastone, who have given me their love, patience, and understanding over all of these years.

I am truly grateful for the most important person in my life, my loving wife Katya, for her love and moral strength she unconditionally always gives me. She constantly supports me to face all the challenges in my life.

Contributions and publications

This work has been developed in collaboration with various people (as the publications indicate) and in particular with: Maurizio Marchese, Fausto Giunchiglia, Pavel Shvaiko, Paolo Besana, Gaia Trecarichi, Juan Pane, Veronica Rizzi, Fiona McNeill, Nardine Osman, and Alexander Ivanyukovich.

This thesis makes the following contributions:

- The presentation of the Spatial Data Infrastructure (SDI) phenomenon and motivation behind its adoption.
- An overview of semantic heterogeneity issues in SDIs both on geographic data and geographic services.
- A detailed survey of state of the art approaches and systems to solve the semantic heterogeneity problem among distributed geographic data sources and service providers.
- The development of a realistic emergency response scenario based on the organizational model of Trento, Italy.
- The analysis, within the aforementioned emergency response scenario, of selected SDI services, namely the gazetteer, map and download services.
- The formalization and the implementation of the gazetteer, map, and download services with the OpenKnowledge P2P-based system.

- The extensive evaluation of the SPSM approach to semantic heterogeneity problem between distributed geo-services. Specifically, we:
  - Built an evaluation dataset based on the signatures of a real world set of GIS ESRI ArcWeb services[1].
  - Developed a methodology to evaluate both the robustness and the classification capabilities of the SPSM approach.
  - Applied the aforementioned methodology to the GIS dataset and discussed the SPSM results.

Part of the material of the thesis has been published as articles in various conferences and journals and as technical reports in the EU FP6 OpenKnowledge[2] Specific European Targeted Research Project (STREP) project IST-FP11V341. In what follows, journal articles, conference papers, and technical reports are listed.

Journal articles (in order of appearance):

- [144]: Lorenzino Vaccari, Pavel Shvaiko, and Maurizio Marchese. A geo-service semantic integration in spatial data infrastructures. *International Journal of Spatial Data Infrastructures Research (IJSDIR)*, Volume 4(2009), pages 24-51, 2009.
- [82]: Maurizio Marchese, Lorenzino Vaccari, Gaia Trecarichi, Nardine Osman, Fiona McNeill, and Paolo Besana. An Interaction-Centric Approach to Support Peer Coordination in Distributed Emergency Response Management, *Intelligent Decision Technologies (IDT), Special Issue on Incident Management*, 2009, to appear.
- [142]: Lorenzino Vaccari, Pavel Shvaiko, Paolo Besana, Maurizio Marchese, and Juan Pane. An evaluation of ontology matching in geo-service applications. *Submitted to GeoInformatica*, 2009.

[1] http://www.esri.com/software/arcwebservices/

[2] http://www.openk.org/

Conference papers (in order of appearance):

- [134]: Gaia Trecarichi, Veronica Rizzi, Lorenzino Vaccari, Maurizio Marchese, and Paolo Besana. OpenKnowledge at work: exploring centralized and decentralized information gathering in emergency contexts. *In Proceeding of the 6th International Conference on Information Systems for Crisis Response and Management (ISCRAM)*, 2009.

- [140]: Lorenzino Vaccari and Juan Pane. Ontology matching evaluation using GIS web services. *In Proceedings of the Workshop on Matching and Meaning: automated development, evolution and interpretation of ontologies at the Conference on Adaptive & Emergent Behaviour & Complex Systems (AISB 09)*, 2009, to appear.

- [143]: Lorenzino Vaccari, Pavel Shvaiko, and Maurizio Marchese. An emergent semantics approach to semantic integration of geo-services and geo-metadata in spatial data infrastructures. *In Proceedings of the 10th Global Spatial Data Infrastructure (GSDI) Conference*, 2008.

- [81]: Maurizio Marchese, Lorenzino Vaccari, Gaia Trecarichi, Nardine Osman, and Fiona McNeill. Interaction models to support peer coordination in crisis management. *In Proceedings of the 5th International Conference on Information Systems for Crisis Response and Management (ISCRAM)*, pages 230-241, 2008.

- [80]: Maurizio Marchese, Lorenzino Vaccari, Pavel Shvaiko, and Juan Pane. An application of approximate ontology matching in eResponse. *In Proceedings of the 5th International Conference on Information Systems for Crisis Response and Management (ISCRAM)*, pages 294-304, 2008.

- [79]: Maurizio Marchese, Ivanyukovich Alexander, and Lorenzino Vaccari. A web service approach to geographical data distribution among

public administrations. *In Proceedings of the 5th IFIP Conference on e-Commerce, e-Business, and e-Government (I3E)*, volume 189, pages 329-343, 2005.

- [138]: Lorenzino Vaccari, Maurizio Marchese, and Alexander Ivanyukovich. A service oriented approach for geographical data sharing. *In Proceedings of the Workshop 11th EC GI & GIS Workshop: European Space Data Infrastructure: Setting the framework*, pages 134-136, 2005.

Technical reports (in order of appearance):

- [135]: Gaia Trecarichi, Veronica Rizzi, Lorenzino Vaccari, Juan Pane, and Maurizio Marchese. OpenKnowledge deliverable 6.8: Summative report on use of OpenKnowledge approach in eResponse: integration and evaluation results. *Technical report, OpenKnowledge*, 2008.

- [12]: Paolo Besana, Fiona McNeill, Fausto Giunchiglia, Lorenzino Vaccari, Gaia Trecarichi, and Juan Pane. Web service integration via matching of interaction specifications. *Technical report, University of Trento, Dipartimento di Ingegneria e Scienza dell'Informazione (DISI)*, 2008.

- [83]: Maurizio Marchese, Lorenzino Vaccari, Gaia Trecarichi, Pavel Shvaiko, Juan Pane, Nardine Osman, and Fiona McNeill. OpenKnowledge deliverable 6.7: Interaction models for eResponse. *Technical report, OpenKnowledge*, 2008.

- [141]: Lorenzino Vaccari, Juan Pane, Pavel Shvaiko, Maurizio Marchese, Fausto Giunchiglia, Paolo Besana, and Fiona McNeill. OpenKnowledge deliverable 3.7: Summative report on matching implementation and benchmarking results. *Technical report, OpenKnowledge*, 2008.

- [139]: Lorenzino Vaccari, Maurizio Marchese, and Pavel Shvaiko. OpenKnowledge deliverable 6.6: Emergency response GIS service cluster. *Technical report, OpenKnowledge*, 2007.

- [137]: Lorenzino Vaccari, Maurizio Marchese, Fausto Giunchiglia, Fiona McNeill, Stephen Potter, and Austin Tate. OpenKnowledge deliverable 6.5: Emergency monitoring scenarios. *Technical report, OpenKnowledge*, 2006.

In preparation:

- [3]: George Anadiotis, Paolo Besana, David de la Cruz, Dave Dupplaw, Frank van Harmelen, Spyros Kotoulas, Juan Pane, Adrian Perreau de Pinninck, Marco Schorlemmer, Ronny Siebes, and Lorenzino Vaccari. The OpenKnowledge system: an interaction-centered approach to knowledge sharing. *In preparation*, 2009.

Whenever results of any of these works are reported, proper citations are made in the body of the thesis.

Contents

Introduction **1**

I Interoperability in Spatial Data Infrastructures **9**

1 The SDI phenomenon **11**

1.1 The SDI motivation 12

1.2 SDI definition and architecture 21

1.3 Summary 25

2 Information systems' interoperability **27**

2.1 Dimensions of interoperability 28

2.2 Ontologies 30

2.3 Semantic heterogeneity of geo-information 32

2.4 Summary 39

II State of the art **41**

3 Semantic heterogeneity in geo-information **43**

3.1 Geo-data semantic integration 45
3.2 Geo-service semantic integration 50
3.3 Ontology matching 53
3.4 Summary 55

4 P2P architectures in GIS applications 57
4.1 P2P model 57
4.2 GIS P2P applications 59
4.3 Summary 63

III A P2P semantic matching framework 65

5 Motivating scenario 67
5.1 e-Response scenario 68
5.2 Evacuation use case 77
5.3 SDI service coordination 82
5.4 Summary 87

6 Supporting the scenario: the OpenKnowledge system 89
6.1 The context 91
6.2 Lightweight Coordination Calculus 93
6.3 Model of the system 96
6.4 Architecture of the system 98

6.5 The SPSM implementation 102
6.6 Summary 108

7 SDI services implementation 109
7.1 The OKCs architecture 109
7.2 The gazetteer service 112
7.3 The map service 118
7.4 The download service 124
7.5 The emergency GUI. 129
7.6 Summary 131

IV SPSM Evaluation 133

8 The GIS web service evaluation dataset 135
8.1 Evaluation dataset 136
8.2 Evolution experiment setup 138
8.3 Classification experiment setup 141
8.4 Summary 142

9 Evaluation method 145
9.1 Evaluation measures 145
9.2 Evolution experiment: the evaluation method 146
9.3 Classification experiment: the evaluation method 150
9.4 Number of matching tasks 153
9.5 Summary 154

10 Evaluation results **155**

10.1 Evolution experiment: the results 155

10.2 Classification experiment: the results 167

10.3 Performance evaluation 168

10.4 Evaluation summary 169

10.5 Summary 170

V Conclusions **173**

11 Summary and future work **175**

11.1 Dimensions of interoperability 177

11.2 Application scenarios 179

11.3 Future work 183

Bibliography **185**

List of Figures

1.1 Spatial Data Infrastructure technological implementation. . 23
1.2 Spatial Data Infrastructure components. 25

2.1 Service Oriented Architecture. 37

5.1 Flooding areas, Trento town 70
5.2 Organizational schema of emergency coordination in the Province of Trento 73
5.3 The overall e-Response use case 79
5.4 Overall Architecture for Map Request Service 84
5.5 Activity Diagram for the Gazetteer Service 85
5.6 Activity Diagram for the Map Request Service 86
5.7 Activity Diagram for the Download Request Service 87

6.1 Abstract syntax of LCC. 94
6.2 LCC example: double arrows ($\Rightarrow, \Leftarrow$) indicate message passing, single arrow ($\leftarrow$) indicates constraint satisfaction. . . . 95
6.3 OpenKnowledge model 99
6.4 OpenKnowledge kernel architecture 101
6.5 Two web service descriptions (trees) and correspondences (lines) between them. 103

7.1 OK enabled SDI services 110
7.2 Gazetteer service sequence diagram. 113

7.3 Gazetteer requestor IM 114
7.4 Gazetteer requestor IM 115
7.5 Java class diagrams of gazetteer service. 116
7.6 Map request service. 118
7.7 LCC fragment for the GIS agency service requestor role. . 120
7.8 LCC fragment for the GIS agency service provider role. . . 120
7.9 Java class diagrams of map service. 122
7.10 Sequence diagram of the download service. 124
7.11 Download service requestor role IM. 126
7.12 Download service provider role IM. 127
7.13 Java class diagrams of map service. 128
7.14 e-Response visualizer. 130

10.1 Recall of *Replace a node name with an unrelated one* alteration operation. 156
10.2 Precision of *Replace a node name with an unrelated one* alteration operation. 157
10.3 F-measure of *Replace a node name with an unrelated one* alteration operation. 158
10.4 Recall of *Add or remove a label in a node name* alteration operation. 159
10.5 Precision of *Add or remove a label in a node name* alteration operation. 160
10.6 F-measure of *Add or remove a label in a node name* alteration operation. 160
10.7 Recall of *Alter syntactically a label* alteration operation. . 161
10.8 Precision of *Alter syntactically a label* alteration operation. 162
10.9 F-measure of *Alter syntactically a label* alteration operation. 162

10.10Recall of *Replace a label in a node name with a related one* alteration operation. 163
10.11F-measure of *Replace a label in a node name with a related one* alteration operation. 164
10.12SPSM vs. baseline on *Replace a label with an unrelated one* alteration operation. 165
10.13SPSM vs. baseline on *Replace a label in a node name with a related one* alteration operation. 166
10.14F-measure values for SPSM matcher. 167
10.15F-measure values for edit-distance (baseline) matcher. . . . 168
10.16Recall, precision and F-measure values for the classification experiment. 168

11.1 WPS services catalog. Courtesy of Fondazione Graphitech, Trento, Italy. 181

List of Tables

6.1 The correspondence between abstraction operations, tree edit operations and costs. 107

8.1 Reference alignment of ArcWeb services. 142
8.2 Summative statistics for the test cases. 142

9.1 Example of quality measures results. 150

Introduction

Interoperability in Spatial Data Infrastructures

Geographic information has radically changed in the past decade. With the significant impact of the world-wide-web an increasing number of different geographic services have become available from different sources. Google Maps[3], Microsoft Live Search Maps[4] and Yahoo Local Maps[5], for example, introduce Geographic Information System (GIS) services to ordinary Internet users with aerial imagery and with responsive performance. Geographic information systems, sensor systems, automated mapping, facilities management, traffic analysis, geo-positioning systems, and other technologies for geospatial information are entering a period of radical integration [102]. Meanwhile, different service providers are publishing numerous APIs, enabling light-weight integration of geographical data into any web pages and applications. This is a clear indication that a broader wave of GIS Web services applications is on its way. However, this growing field presents interoperability problems.

Technically, web service technologies have provided the necessary standards for applications in different domains to get integrated with GIS data and services. In this context, to support interoperability among different providers and users, the Open Geospatial Consortium (OGC) has success-

[3] http://maps.google.it/

[4] http://maps.live.it/

[5] http://maps.yahoo.com/

fully published GIS interoperability specifications which support service providers in integrating different online geo-processing and services. Moreover, GIS agencies have started to adopt Spatial Data Infrastructure (SDI) models [10, 53, 96, 85]. SDIs goals differ from the previous ones, adopted by GISs. While a GIS is a self-contained system in which data and software applications are used mainly internally, the main aim of an SDI is to support the interoperability among different kinds of geographical information providers and users.

The semantic heterogeneity problem

Usually, different SDIs represent heterogeneous information systems, thus, among interoperable SDIs, three fundamental dimensions of information systems have to be taken into account: *distribution, autonomy* and *heterogeneity* [122]. In particular, there have been many different classifications to types of heterogeneity [8, 122, 14, 65, 148, 66, 32] and the most obvious types of heterogeneity that pervade information systems are at system, syntax, structure, and semantics levels [120, 123].

While system, syntax, and structural heterogeneity have addressed many issues by increasing standardization and interoperability, nowadays the key challenges to be faced are at semantic level [71]. Previously, the management of geographic information was a specific task, for which complex and autonomous desktop GISs were built and geographic data were maintained locally, managed by a restricted number of specialized technicians. Thus, each organization maintained its local domain vocabulary of terms related to geographic features and relations among them. Also, GIS services and applications were specifically developed to perform internal requirements, with little or no interactions with other GIS providers and users. These systems were often based on locally defined semantics, sometimes even

explicitly encoded, though not shared with other stakeholders.

Hence, discovering, chaining, and using services require a semantic interoperability level between different providers and users, where services understand each other information. This problem is typically referred to as the need for *semantic interoperability among autonomous and heterogeneous systems* and it is an actual challenge for current SDIs, due to their distributed architecture.

Solution proposed

This thesis presents and evaluates a framework to approach the semantic heterogeneity problem in the field of integration of geographic services among distributed SDIs. The framework is based on two main keystones:

- A peer-to-peer (P2P) view of the semantics of web service coordination, implemented by using the OpenKnowledge system and the Lightweight Coordination Calculus (LCC) language. Technically, in the OpenKnowledge system, the peers, namely service providers and service requestors, share explicit knowledge of the interactions in which their services are engaged. These models of interaction are used operationally as the anchor for describing the semantics of the interaction. During each interaction web service discovery and integration are requested, for example, to query information systems, to perform data elaboration on retrieved data, to visualize results, etc.

- The use, on a set of GIS web services, of a specific semantic matching solution called *structure preserving semantic matching* (SPSM) which is implemented by a particular OpenKnowledge module. In our scenario, since there is no a priori semantic agreement (other than the models of interation), the semantic matching module is needed during

interactions to automatically make semantic commitments between the invocation of the services and peers web service descriptions.

The main contributions of this thesis are (*i*) the application of the OpenKnowledge P2P system to a set of real world SDI services, and (*ii*) the extensive evaluation of the proposed framework, based on a semantic matching approach between distributed web services, on real world GIS web services. Specifically:

- We present the SDI phenomenon and motivation behind its adoption, we give an overview of semantic heterogeneity issues in SDIs both on geographic data and geographic services, and we perform a detailed survey of state of the art approaches and systems to solve the semantic heterogeneity problem among distributed geographic data sources, service providers and users.

- We illustrate an emergency response (e-Response) overall scenario which we implemented as a testbed of the OpenKnowledge P2P system. As an example of the e-Response scenario, we propose a simulated flooding scenario in Trento, as collected from interviews of the involved institutions personnel and from related documents.

- Within the e-Response scenario, we select, describe, analyze, and formalize specific SDI services, namely the map, gazetteer, and download services. We show how these services are implemented in the OpenKnowledge system by means of the LCC language.

- We assess the semantic matching SPSM solution with the real world GIS ESRI ArcWeb services. We describe the setup, we show the method and we discuss the results of two kinds of experiments. In the first experiment we match, by using SPSM, original GIS web service operation signatures to synthetically altered ones. In the second

experiment we compare a manual classification of GIS web service operations to the unsupervised one produced by SPSM. The evaluation results demonstrate robustness and good performance of the SPSM approach on a large (ca. 700.000) number of matching tasks.

- We summarize the applicability of the proposed approach to a number of fields including: (*i*) GIS web service discovering on geo-service catalogs, (*ii*) composition of GIS processing services, and (*iii*) geo-sensor networks supporting.

Note that some lines of work on the topic of this thesis have been supported by the OpenKnowledge project, and by the Autonomous Province of Trento.

Structure of the thesis

The thesis is organized in five parts.

Part one describes the interoperability issues between distributed and heterogeneous SDIs. SDIs' motivation, definition and architecture are given in Chapter 1. In Chapter 2, we first present the dimensions of interoperability among distributed information systems. Then, we focus on semantic heterogeneity issues of geo-information.

Part two provides a comprehensive coverage of approaches used to solve heterogeneity problems among distributed geographic systems. Specifically, Chapter 3 illustrates state of the art approaches and systems that attempt to reduce the semantic heterogeneity problem both for geo-data and geo-services. Moreover, it analyzes some solutions for the ontology matching problem and related evaluation methods. Chapter 4 presents an overall discussion on the P2P model along with some of its recent applications to the GIS field.

Part three provides the motivation scenario, the description of the distributed semantic matching framework proposed in this thesis, and its application when coordinating the SDI services within the motivation scenario. In particular, Chapter 5 delineates the natural disaster scenario (flooding) geographical services which we adopt as a testbed for the OpenKnowledge system, focusses on the people evacuation use case, and analyzes SDI services coordination which supports the emergency activities. Chapter 6 introduces basic notions about the OpenKnowledge system - which we consider as a novel P2P view of the semantics of web service coordination - and describes how the SPSM approach enables web service discovery and integration by using semantic matching between invocations of web services and web service descriptions. Chapter 7 presents the application of the OpenKnowledge system to the coordination of the aforementioned SDI services, namely the gazetteer, map and download services.

Part four presents the evaluation of the SPSM solution, which we applied within an e-Response scenario for geographic service coordination. Specifically, we evaluate the SPSM solution on real world GIS ESRI ArcWeb services by conducting two experiments. The first experiment included matching of original web service signatures to synthetically altered ones. In the second experiment we compared a manual classification of our dataset to the unsupervised one produced by SPSM.

In Chapters 8, 9, and 10 we present, respectively, the evaluation dataset, the evaluation methodology, and the evaluation results of the aforementioned experiments. In the former experiment a high overall matching relevance quality (F-measure) was obtained (over 50%). Moreover, a comparison to a baseline matcher showed how the SPSM approach is always better (in average by 20%) when semantic alterations are introduced. In the second experiment the best F-measure values exceeded 50% for the given GIS operations set. SPSM performance is good, since the average

execution time per matching task was 43 ms. That opens to the possibility of real time execution. The evaluation results demonstrate robustness and good performance of the SPSM approach on a large (ca. 700.000) number of matching tasks.

Finally, part five concludes. Chapter 11 summarizes the work done in this thesis, describes how our approach fulfill the dimensions of interoperability, and outlines application scenarios and future work.

Part I

Interoperability in Spatial Data Infrastructures

Chapter 1

The SDI phenomenon

Since its introduction in the 1960s, GIS has become useful and almost indispensable instrument in a vast range of applications. From urban planning to civilian protection, from environmental protection to agriculture assessment and so on. Geographic information was, until ten years ago, managed autonomously and aimed to specific tasks by GIS agencies, whose majority were affiliated to governmental institutions.

Then, the possibility to share information, by adopting distributed systems over the Internet infrastructure, opened new scenarios in which the geographical information became the paradigm of a new vision called *Digital Earth*. According to this vision geographic data are now available and exploitable also by common users, that can easily use digital services to query and obtain significant and precise geographic information.

Nowadays, the majority of existing geographic information systems publish their data and services in a centralized way, but there is an increasing necessity to share this information between different and heterogenous providers and users, and the challenge is to obtain interoperability between heterogeneous geographic systems.

In this chapter we first present (§1.1) the requirements and the scenarios of interoperability between geographical infrastructures that support the paradigm of distributed geographical information system, namely Spatial Data Infrastructures (SDIs). Then we discuss definition and architecture of current SDIs (§1.2).

1.1 The SDI motivation

The domain of geographic information[1] is experiencing a rapid growth of both computational power and quantity of information, making large spatial data archives available over the Internet.

A visionary concept of the integration of geo-information was posed on 1998 by the U.S. vice president Al Gore [52]. His *Digital Earth* label became popular for describing a virtual representation of the Earth on the Internet that is spatially referenced and interconnected with the world's digital knowledge archives. One of the issues tacked in his speech, given at the California Science Center, was that we have more information than we know what to do with. Part of the problem has to do with the way information is displayed. The tools we have most commonly used to interact with data, such as the *desktop metaphor* employed by the Macintosh, Linux and Windows operating systems, are not really suited to this new challenge. Al Gore's believed that we need a Digital Earth. A multiresolution, three dimensional representation of the planet, into which we can embed vast quantities of geo-referenced data. Al Gore's example was about a young child going to a Digital Earth exhibit at a local museum that, using a special human computer interface, could explore a virtual world both moving through space and traveling through time.

[1] In this thesis, we will use the term *geographic information* to group different kinds of geographic objects: geographic services or geo-services, geographic metadata or geo-metadata and geographic data or geo-data.

Al Gore identified main technologies and capabilities that would be required to build a Digital Earth, including:

Computational science. With high speed computers as a new tool we can simulate phenomena that are impossible to observe, and simultaneously better understand data from observations. Computational science allows us to overcome the limitations of both experimental and theoretical science.

Mass storage. The Digital Earth requires storing quadrillions of bytes of information. E.g., NASAs missions to Planet Earth program generated a terabyte of information each day.

Satellite images. The Digital Earth needs a level of accuracy sufficient for detailed maps. Nowadays, commercial satellite systems provide very high resolution imagery. E.g., QuickBird is a high resolution satellite and collects image data to 0.61m pixel resolution degree of detail[2], while IKONOS high-resolution satellite capabilities include capturing a 3.2m multispectral, Near-Infrared (NIR)/0.82m panchromatic resolution images at nadir[3].

Broadband networks. The data needs for a digital globe are maintained by thousands of different organizations. That means that the servers that are participating in the Digital Earth need to be connected by high speed networks.

Interoperability. The Internet and the World Wide Web have succeeded because of the emergence of a few, simple widely agreed upon protocols, such as the Internet protocol. The Digital Earth also needs some level of interoperability, so that geographical information generated by one kind of application software can be read by another.

[2] http://www.satimagingcorp.com/satellite-sensors/quickbird.html

[3] http://www.satimagingcorp.com/satellite-sensors/ikonos.html

Metadata. Metadata is *data about data*. For imagery or other geo-referenced information to be helpful, it may be necessary to know its name, location, author of source, date, data format, resolution, etc.

Al Gore not only described some potential applications of the Digital Earth project (conducting virtual diplomacy, fighting crime, preserving biodiversity, predicting climate change, increasing agricultural productivity, and responding to manmade or natural disaster), but also he dealt with *The Way forward*, a summary of the main points useful to build the project: a 3D user interface, a distributed and interoperable system, a development of prototypes to test potential applications and technologies, an integration of available multiple resources, a development of a digital map of the world at 1 meter resolution.

In order to put the Al Gore' visions into practice, some preliminary initiatives were born [39]. In 1999, the US Digital Earth initiative was cooperatively defined by the creation of an interagency working group known as the *Interagency Digital Earth Working Group* (IDEWG), led by NASA. IDEWG comprised 17 federal agencies with guest advisors from industry and academia. A task force was established to focus on the several sectors which defined the early concepts for Digital Earth as follows:

- Visualization and exploration.
- Education and outreach.
- Science and applications.
- Advanced display sites.
- Data access and distribution.
- Standard and architectures.

In March of 2000, industry representatives showcased for the IDEWG over a dozen enterprizing technologies which demonstrated promising 3D visualization prototypes. Within two years, these prototypes were captivating international audiences in government, business, science, and mass media who began to purchase the early commercial geo-browsers, and hence became symbolic precursors for the Digital Earth initiative and the development of new technologies (e.g., the *International Digital Earth* SRI project) [73].

NASA's leadership for Digital Earth had waned by 2001 owing to a change in the US administration and therein disbanded back into multiple agency internal activities. As of this writing, Digital Earth initiatives in the US Government are limited primarily to NASA's World Wind [4] and Earth Observatory programs[5], and NOAA's Science on a Sphere[6]. There exist, however, many enthusiastic government supporters of the Digital Earth framework and vision, if not the name, and rapid growth can be expected in the immediate future with the greater public awareness of Digital Earth through the success of Google Earth[7].

International collaboration for the Digital Earth concept has been led by the Chinese Academy of Sciences' (CAS) Institute for Remote Sensing Applications. CAS sponsored and hosted the Beijing meeting of the 1st International Digital Earth Symposium in November 1999. A network of agencies and citizens is harmonizing efforts to capture the progress of Digital Earth technology for sustainable development throughout Asia and Europe. A host of Digital Earth workshops are being conducted throughout China, Asia, and the Pacific on a continuing basis, however, due to language barriers much of this progress has not been recognized internation-

[4] http://worldwind.arc.nasa.gov/

[5] http://earthobservatory.nasa.gov/

[6] http://sos.noaa.gov/

[7] http://earth.google.com/

ally. The formation of the International Society for Digital Earth (ISDE) was proposed and initiated by the CAS to create a non-profit entity to act as Secretariat for this growing international Digital Earth community[8]. Hosted by the CAS, this formal organization is now chartered to coordinate the implementation of the Beijing Declaration and ensure effective implementation of the bi-annual International Symposium series[9].

After ten years, the Al Gore's milestone vision is partially implemented [51, 54], by recently available geo-browsers (like Google Earth, Microsoft Virtual Earth, NASA Worldwind and ESRI ArcGIS Explorer) and web applications (like Google Maps, Microsoft Live Search Maps and Yahoo Local Maps). These systems have introduced GIS services to ordinary Internet users, offering them high-resolution aerial imagery with responsive performance [25, 136]. Moreover, there is an increasing necessity to share this information between different stakeholders (e.g., departments in public administration, professionals, citizens, and GIS expert users) and diverse information systems in order to enable a coherent and contextual use of geographic information.

This necessity forms the basis for a number of initiatives, to set up global, international, national and regional infrastructures for the collection and dissemination of geographical data, including among others:

- The **Shared Environmental Information System (SEIS)**[10] [23] is a collaborative initiative of the European Commission, the European Environmental Agency (EEA) and the member countries of the Agency. SEIS communication sets out an approach to modernize and simplify the collection, exchange, and use of the data and information required for the design and implementation of environmental policy,

[8] http://www.digitalearth-isde.org/

[9] http://www.isde6.org/

[10] http://ec.europa.eu/environment/seis/index.htm

according to which the current, mostly centralized systems for reporting are progressively replaced by systems based on access, sharing and interoperability. Ongoing activities at European, national and regional level, including INSPIRE[11], GEOSS[12], GMES[13], and WISE[14], need to be reinforced and coordinated in line with SEIS. Within the European Commission, priority will be given to the implementation of the INSPIRE directive and further development of the GMES initiative, as a basis for improving respectively the sharing of environment-related data and information within Europe and the provision of services to public policy makers and citizens. The success of both these activities in solving the problems they have been designed to address will be carefully monitored, along with the possible need to launch complementary initiatives. In this way, it will be ensured that SEIS, INSPIRE and GMES are mutually supportive. As noted above, a key step in implementing SEIS, and especially to trigger the expected simplification benefits, will be to modernize the legal provisions relating to way in which information required by environmental legislation is made available. It is expected that this will be done by revising the Standardized Reporting directive 91/692/EC, which needs to be updated and brought into line with the SEIS principles. To this end, the European Commission intends to come forward with a relevant legislative proposal, including a repeal of outdated provisions in the current standardized reporting directive.

- The **INfrastructure for SPatial InfoRmation in Europe (INSPIRE)** [62] is an European initiative, started in 2001 by the European Commission, based on the goal to improve the accessibility,

[11] http://inspire.jrc.ec.europa.eu/
[12] http://www.epa.gov/geoss/
[13] http://www.gmes.info/
[14] http://water.europa.eu/

interoperability and affordability of spatial data and information systems in Europe. INSPIRE lists among its main objectives: *Experience in the Member States has shown that it is important, for the successful implementation of an infrastructure for spatial information, that a minimum number of services be made available to the public free of charge. Member States should therefore make available, as a minimum and free of charge, the services for discovering and, subject to certain specific conditions, viewing spatial data sets.* The INSPIRE concept is ambitious. The initiative intends to trigger the creation of a European spatial information infrastructure that delivers to the users integrated spatial information services. These services should allow the users to identify and access spatial or geographical information from a wide range of sources, from the local level to the global level, in an interoperable way for a variety of uses. The target users of INSPIRE include policy-makers, planners, and managers at European, national and local level and the citizens and their organizations. Possible services are the visualization of information layers, overlay of information from different sources, spatial and temporal analysis, etc. INSPIRE should be based on the infrastructures for spatial information that are created by the Member States and that are made compatible with common implementing rules and are supplemented with measures at Community level. Implementing rules are being developed in the following areas: creation and updating of metadata, monitoring and reporting, discovery and view services, download services, coordinates transformation services, governing the access rights of use to spatial data sets and services for Community institutions and bodies, interoperability and harmonization of spatial data sets and services for Annex I spatial data themes. INSPIRE defines a roadmap until 2019.

- The **Global Earth Observation System of Systems (GEOSS)**[15] is being built by the Group on Earth Observations (GEO). GEOSS seeks to connect the producers of environmental data and decision-support tools with the end users of these products, with the aim of enhancing the relevance of Earth observations to global issues. The ultimate result is to provide a global public infrastructure that generates comprehensive, near-real-time environmental data, information and analyzes for a wide range of users. The GEOSS is simultaneously addressing nine areas of critical importance to people and society. It aims to empower the international community to protect itself against natural and human-induced disasters, understand the environmental sources of health hazards, manage energy resources, respond to climate change and its impacts, safeguard water resources, improve weather forecasts, manage ecosystems, promote sustainable agriculture and conserve biodiversity. GEOSS coordinates a multitude of complex and interrelated issues simultaneously. This cross-cutting approach avoids unnecessary duplication, encourages synergies between systems and ensures substantial economic, societal and environmental benefits.

- The **Global Monitoring for Environment and Security (GMES)**[16] is a joint initiative of the European Commission and European Space Agency, adopted by EU Heads of State at the Gothenburg Summit in 2001, and aimed at achieving an autonomous and operational capability in the exploitation of geo-spatial information services by 2008. GMES will be the European programme implementing an Earth observation service system with satellites, sensors on the ground, floating in the water or flying through the air to monitor our planet's environ-

[15] http://www.earthobservations.org/geoss.shtml
[16] http://ec.europa.eu/gmes/overview.htm

ment and to support the security of every citizen. The information provided by GMES will help us understand better how and in what way our planet may be changing, why this is happening, and how this might influence our daily lives. Besides affecting our daily lives, GMES will provide vital information to decision-makers and business operators that rely on strategic information with regard to environmental, for instance, climate change and adaptation, or security issues. The infrastructure needed to collect the observations used by GMES services is owned and operated either by international, European or national entities with their respective political and financial responsibilities. GMES aims at ensuring seamless data flow for sustainable services through effective coordination of all these capacities. GMES is the European Union contribution to GEOSS.

Moreover, a growing number of public institutions and private companies have adopted a GIS to handle their internal geographical information. A number of commercial and open source software packages are available to support such local activities (e.g., ESRI ArcGIS[17], Pitney Bowes MapInfo[18], Intergraph GeoMedia[19], GRASS[20], etc.). These products give a complete and powerful set of functionalities to manage geographical information for every GIS agency. Beside this management challenge, the growing number of geographic information providers, the large quantity of the produced GIS data, the availability of high speed networks and sophisticated computer science technologies have been creating a heterogeneous set of producers and final users. Typical user roles include:

- International, national and regional institutions that coordinate and integrate geographic information provided by different GIS agencies.

[17] http://www.esri.com/software/arcgis/

[18] http://www.mapinfo.com/

[19] http://www.intergraph.com/sgi/products/

[20] http://grass.itc.it/index.php

- Public institutions that require geographic information to support institutional duties (e.g., emergency, health, urban planning, and tourism).
- Research institutions that want to analyze the availability and the quality level of geographic information covering a specific study area.
- Private companies that need geographic information in order to create business services and products (geo-marketing).
- Non expert users that need to locate quickly and easily a geographical feature (e.g., address, location name, institution, and business activity).

1.2 SDI definition and architecture

To support all these kinds of initiatives, providers, users, and user's requests, GIS agencies have started to adopt a Spatial Data Infrastructure (SDI) model [10, 53, 96, 85]. While a GIS is a self-contained system in which data and software applications are used mainly internally for capturing, managing, integrating, manipulating, analyzing, and displaying data that is spatially referenced to the Earth, the SDI goal is to support the interoperability among different kinds of providers and users. A Spatial Data Infrastructure is *the means to assemble geographic information that describes the arrangement and attributes of features and phenomena on the Earth. The infrastructure includes the materials, technology, and people necessary to acquire, process, and distribute such information to meet a wide variety of needs* [95].

An exhaustive definition of SDI is also given in [96]: *The term SDI is often used to denote the relevant base collection of technologies, policies and institutional arrangements that facilitate the availability of and*

access to spatial data. The SDI provides a basis for spatial data discovery, evaluation, and application for users and providers within all levels of government, the commercial sector, the non-profit sector, academia and by citizens in general. The word infrastructure is used to promote the concept of a reliable, supporting environment, analogous to a road or telecommunications network, that, in this case, facilitates the access to geographically-related information using a minimum set of standard practices, protocols, and specifications. Like roads and wires, an SDI facilitates the conveyance of virtually unlimited packages of geographic information. An SDI must be more than a single data set or database; an SDI hosts geographic data and attributes, sufficient documentation (metadata), a means to discover, visualize, and evaluate the data (catalogs and web mapping), and some methods to provide access to the geographic data. Beyond this are additional services or software to support applications of the data. To make an SDI functional, it must also include the organizational agreements needed to coordinate and administer it on a local, regional, national, and transnational scale. The creation of specific organizations or programs for developing or overseeing the development of SDI, particularly by government at various scales can be seen as the logical extension of the long practice of coordinating the building of other infrastructures necessary for ongoing development, such as transportation or telecommunication networks.

From the technological point of view it is difficult to define a precise architecture of an SDI, because of the continue technological evolution and of the different solutions adopted to implement a logical system architecture into a physical multi-tier architecture [102]. Figure 1.1 presents an example of a logical service-oriented architecture of an SDI within a generic GIS agency. Main services are represented in this figure, as a summary of [102] and [96].

Logical architecture identifies four main kinds of services, plus an addi-

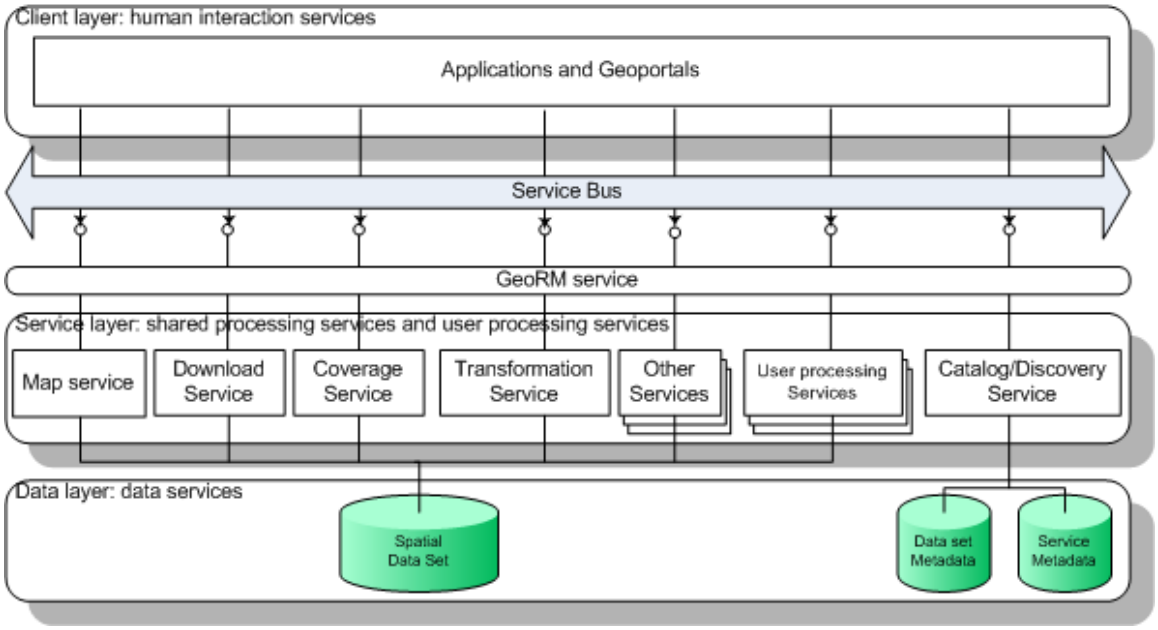


Figure 1.1: Spatial Data Infrastructure technological implementation.

tional Geographic Right Management (GeoRM) service:

- **Data layer** contains services that manage spatial data and metadata.
- **Service layer** contains shared processing services (map, download, coverage, transformation, and other services) and user processing services.
- **Client layer** contains human interaction services like desktop and web applications (e.g., GIS portals).

The overriding objective of SDIs is to facilitate access to geographic information assets that are held by a wide range of stakeholders in both the public and the private sectors in a nation or a region with a view to maximizing overall usage. This objective requires coordinated action by governments. SDIs must also be user driven, as their primary purpose is to

support decision making for many different purposes. SDI implementation involves a wide range of activities. These include not only technical matters such as data, technologies, standards, and delivery mechanisms but also institutional matters related to organizational responsibilities, overall national information policies, and availability of financial and human resources.

The work in [86] describes the process of SDI development and implementation as a set of four main components. As can be clearly seen in the Victorian Spatial Information Strategy (VSIS, Australia) from 2004 to 2007 (Department of Sustainability and Environment, Victoria, Australia, 2005), the four components are (see Figure 1.2): (*i*) institutional arrangements that are required for delivering geographic information, (*ii*) tasks related to the creation and maintenance of fundamental datasets, (*iii*) procedures for making geographic information accessible, and (*iv*) ways of facilitating the development of strategic technology and applications.

Based on current SDI initiatives as summarized above, many countries are developing SDIs at different levels ranging from local to state/provincial, national and regional levels. As a result of developing SDIs at different levels, a model of SDI hierarchy that includes SDIs developed at different political-administrative levels was developed and introduced [114]. Thus, the number of SDIs is rapidly growing and one of the main challenges is to achieve international cooperation and collaboration in order to support regional, national and international SDI developments. It should allow nations to better address social, economic, and environmental issues [85]. This is the main goal of the Global Spatial Data Infrastructure[21] (GSDI) association, an inclusive organization of organizations, agencies, firms, and individuals from around the world.

[21] http://www.gsdi.org/

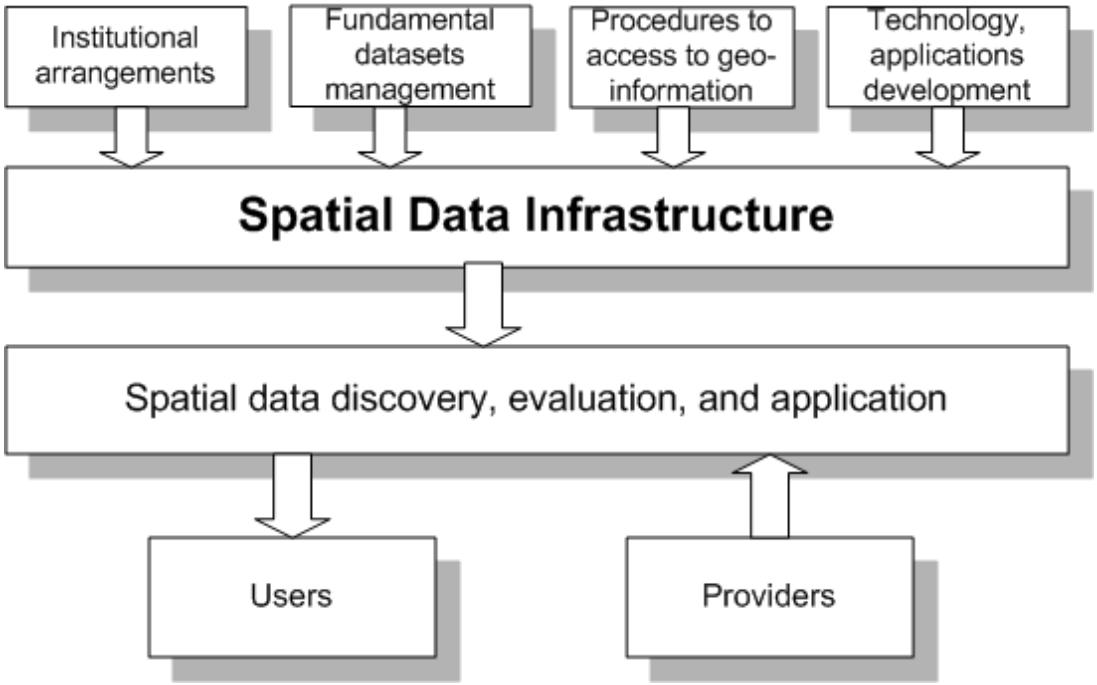


Figure 1.2: Spatial Data Infrastructure components.

1.3 Summary

In this chapter we presented a vision of the usefulness and the integration of geographic information that was first posed in 1998 by the U.S. vice president Al Gore. We then delineated the evolution of this vision toward the SDI model, a modern paradigm that implements distributed geographical information systems. Next, we summarized the main international initiatives that require international, national and regional SDI models for the collection, integration, and dissemination of geographical information, such as SEIS, INSPIRE, GEOSS, and GMES. Finally, we focused on the definition of SDI, we illustrated an example of its technological implementation, and we drew the main components of SDI development and implementation.

In the next chapter we will present some interoperability issues among distributed information systems. We will discuss the use of ontologies to

support semantic interoperability among heterogeneous systems, and we will focus on semantic heterogeneity in geographic information integration among distributed SDIs.

Chapter 2

Information systems' interoperability

A key issue in the development of SDIs is the advancement of *interoperability* that is one of the key conditions for information system integration[1]. IEEE [60] gives a very general definition of interoperability as *the ability of two or more systems or components to exchange information and to use the information that has been exchanged.* Interoperability among software systems is defined by ISO[2] as follows: *the capability to communicate, execute programs, or transfer data among various functional units in a manner that requires the user to have little or no knowledge of the unique characteristics of those units.* In what follows, we first give a description of the main dimensions of interoperability in the field of information systems (§2.1). Then, we give a brief explanation of the key role of ontologies in information integration techniques (§2.2). Finally, we focus on the semantic heterogeneity problem (§2.3) in the field of geographic information, both for geographic data and for geographic services.

[1]We consider SDI specialized information systems, dedicated to the management of the geographical information.

[2]ISO/IEC 2382-01, Information Technology Vocabulary, Fundamental Terms

2.1 Dimensions of interoperability

One of the enduring approaches to studying the key interoperability issues in distributed information systems has been to use the fundamental dimensions of *autonomy, distribution, dynamics, scalability, and heterogeneity* [122, 155].

2.1.1 Autonomy

A component participating in a distributed system may exhibit several types of autonomy, including: (*i*) *design*, i.e., the universe of discourse, the representation of the data, the context, the constraints, the functionality of the system, the association with other systems, and the implementation, (*ii*) *communication*, i.e., the ability of a component to decide whether to communicate with other components, (*iii*) *execution*, i.e., the ability of a component to execute local operations without interference from external entity and to decide the order in which to execute external operations, (*iv*) *association*, i.e., the ability to decide whether and how much to share functionalities and resources, and (*v*) *participation*, i.e., the ability to associate or disassociate itself from one or more distributed systems.

2.1.2 Distribution

With significant improvement of the communication technology, global information infrastructure, and distributed computing infrastructure, the dimension of *distribution* of data has achieved a very broad scope, from a single system to a global, interoperable and complex system. Distribution is defined along three dimensions: (*i*) *physical*, i.e., data can be stored in a central node or reside on different nodes which are geographically distributed and connected, (*ii*) *logical*, i.e., data can be either described by means of a single global schema or no single global schema is defined, and

(*iii*) *operational*, i.e., either there exists a global shared register or index of the resources managed by a central authority, or the register is stored and managed locally by each node.

2.1.3 Dynamics

If an information system possesses design, association, and participation autonomy, then it is a subject of dynamics and whenever a change is introduced, the system must take an action (either centrally or locally) in order to compensate for this change.

2.1.4 Scalability

Participation autonomy implies that an information system can include an arbitrarily large number of nodes. We say, that an information system is scalable if, with the growth of the number of nodes, it is capable of maintaining or improving the level of Quality of Service; and it is unscalable otherwise [155].

2.1.5 Heterogeneity

Interoperability problems emerge when there exists *heterogeneity* in information (represented by data sets or services). Heterogeneity can be characterized by the conflicts that occur when two resources (data sets and/or services) are combined. There have been many different classifications to types of heterogeneity [8, 14, 32, 65, 66, 122, 120, 123, 130, 148, 155]. We can say that heterogeneity pervades information systems, because of their design and implementation differences. Heterogeneity problems arise at *system* level and at *information* level that, in turn, can be classified into *syntax*, *structure* and *semantics* heterogeneity. Interoperability can be achieved at each of these levels when it resolves the heterogeneity issues

at that level. *System heterogeneity* is related to hardware, system software and communication protocols (e.g., DBMS heterogeneity, operating systems heterogeneity, transmission heterogeneity, etc). *Syntax heterogeneity* occurs when information is expressed into two different languages or formalism. *Structural* or *schematic heterogeneity* means that different information systems store their data in different structures. *Semantic heterogeneity* is the focus of this work and considers the contents of an information item and its intended meaning. As the actual and future information system increasingly addresses the information and knowledge level issues, it will increasingly require semantic interoperability.

The use of ontologies for the explication of implicit and hidden knowledge is a possible approach to overcome the problem of semantic heterogeneity [36]. The work in [123] mentions interoperability as a key application of ontologies, and many ontology-based approaches to information integration in order to achieve interoperability have been developed [148]. Thus, in what follows, and to support next sections, we will give an explanation of the key role of ontologies in information integration techniques.

2.2 Ontologies

Ontology is the science that seeks to study in a rational, neutral way all the various types of entities and to establish how they hang together to form a single whole (*reality*). Thus, this science studies relationships among objects, like the ones used by people when they classify objects, with different levels of focus and granularity.

The term *ontology* has another use, however, which arose in recent years within the domain of computer and information science from information systems, database specifications and the like. This definition is based on the notion of *conceptualization*, i.e., a system of concepts and categories

which divide up the corresponding universe of discourse into objects, processes and relations in different sorts of way. Often this conceptualization is not explicit, but tools can be developed to render it explicit.

The work in [55] defines an ontology as a *specification of a conceptualization*. To share information between separately engineered system components (such as databases, agents, peers, etc.) in a meaningful way requires that separate components commit, to some extent, to an agreed conceptualization of the application domain. Commonly, this specification of such a conceptualization includes a definition of the ontology's vocabulary. In information science sense, an ontology is *a neutral and computationally tractable description or a theory of a given domain which can be accepted and reused by all information gatherers in that domain* [126].

More formally, the work in [57]: *(i)* defines an ontology as a *logical theory accounting for the intended meaning of a formal vocabulary* (i.e., its ontological commitment to a particular concept of the world) and *(ii)* shows that the study of good conceptualizations, i.e., that are transparent to some corresponding independent domains of reality, can have advantages also in eliminating certain kinds of errors in data collection and representation.

A key role is played by machine ontologies, which are machine-accessible representations of knowledge that are used for inferring intra- and inter-resource relationships. Recent research efforts in the field of the Semantic Web [11] have contributed considerably to the deployment of ontology-based applications by providing a theoretical foundation (Description Logics [7]), ontology languages (e.g., the Web Ontology Language (OWL) [6]), and tools for ontology creation, access and reasoning with web-based (machine) ontologies. The power of web-based ontologies lies in their interoperable (XML based) representation, the use of unique namespaces and the fact that they allow for automated reasoning.

As ontologies are produced in larger numbers and exhibit greater com-

plexity and scale, research efforts are a new generation of complex systems, which can make available both large volumes of data and large reusable semantic resources. These systems will provide new functionalities in the emerging semantic web, in the automation of business to business relationships, and also in company intranets. As an example of these kinds of systems we mention the *Ne*tworked *On*tologies (NeOn)[3] project whose goal is to advance the state of the art in using ontologies for large-scale semantic applications in the distributed organizations. The NeOn project goal is to create the first ever service-oriented, open infrastructure, and associated methodology, to support the overall development life-cycle of a new generation of large scale, complex, semantic applications.

Depending on the precision of the its specification, the notion of ontology includes various data and conceptual models [34]. The term ontology is used in this thesis in a wide sense, and, hence, encompasses, e.g., sets of terms, classifications, database schemas, web service descriptions, and thesauri. Also in geographic field, different service providers can specify their background knowledge by using different application ontologies [57], so, heterogeneity problems arise when integrating the information from different application ontologies. Heterogeneity of GIS ontologies has been addressed in many works during the last decades, [98, 152]. Semantic heterogeneity of GIS can be undertaken by considering their ontological aspects.

2.3 Semantic heterogeneity of geo-information

Beyond the ability of two or more computer systems to exchange information, *semantic interoperability* is the ability to automatically interpret

[3] http://www.neon-project.org/

the information exchanged meaningfully and accurately in order to produce useful results as defined by the end users of both systems. To achieve semantic interoperability, both sides must defer to a common information exchange reference model. Semantic heterogeneity arises when the content of the information exchange requests are not clearly defined: what is sent is not the same as what is understood. So, in order to achieve semantic interoperability in a heterogeneous information system, the meaning of the information that is interchanged has to be understood across the systems. Semantics conflicts occur whenever two systems do not use the same interpretation of the information [148]. Semantic heterogeneity includes *terminological or naming conflicts*, *data type conflicts* and *conceptual conflicts* [123]. *Terminological* conflicts occur when names refers to the same entity in different information systems. This can be caused by the use of different natural languages, e.g., *River* vs *Río*, the use of synonyms, e.g., *Watercourse* vs *Stream*, etc. *Data type conflicts* occur when the same name in different information systems refers to the same concept, but is represented with different data types, e.g., *Address* can be either a complex type or a *String*. *Conceptual conflicts* occur when the same name in different information system is represented by the same data type, but refers to different domain concepts, e.g., *java* can be represented as a string but can refer to an island, a kind of coffee or a programming language.

SDI, like other information technologies, must be implemented in a manner that allows easy interoperability between heterogeneous organizations and systems. The common SDI framework is based on a generic software platform, which supports a variety of geographic dataset types as well as comprehensive tools for data management, editing, analysis, and visualization. Moreover, to share geo-datasets, a number of geographic services have to be provided by the system. Heterogeneity issues pervade both geographic data and geographic services, since they can connect heterogeneous

organizations and systems. In this section separate analysis' for geographic data and geographic services heterogeneity issues are presented.

2.3.1 Geo-data heterogeneity issues

One of the key services supplied by an SDI is the possibility to retrieve geographical datasets provided by heterogeneous resources. Due to the fact that the logical architecture of an SDI can be based on a set of different data resources, heterogeneous geographical information has to be integrated. Since each geo-data producer adopts internal rules in order to manage its geographical datasets, heterogeneity at the data level arises from a number of different reasons [13, 41, 53]:

- **Different syntax.** Geo-datasets are retrieved from different sources that can use different representation of geospatial objects and data formats (e.g., raster or vectors, coordinate systems, and different formats like ESRI shape files, Mapinfo Files, AutoCAD DWG files, and GRASS files).

- **Different structure.** Geographical features can be represented using different geometrical and data schemas. This refers to the differences in database models or schemas, e.g., a particular geographic feature may be classified using different classes in different databases or different geometric representations (for instance, roads can be represented using polygons or lines) or may be represented using multi-temporal techniques [16, 111].

- **Different semantics.** Interoperability problems due to different semantics are caused by different reasons. Naming conflicts occur when classes or attribute types with different semantics are given the same

names (homonyms) or when classes or attribute types that are semantically the same are named differently (synonyms). This will also influence the geometrical representation of objects, because abstraction of the world is based on the semantics of each discipline. It is intimately tied to the application context or discipline for which the data is collected and used.

Specifically, for distributed spatial systems, heterogeneity is accentuated for geo-data that have specific properties, different from other types of data, including [13, 71, 74, 75, 128, 152]:

- **Multiple versions**: geo-data can be represented at different representation (e.g., raster or vector modes), granularities or levels of detail for the spatial features. Multiple versions of the same entities on the Earth's surface can differ radically in terms of data model, scale, data generalization, conceptual model, and semantics the data collectors use. The main reason is that data collectors are often represented by different government agencies at different levels (e.g., regional, national and international) in different countries. Specifically, for the case of geo-data integration, we have also scale conflicts and different precisions/resolutions issues. For example, NASA lost a $125 million Mars orbiter because a Lockheed Martin engineering team used English units of measurement while the agency's team used the more conventional metric system for a key spacecraft operation[4]. Additional factors have to be also considered like *integration alignment* problem (e.g., data collected at different scales, data corrected using different elevation models, and data produced using different topographic sources).

- **Implicit linking**: we have to consider topological relationships be-

[4] http://www.cnn.com/TECH/space/9909/30/mars.metric.02/

tween objects, thus, geographic information enables linking without explicit references, for instance, via coordinate reference systems. For instance, a bridge can be implicitly linked to a river or to a road it crosses. Thus, some GIS systems can provide different services to check and compute implicit links.

2.3.2 Geo-service heterogeneity issues

Distributed service discovery, composition and coordination are the main research topics in the field of web services. GIS desktop applications provide to the user a lot of complex functions in order to perform GIS data acquisition, creation, analysis, visualization and mapping. For years these functions were accessible only through the GIS desktop applications, but recently, GIS services have became published and available on the web. Distributed Service Oriented Architecture (SOA) is a common framework for modern distributed information systems [31, 110]. SOA and Open Geospatial Consortium (OGC)[5] specifications are the base technology used by an SDI in order to provide catalog services for discovering appropriate data and services for a specific task [79, 138]. Figure 2.1 shows the three main building blocks in GIS SOA: (*i*) a GIS user community (potential users of GIS services), (*ii*) GIS web services (published by some GIS service providers), and (*iii*) a GIS catalog service (where available data and services are published by providers and discovered by users).

After discovering, services can be composed or coordinated to provide complex functionalities. Although at present, the main available web service in GIS is the map request service, the trend is to supply a technological environment that provides a number of stand-alone GIS services. At the moment, the majority of these geo-services exist as single services. In the

[5] `http://www.opengeospatial.org`

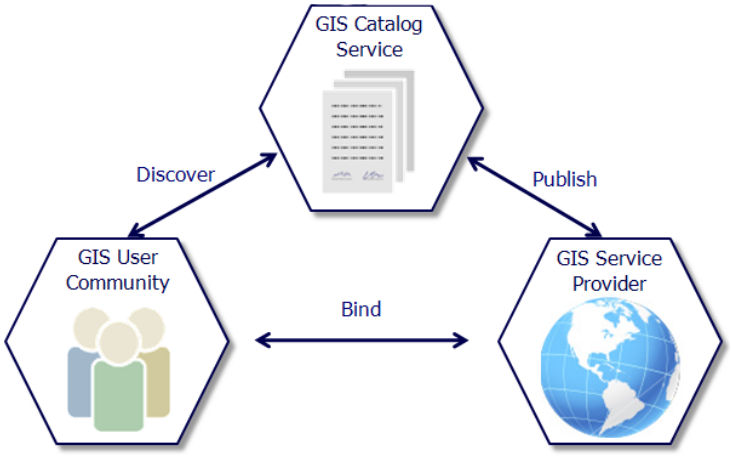


Figure 2.1: Service Oriented Architecture.

case of a request for a complex service a manual and static composition of a number of predefined geo-services has to be performed. Composition and coordination of services is very important, because of the reusability and modularity of services and hence permits the reduction of the costs in the development of an information system. However, discovery of services using the process specification or the syntactic description of the tasks performed by a service has to be performed manually by an expert employer. Modern software systems may provide a huge amount of services and often, discovering of proper services by browsing service catalogs based on a simple classification criteria (e.g., the Universal Description, Discovery, and Integration (UDDI)[6]), is a difficult task. Manual efforts when chaining services depends from the syntactic nature of the specification of a services. Usually, the technical invocation of services is described in terms of its structure and data schema specifications. A formal description of its functionality and the meaning of data are often missing. Thus in these cases, when using automatic composition, only the syntactical structure of the service can be verified [72]. Moreover, when integrating geo-services from

[6] http://www.uddi.org/pubs/Iru_UDDI_Technical_White_Paper.pdf

heterogeneous sources, some specific issues have to be taken into account [74]:

- **Maps as implicit interfaces**: everyone is familiar with reading maps, so they are a natural human-machine interface for the services interacting with the user and presenting (intermediate) results of geo-information. Thus, the map service is the typical and most used service in the GIS field.

- **Geometry based information**: since geo-information is geometry based, geo-service interface has always to take into account the geometric component of the data they provide and process. So GIS services input and output parameters often include, e.g., the bounding box of a map and the coordinate reference system of geographical layers.

- **Specific topological operations**: it is also possible to apply a whole set of common mathematical tools in geo-services to compute their topological relationships, e.g., to compute the distance between two objects, the buffer around an object, the intersection between different features, and the neighbors of a polygon.

The challenge is the (semi)-automatic composition of arbitrary services in order to obtain flexible complex services based on the available web services. In practice, however, chaining geographic services is a nontrivial task, mostly because individual web service providers use different syntactic structures and different vocabularies to define web service signatures and descriptions. At present, even if there exist some languages to formally specify the semantics of a web service (e.g.,WSDLS [2], SWSF [9], WSMO/WSML [15, 117], SA-WSDL [35] and OWL-S [84]), no standard notions are used for defining the semantics of a geographic web service.

In most of the cases the unique source that describes a web services is its WSDL file. The WSDL file contains the syntactical description of the signature of each service operation. But, as described in [71] *in today's GIS service architectures, the interfaces between agents, computational and human, are those of web services...*and*...the interface of a service is formally captured by its signature.* We can consider signatures (name, inputs and outputs) of web services as tree-like structures or simple ontologies. The terms of these tree-like structures implicity contain a classification of the background knowledge of the provider.

2.4 Summary

In this chapter we investigated specific dimensions of interoperability in the field of information systems, such as autonomy, distribution, dynamics, scalability, and heterogeneity. Specifically, we classified different types of heterogeneity, i.e., system, syntax, structural, and semantic heterogeneity. Then, we discussed about ontologies and their usefulness to achieve interoperability among heterogeneous systems. Finally, we focused on specific characteristics of semantic heterogeneity in the field of geographic information integration. In particular, we presented semantic heterogeneity issues when either geographic data or geographic services have to be integrated.

As we saw in the previous part, diverse research domains are interested when integrating information systems, and in particular when integrating geographic information. Thus, in the following part we will illustrate some state-of-the-art related to semantic heterogeneity in GIS data and services integration, ontology matching techniques and evaluation (being a solution to the semantic heterogeneity problem), and application of P2P systems to distributed GISs.

Part II

State of the art

Chapter 3

Semantic heterogeneity in geo-information

The research area of semantic integration of geographic information is relatively young. In fact, even if the concept of geo-service publication is not new, OGC specifications and ISO standards have become stable only during the last years. Thereafter, different geo-information providers started to publish their geo-data and services on the web in a standardized manner. Only recently, the integration of geographic information became relevant and feasible because of the availability of GIS web services.

The main technological infrastructure to support web service publication, discovery, selection and composition is based on SOA. This architecture is rapidly becoming the standard in the domain of distributed systems. In the case of geographic information, a SOA framework has been developed by OGC [102]. Technical interoperability among geo services is mainly approached by using the OGC interoperability specifications. The most frequently used are the Web Map Service (WMS)[1] [101] and the Web Feature Service (WFS)[2] [103] specifications.

OGC specifications and SOA technological solutions provide syntactic

[1] http://www.opengeospatial.org/standards/wms

[2] http://www.opengeospatial.org/standards/wfs

interoperability and cataloguing of geographic information. Specifically, OGC has been published the OGC Reference Model (ORM)[3] set of specifications [102], the OpenGIS Web Services Common (WS-Common) [4], the OpenGIS Web Processing Service (WPS)[5] and the Catalog Service (CAT)[6] specifications:

- **WS-Common** specifies parameters and data structures that are common to all OGC Web Service (OWS) standards. The standard normalizes the ways in which operation requests and responses handle such elements as bounding boxes, exception processing, URL requests, URN expressions, and key value encoding.

- **WPS** provides rules for standardizing inputs and outputs (requests and responses) for geospatial processing services, such as polygon overlay. The standard also defines how a client can request the execution of a process, and how the output from the process is handled. It defines an interface that facilitates publishing of geospatial processes and clients' discovery and of binding to those processes.

- **Catalog Service** specification supports the ability to publish and search collections of descriptive information (metadata) for data, services, and related information objects. However catalog services, still do not define any method for overcoming the semantic heterogeneity problem, described in the previous section.

Some works have already been performed in automatically and syntactically locating distributed SDI resources: Skylab Mobilesystems Ltd.[7] uses a form of web crawling to locate WMS servers. Mapdex[8] has a similar so-

[3] http://www.opengeospatial.org/standards/orm
[4] http://www.opengeospatial.org/standards/common
[5] http://www.opengeospatial.org/standards/wps
[6] http://www.opengeospatial.org/standards/cat
[7] http://www.ogc-services.net
[8] http://www.mapdex.org

lution which is oriented toward ESRI ArcIMS servers in addition to WMS. Mapdex uses Google search API to find possible WMS sites by searching for WMS-specific query strings appended to URLs.

Beside these results which adopt syntactic and structural specifications and solutions, semantic heterogeneity is the actual research issue on geo-information integration. Thus, in what follows, we first focus on related work in this research field by analyzing both geo-data semantic integration (§3.1) and geo-service semantic integration (§3.2). Then, we present recent advances in the field of ontology matching and ontology matching evaluation (§3.3)

Part of the material presented in this chapter has been published in [143, 144].

3.1 Geo-data semantic integration

Integrating data from heterogeneous sources is the fundamental task in order to enable value added services. Such a task is complex, especially if the goal is the integration of different geographic datasets managed by different providers. In the overall task, when integrating geo-data from different sources it is possible to identify two main issues: (i) *geo-data integration alignment* and (ii) *geo-data heterogeneity general issues.*

The first problem (*geo-data integration alignment*) depends on a number of factors including: different geographic projections, data collected at different scales, corrected using different elevation models, and data production using different topographic sources. Such problems have been identified and addressed by current research. For example, the work in [21] and follow-up studies in [20, 89] propose a general-purpose geospatial data integration framework to access and retrieve geospatial sources, to

accurately and efficiently integrate these sources using dynamically conflation operations in the integration plans, and quickly incorporate new sources that support geo-data standards. Based on these ideas the Terraworld[9] system, which integrates various geospatial data types, has been developed.

The second problem (*geo-data heterogeneity general issues*) depends on many aspects: syntax aspects (e.g., different data encoding), structural aspects (e.g., different schemas) and semantic aspects (that refers to the differences in interpretation of real-world phenomena). Syntactic and structural aspects are covered by standards developed by OGC and ISO/TC211 and involve specifications on feature data, metadata, services, etc. These standards (e.g., the Open Geospatial Consortium Geographic Markup Language (GML) specification [107]) may specify contracts of different levels of abstraction, representation, and detail. Therefore, semantic aspects are now the actual challenge in the field of geo-information integration. In what follows, a number of initiatives which aim to face semantic interoperability problems are described.

The work in [38] uses ontologies to reduce geographic information heterogeneity. This work proposed a detailed description of the role of ontologies in geographic data modeling and a solution called ontology-driven geographic information system (ODGIS) that acts as a system integrator. In ODGIS, an ontology is a component, such as a database, cooperating to fulfill the system's objectives. The work suggests an architecture for ODGIS which includes an ontology editor and its embedded translator plus a user interface to browse ontologies.

In the GEOscience Network (GEON)[10] project an interoperability framework has been developed to allow a data provider to register a geographic

[9] http://www.isi.edu/integration/TerraWorld/

[10] http://www.geongrid.org

dataset with one or more mediation ontologies (e.g., standards for data structure and content) and subsequently query the different datasets in a uniform fashion [94]. The system comprises an ontology repository, a dataset registration procedure, and a query rewriting system. Structural and semantic heterogeneities of data sources are resolved using information from the dataset registration procedure and ontologies. Multiple ontologies are supported in the system by allowing users to manually define an articulation between two ontologies which equates some concepts in the source ontology to some concepts in the target ontology. Users are able to switch between ontologies for which an articulation exists. Nevertheless, this system can be adopted only in the case when the user adheres to the community (using the GEON registration procedure).

A specific methodology for geo-ontologies integration was proposed in [59], where G-Match, an algorithm and an implementation of a geographic ontology matcher, was presented. The goal is to give a similarity measure between two different geographic ontologies when integrating them. In order to do that, the algorithm considers the features of a concept separately and then gives some weights for each geographical feature (name, attributes, taxonomy, conventional, and topological relationships) to compute the overall similarity between two concepts. As the information may be defined in different levels of detail, there is no perfect combination of the weight factors assigned to each concept features. So, some sort of self-adaptation of the weight, depending on the input ontology, has to be performed.

The main focus in [112] was to integrate diverse spatial repositories for geographic applications using a SOA for the discovery and retrieval of geospatial information. The architecture uses a central ontology as metadata information, which acts as service broker. Also here, the system is composed of a domain ontology (a global shared vocabulary) and of the

service providers application ontologies that need to adopt the central ontology.

In the Semantic Web-Service Interoperability for Geospatial Decision Making (SWING)[11] project, the issue of geographic information semantic integration has also been tackled. The main aim of the project is to *deploying Semantic Web Service (SWS) technology in the geospatial domain. In particular, SWING project addresses two major obstacles that must be overcome for SWS technology to be generally adopted, i.e., to reduce the complexity of creating semantic descriptions and to increase the number of semantically described services. The objective of SWING is to provide an open, easy-to-use SWS framework of suitable ontologies and inference tools for annotation, discovery, composition, and invocation of geospatial web services.* Below, we mention the most related (to this thesis) works from SWING on geo-data integration:

- The work in [77] presented an ontology based approach to geographic information retrieval that contributes to the solution of existing problems of semantic heterogeneity and hides most of the complexity of the required procedure from the requestor. Nevertheless, in the proposed approach, it is assumed that a requestor searches for only one source that provides all the required information. Moreover, the data provider has to create and register an application ontology that represents one of the bottlenecks for scalability.

- The problem of generating semantic annotation of geo-data was tackled in [67]. In this work, semantic annotation is understood as making explicit the relationship between a data schema and a domain ontology by defining mappings from elements of the schema to elements in the ontology. Specifically, a strategy for partially automating this

[11] http://www.swing-project.org/

process is introduced. It transforms a data schema into an ontology and applies spatial analysis methods during the matching process for exploiting extensional knowledge.

- A similarity-based information retrieval system has recently been introduced by [64]. This work proposes an architecture, based on the SIM-DL similarity theory [63], to support users and systems during information retrieval operations. Use cases for a human web interface, as well as for an SDI system integration workflow and analysis are provided. The proposed architecture includes standard services, such as WMS and WFS, as well as a catalog service including a feature type catalog (CS-W and FTC) and a Web Similarity Service (WSS) based on SIM-DL. Both services and the client are assumed to use the same ontology and CS-W needs to store metadata about three types of resources: (*i*) services, (*ii*) data, and (*iii*) feature types.

The work in [119] has proposed and evaluated a semantic similarity method useful in retrieval of geo-information. This work presents a computational model for semantic similarity measurement. The similarity measure retrieves relevant information by measuring the semantic similarity of geo-information to a given query. The model is hybrid in the sense that it enables the necessary expressiveness to capture semantics underlying geospatial data by combining two existing similarity measures: the geometric for representing concept properties and the network model for representing (spatial) relations between geographic features.

Recently, a structured-based ontology alignment method has been proposed by [26]. This work presents two fully automatic alignment methods that use the graph structures of a pair of ontologies to establish their alignment, that is, the semantic correspondences between their concepts. Specifically, the Descendants Similarity Inheritance (DSI) method, which

uses the relationships between ancestor concepts, and the Siblings Similarity Contribution (SSC) method, which uses the relationships between sibling concepts, are presented. Both methods are used in conjunction with a concept based method, i.e., the base similarity method (that determines the similarity between a target ontology concept and a destination ontology concept with the help of a dictionary). These methods were implemented and evaluated on the alignment of two wetland ontologies: (*i*) the United States Cowardin classification system [24] and (ii) the South Africa National Wetland Classification Inventory [30]. Moreover, in this work, a comparison with the Similarity Flooding algorithm [88] was established and results of the participation at the OAEI [131] competition were discussed.

3.2 Geo-service semantic integration

The activity of integrating services is commonly referred as different terminology such as, for example, service composition, coordination [132] and chaining (dynamic composition) [150]. This requires that services can be easily found, and that they are executable and interoperable. As we saw in the previous chapter, a major problem to integrate services is the semantic heterogeneity between different service providers. In what follows, we list some recent research initiatives which aim is to tackle with this issue in geo-information domain.

The work in [37] presented an approach to systematic composition of web services. Among the main contributions, there was a support to web service composition using domain ontologies with multiple dimensions (e.g., space, time, and object description). Specifically, web services were composed under *utilization scopes*, i.e., specific context in which different data

sets and distinct versions of a repertoire of services can be used. The second main contribution of this work consists in showing how it resolves some open issues in web service composition. This is done by modeling a concrete application scenario of agro-environmental planning.

The work in [29] presents a study on automatic creation end execution of geospatial processing models based on users' product specification in GeoBrain [28], a web-service based geospatial knowledge system, to produce the user-specific result. The whole process (design, information fusion and data generation) is implemented on semantic and syntactic interoperability between data and processes. Specifically, this process is driven by the knowledge represented in geospatial and application-specific ontologies. Here, Ontology Web Language (OWL) [6], OWL-S[12] and Business Process Execution Language (BPEL) [4] were used to give meaning to diverse data sources and geo-processing services. In order to chain and discover geo-web services an OWL reasoner was applied as inference engine for the knowledge-base in use.

For the geo-service chaining specific case, a syntactic and semantic analysis was made in [74]. This work develops a methodology that combines service discovery, abstract composition (identifying service chain functionality with the help of conceptual parameters), concrete composition (managing control and data flow among specific services), and execution. The specific application scenario is represented by a Risk Map service chain. The presented approach uses domain ontologies for the different steps in geographic service chaining.

The work in [156] proposed a tool-set to compose geo-web services using BPEL. In turn, the work in [133] combined WSMO [117] and IRS-III[13] for semantically composing geo-spatial web services.

[12] http://www.w3.org/Submission/2004/SUBM-OWL-S-20041122/

[13] http://technologies.kmi.open.ac.uk/irs/

Geo-service integration has been also investigated in the SWING project (§3.1), in particular:

- The work in [76] proposed a methodology for service discovery. This approach uses ontologies describing geospatial operations to create descriptions of requirements and service capabilities. This work investigates how the methodology can be integrated into existing architectures for SDIs, and presents a prototypical implementation. This approach currently considers only plug-in or exact matches between signatures in order to limit the number of found services.

- A comparison between BPEL and WSMO approaches has been made in [49]. This work proposed a semantic web service composition using WSMO as an improvement of BPEL limitations. Moreover, a use case application (namely ProCon) was developed and implemented in BPEL and in Web Service Execution Engine (WSMX) [58].

- The work in [87] presented an extensible architecture for a web service catalog which supports multiple service description standards (schema-based, like WSDL [22], as well as ontology-based, like WSMO/ WSML [15]) and discovery tools. The discussion and implementation of the catalog focuses on geospatial web services. In particular, the implementation of the proposed architecture makes the import and discovery of web services described either with WSDL or the OGC *getCapabilities* operation result. Moreover, WSMO has been used to describe service ontologies.

Recent advances in the field of geo-service integration has been made also in the ORCHESTRA project[14]. ORCHESTRA project *designs and implements the specifications for a service oriented spatial data infrastructure for improved interoperability among risk management authorities in*

[14] http://www.eu-orchestra.org/overview.shtml

Europe, which will enable the handling of more effective disaster risk reduction strategies and emergency management operations. ORCHESTRA main result is the development of an open architecture based on standards. Its specifications are contained in a document called the Reference Model ORCHESTRA Architecture (RM-OA). ORCHESTRA documentation provides a set of specifications about various RM-OA aspects, including: its reference model, services abstract specifications and services implementation specifications. Within this project the work in [78] presented a rule-based description framework (a simple top-level ontology as well as a domain ontology) and an associated discovery and composition method that helps service developers to create such service chains from existing services.

3.3 Ontology matching

Ontology matching [34, 123] is a plausible solution to the semantic heterogeneity problem faced by information management systems. Ontology matching takes two graph-like structures such as, for instance, lightweight ontologies [42] and produces an alignment (set of correspondences) between the nodes of those graphs that correspond semantically to one another. In what follows, we first illustrate a survey on the ontology matching techniques. Then, we present some approaches to ontology matching evaluation methods.

3.3.1 Ontology matching techniques

A substantial amount of work that tackles the problem of semantic heterogeneity has been done in the semantic web, artificial intelligence and database domains, where ontology matching is viewed as a plausible solution, see, e.g., [34, 97, 124] for recent surveys, while examples of individual

approaches addressing the matching problem can be found on the *ontology matching web site*[15]. These solutions take advantage of the various properties of ontologies (e.g., labels, structures) and use techniques from different fields (e.g., statistics and data analysis, machine learning, linguistics). These solutions share some techniques and attack similar problems, but differ in the way they combine and exploit their results. A detailed analysis of the different techniques in ontology matching has been given in [34].

The most similar to the solution that we used in our scenario are the approaches taken in [1, 108, 129], where the services are assumed to be annotated with the concepts taken from various ontologies. The matching algorithms of those works combine the results of atomic matchers that roughly correspond to the element level matchers[16] exploited as part of the work in [48] and [43] which we applied in our scenario.

3.3.2 Ontology matching evaluation

The ontology matching evaluation theme has been given a chapter account in [34]. There are several individual approaches to the evaluation of matching approaches in general, see, e.g., [46], as well as with web services in particular, see, e.g., [90, 108, 129]. Beside, there are annual Ontology Alignment Evaluation Initiative (OAEI)[17] campaigns [17, 33]. OAEI is a coordinated international initiative that organizes the evaluation of the increasing number of ontology matching systems. The main goal of OAEI is to support the comparison of the systems and algorithms on the same basis and to allow anyone to draw conclusions about the best matching strate-

[15] `http://www.OntologyMatching.org`

[16] Element level matching techniques compute correspondences by analyzing concepts in isolation, ignoring their relationships with other concepts. In turn, structure level matching techniques compute correspondences by exploiting the results of element level matchers and by analyzing relationships between concepts.

[17] `http://oaei.ontologymatching.org/2006`

gies. Unfortunately, at present, matching of web services has not been addressed by OAEI. In turn, there is the Semantic Web Service (SWS) challenge initiative which aims at evaluation of various web service mediation approaches [113]. However, as also noticed in [118], the key problem with the current evaluations of web service matching approaches is the lack of real world web service data sets as well as their size, for example as from [113], the participants of SWS were operating with 20 web services[18].

3.4 Summary

In this chapter we presented some recent advances in the field of semantic heterogeneity solutions among distributed systems. Specifically, we first defined the role of ontologies when integrating heterogeneous distributed systems. Then, we discussed geographic semantic integration both for geo-data and geo-services.

It is worth noting that most of the illustrated solutions employ a single (top-level) ontology. This allows for the reduction of semantic heterogeneity problem to the problem of reasoning within the shared ontology. However, the adoption of a common ontology for the geographic information communities is not practical, because the development of a common ontology has proven to be difficult and expensive [127]. In contrast we will see that, in our approach, we assume that geo-data and geo-services are described using terms from different ontologies. Therefore, the problem is shifted to the matching of different domain ontologies. Thus, in this chapter we also presented recent solutions for ontology matching techniques and their evaluation.

In the next chapter we will discuss the application of P2P systems in

[18] http://sws-challenge.org/wiki/index.php/Workshop_Innsbruck

the GIS field. We will specifically define the model underlining P2P architecture by describing a set of issues which have a major impact on the effectiveness and illustrate recent developments in the research field of P2P systems applied to GIS.

Chapter 4

P2P architectures in GIS applications

In our approach we use P2P technology to exchange information in an e-Response scenario and in particular to coordinate SDI services. Thus, in this chapter, we will present the P2P model, that is a paradigm of a distributed computing and refers to the computing systems and applications that connect distributed resources and perform functions in a decentralized way. Moreover, P2P technology supports the dimensions of interoperability we identified in the previous section [155]. In what follows, we describe main characteristics of P2P information systems (§4.1). Then, we argue for the application of P2P model to the GIS field (§4.2).

4.1 P2P model

P2P computing is considered to be the next evolutionary step in computing. This new direction in distributed computing focuses on networking and resource sharing with better reliability and scalability. There have been many attempts to define P2P systems mainly distinguished by the *broadness* they attach to the term. Generally speaking, the term *P2P* refers to *a class of systems and applications that employ distributed re-*

sources to perform a function in a decentralized manner [70, 93]. A more precise definition is given by [5]: *P2P systems are distributed systems consisting of interconnected nodes able to self-organize into network topologies with the purpose of sharing resources such as content, CPU cycles, storage and bandwidth, capable of adapting to failures and accommodating transient populations of nodes while maintaining acceptable connectivity and performance, without requiring the intermediation or support of a global centralized server or authority.*

As presented in [155] this definition captures some of the essential characteristics of P2P, even if it does not fully discriminate P2P from the conventional distributed data management systems and *locality* and *transitivity* property are missing. Nevertheless, this definition includes a set of issues which have a major impact on the effectiveness and deployment of P2P systems and applications [5, 93]:

Decentralization. *P2P systems are distributed systems... without requiring the intermediation or support of a global centralized server or authority.* Depending on the level of decentralization, P2P networks are classified as *pure* or *hybrid* centralized systems. In a *pure* P2P network every peer is an equal participant, and there is no centralization. This makes the implementation of the P2P models difficult in practice because there is no centralized server with a global view of all the peers in the network of files they provide. This is the reason why many P2P file systems are built as *hybrid* approaches, where some resources or services are centralized, e.g., there is a centralized directory of the files but the nodes download files directly from their peers.

Scalability. *...while maintaining acceptable connectivity...* An immediate benefit of decentralization is improved scalability. P2P networks scale well with increase in number of resources, while maintaining their

autonomy. Scalability is limited by factors such as the amount of centralized operations that needs to be performed and the ratio of communication to computation between the nodes.

Self-organizing. *...interconnected nodes able to self-organize into network topologies...* In P2P systems, self-organization is needed because of scalability, fault resilience, intermittent connection of resources, and the cost of ownership.

Cost of ownership. *...with the purpose of sharing resources such as content, CPU cycles, storage and bandwidth...* Shared ownership reduces the cost of owning the system and the content, and the cost of maintaining them.

Ad-hoc connectivity. *...accommodating transient populations of nodes...* In content sharing P2P systems and applications, users expect to be able to access content intermittently and autonomously, subject to the connectivity of the content providers.

Performance. *... while maintaining acceptable... performance...* P2P systems aim is to improve performance by aggregating distributed storage capacity and computing cycles of devices spread across a network. Performance is influenced by three factors: processing, storage, and networking.

4.2 GIS P2P applications

SDIs are pervaded by interoperability issues also because services offered by SDI are data-oriented services, which include a variety of complex data models and metadata. Discovering the appropriate services with related

geospatial datasets among a large number of available ones is a key task in the geospatial web service domain. Moreover, geo-information plays a fundamental role in spatial decision making activities.

For example, in emergency situation, activities are developed and implemented through the essential analysis of information. Fundamental activities such as *assessment, prevention, preparation, response*, and *recovery management* require the appropriate data to be gathered, organized, and displayed logically to determine the size and the scope of emergency management programs [137]. Usually such information is accessible through distributed data sources, the majority of which is spatial and can be mapped. Acquisition, use, and integration of geo-information with a wide range of seemingly unrelated information are crucial in emergency management situation. Specifically:

- *Assessment* and *prevention* activities require GIS data to evaluate the consequences of potential emergencies or disasters, in preparation activities.
- *Preparation* activities require GIS either to display real-time monitoring for emergency early warning or to develop of the preparation plans and of the personnel training with real data.
- In *Response* activities GIS can provide one of the primary components for computer-aided dispatch systems (e.g., selection and routing of the emergency units, identification of the current hazard areas, identification of the closest response unit, etc.)
- *Recovery management* activities require GIS to manage short-term recovery efforts and to evaluate damage assessment (e.g., locate damaged facilities, identify the type and the amount of damage, and begin to establish priority for action). Also long-term plans and progresses can be displayed and tracked utilizing a GIS.

Current geospatial technologies rely on SOA paradigm and are not designed to support collaboration in a group such as, for example, in the case of e-Response situation, where a lot of heterogeneous services have to be integrated and coordinated. A higher level of collaboration, beyond simply sharing geographic data, is required to support these kinds of efforts [70]. The research about integration of P2P system and distributed geo-information systems like SDIs is very novel and focuses on the ways in which P2P paradigm can be used to support distribution and sharing of spatial information.

In what follows we present developments in the recent research field of such P2P systems applied to GIS. These methodologies focus on how networks following the P2P paradigm can be used to support distribution and sharing of spatial data. They concentrate on exchange and transfer of geo-information and its data sources, such as, for example, maps or satellite images by the exchange between different and distributed systems. Of course, recent developments on GIS, such as the adoption of SDIs and the development of OGC specifications and international ISO standards, helped to simplify P2P interoperability between heterogeneous sources of geo-information.

The works in [69, 70, 18] refer to P2P as relatively new architectures to geo information research communities. These works introduce WORKPAD, a system that provides a 2-layered workspace suitable for emergency scenarios. The WORKPAD approach is based on the interplay between emergency networks and collaborative nomadic teams on one side and geo/work-data and content integration on the other side. The most innovative aspects of the WORKPAD system are: P2P data and content integration, adaptive work-flow management services and geo-collaboration. Moreover, these works also present a recent list of recent P2P architectures applied to GIS. Specifically:

- The work in [147] outlines an integration of geo-metadata and P2P architectures. The presented idea combines the P2P paradigm with a service-oriented SDI.

- The work in [56] presents techniques for enabling GIS services in a P2P environment to overcome the limitations of centralized GISs. In particular, the presented system facilitates GIS service discovery, composition and deploying by using information retrieval techniques to cluster similar service provider peers.

- In [154] a new framework of dynamic geo web services based on an ontology is proposed. Within a simple prototype of dynamic GIS services, an integration of agent based technologies and web GIS services is presented.

- The work in [19] proposes a GIS web service composition toolkit and introduces a P2P architecture for dynamically executed service composition. The system provides components to (*i*) manually retrieve GIS web services, (*ii*) store and catalog web service descriptions, (*iii*) parse a service composition specification, and (*iv*) execute the web service composition by using engines in P2P environments that collaborate with each other, aiming at invoking atomic services dynamically.

- In [153] a mechanism to automatically retrieve similar web services, when a web service composition fails because of the unavailability of a web service, is presented. The mechanism was implemented by using distributed service engines in P2P environment.

4.3 Summary

In this chapter, we presented P2P paradigm as the mean to exchange information in distributed environments. First, we defined the P2P model by describing a set of issues including: decentralization, scalability, self-organization, cost of the ownership, ad-hoc connectivity, and performance. Then, we illustrated the applicability of the P2P model to distributed geospatial web service domain, such as, for example, in an e-Response situation. Finally, we presented some relevant developments in the research fields of P2P systems applied to GIS.

In the next part we will present the motivating scenario and the P2P semantic matching framework we use to support this scenario. The novelty of the framework we present in this thesis is the application of a set of innovative technologies and research efforts to the field of distributed SDIs. Basically, we adopt and evaluate a P2P system plus a semantic matching approach to integrate and coordinate distributed GIS web services.

Part III

A P2P semantic matching framework

Chapter 5

Motivating scenario

Disaster scenarios are not, at heart, predominantly P2P. Large amounts of pre-organization will inevitable be done between disaster teams in the area, and standard methods of interacting during the emergency will be developed. Additionally, emergency teams would not normally be autonomous but would report to and be given directions from a control center whose task would be to manage the emergency; thus the structure would be more hierarchical than P2P.

However, disasters, by their very nature, are chaotic and unpredictable and the preparation for the disaster will not necessarily be sufficient. Not only will plans and strategies have to be amenable to change, but also the way in which units and individuals interact and the people that they interact with may also need to be flexible. Lines of communication may be down unexpectedly and communication bottle-necks, such as through command centers, may mean that units are left without instructions at vital moments and may need to turn to others in the vicinity who are not, in the original scheme, supposed to be supplying them with information or assistance.

Also, individuals or units who would not be expected to take part in the disaster effort may be called in unexpectedly: fire units from other areas in

case of the fire being too much for the local units, doctors who happened to be near the emergency area, etc. In these situations, the ability of units or individuals to switch to working in a P2P manner, with on-the-fly information sharing and interaction, at moments when the expected hierarchy breaks down, could be crucial. Moreover, as we have seen in the previous section, the GIS data is often available but in the majority of the situations needs to be located, integrated and fused together from very different sources [137].

The goal of this chapter is to describe specific requirements that our P2P system has to support, namely the e-Response overall scenario which we implemented as a testbed of the OpenKnowledge system [137, 139, 83, 135]. Specifically, we will first present a possible flooding event in the area of Trento (§5.1). We built the emergency scenario by collecting related documents from local emergency plans of the Autonomous Province of Trento and by interviewing the involved institutions personnel. Within this emergency response scenario we will also describe the main physical actors and information sources that support the emergency plan. Next, we will illustrate the main activities related to people evacuation when an emergency flooding alarm occurs (§5.2). Finally, we will describe specific SDI services we want the system to coordinate: the gazetteer, map and download services (§5.3).

5.1 e-Response scenario

In this section the goal is to describe, as an example of a natural disaster scenario, a possible flooding event in the area of Trento. In the first part of this section we will briefly describe the characteristics of the flooding that interests the area of our analysis. In the second part of this section, we will focus on a realistic scenario provided by the past experiences and local

plans of the Autonomous Province of Trento (PAT) as collected from interviews of the involved institutions personnel and from related documents. PAT has developed guidelines for the activities of planning, prevention, preparation, response and recovery for the whole province and for the every municipality of Trento [99] in the case of a civilian emergency. Thus, the description of the event, is based on [100] and [99].

The main civil protection terms of the thesis are taken from [61]. In particular, we have based the description of the non functional requirements on [100]. The individuation of the flooding area is based on the analysis of the past flooding events (1882 and 1966 flooding events) and over various morphologic and geologic observations. The work in [100] reports the identification of three main classes of flooding. From 0 to 1 meter (class 1), from 1 to 3 meters (class 2), and higher than 3 meters (class 3) of water level in the case of flooding (see Figure 5.1). In this section, for the identification of reasonable non-functional requirements (for instance, the number of persons involved in the emergency situation) we are going to consider the most probable situation: a flooding lower than 3 meters, but higher than 1 meter.

5.1.1 Scenario description

At 23:00 on November 4th, 1966, the river Adige, the main river of the Trento region, Italy, broke its banks at different sites and flooded the majority of the territory of Trento town. The main reason was a particularly intense period of rainstorm. Moreover, a considerable amount of oil, from housing heating systems and fuel depositories and petrol stations, mixed with the mud waters of the river invaded flooded areas nearby the river. The majority of the Trento population as well as surrounding areas had been affected.

Today, the flooding of the Adige river is still the most probable emer-

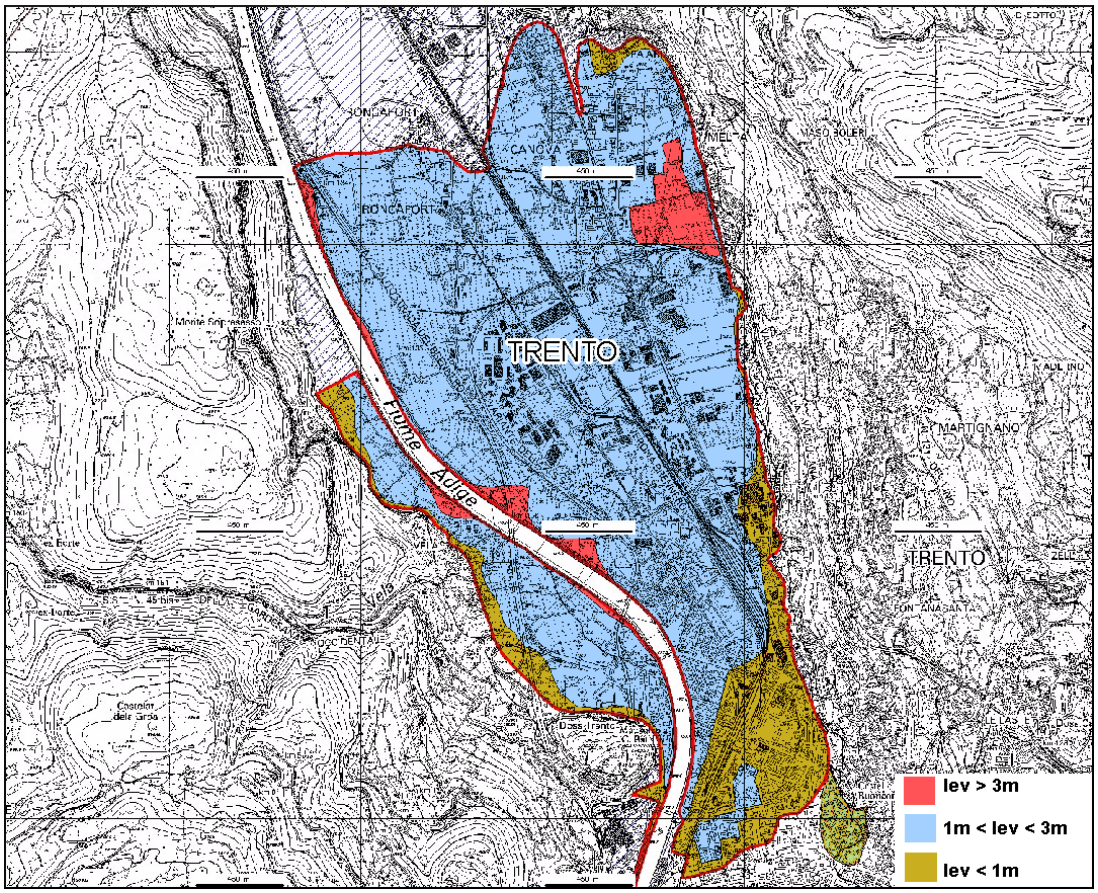


Figure 5.1: Flooding areas, Trento town

gency event in Trento. Therefore, here, we will focus on such a flooding emergency in this town.

Trento is a city situated in the north of Italy and its area covers about 158 km^2. Its territory is mainly composed by mountains (altitudes from 181 meters to 2090 meters over the sea). A lot of rivers pass through the city area and, among this, the most important river is the river Adige. Along this river, there is a high concentration of human activities and population. For these reasons, the main hazard for Trento is represented by a flooding of the river Adige and its related fluvial network.

5.1.2 The emergency plan

The emergency intervention plan of the Autonomous Province of Trento (PAT) is composed by different parts. The emergency plan contains the goals of the plan, the detailed description of its phases, the main actors involved in the e-Response and the description of the PAT Geographic Information System useful to support the activity in the case of disaster management.

The main goal of the municipality emergency plan [100] is to organize the evacuation of the population. In Trento town the resident population in 2001 was about 104.000 individuals. The potential number of persons affected by a flooding of class 2 (see previous section) is estimated around 19.000 (19% of the total residents). About 2.000 are older than 70 years. As we observed in §4.2, GIS data are crucial during e-Response activities. Specifically, in this situation the knowledge of the state of the road structures affected by the flooding event is fundamental. The information about the roads permits to the civilian protection actors to choose the proper road structures for the evacuation plan. For the Trento province the most important ways of communication are situated along the Adige valley (*Brennero Highway*, *Brennero railroad*, and *SS 12 road*), so they are subjected to a high flooding hazard, too. All public buildings that are contained in the flooding area are considered critical sites and potential risk factors since they might contain a high number of persons. It is therefore mandatory to have a census of such buildings (such as technical offices, libraries, schools, churches, museums, hospitals, etc.) and to locate the affected buildings. Also here, the primary goal is to evacuate the persons in such buildings effectively and rapidly.

As a secondary goal, it is mandatory to preserve cultural and historical heritages affected by the flooding event. Knowledge about such sites and

their content, located in the flooded area, could permit the relocation of the main assets in more secure places. Moreover, knowledge about the service infrastructures (such as the electricity network, the waterworks network, the pipeline network, the telecommunication network, etc.) is of uttermost relevance during emergency events.

In an emergency situation in Trento, there are two main levels of coordination: the provincial level and the municipality level. Only in cases of extensive emergencies, other levels have to be coordinated (national level, European level, international aids) with the province and municipality levels. For the case of our scenario, flooding emergency in the city of Trento, our scope is limited to the above two main levels. In the case of such an emergency a PAT (provincial level) coordination center is the responsible institution for the e-Response activities.

The emergency plan of Trento contains, among others, information about emergency procedures; numbers of residential people for each area; list, number, and the location of the public common structures (churches, schools, public offices, pools, etc); list and position of hazard multipliers (gas stations, factories, supermarkets, garages, etc); list and numbers of storehouses (equipments, materials, etc); evacuation centers (school buildings, sport centers, institution buildings); people meeting points (list and locations).

Figure 5.3 illustrates a simplified overall organizational schema of the PAT emergency coordination plan.

The main physical actors indicated in the current PAT emergency plan include [99, 100, 137]:

- Provincial Emergency Coordination Center (PECC). This coordination center is coordinated by the chief of the Civilian Protection and Fire department (emergency coordinator). It has the responsibility to coordinate all the others organizations using the information re-

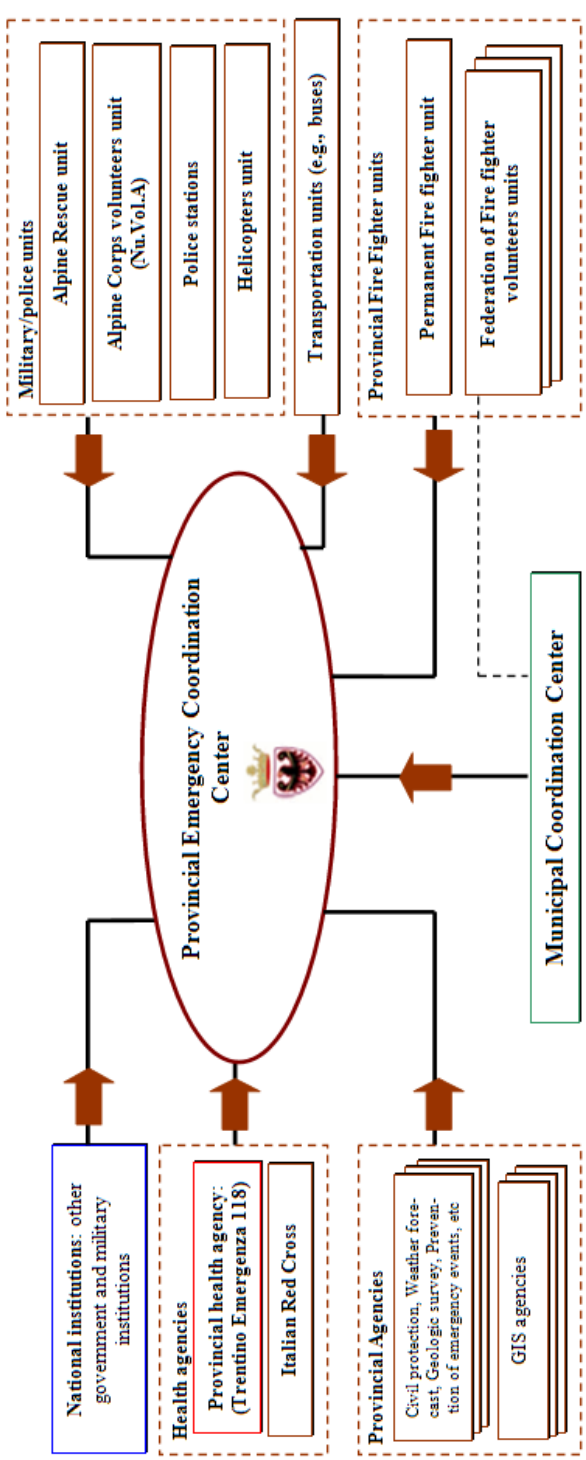


Figure 5.2: Organizational schema of emergency coordination in the Province of Trento

ceived by the municipality coordination centers as well as by institutional emergency signalling channels (e.g., weather forecast service, sensor networks, fire fighters reporting, etc.). It must receive those information and it has to decide whether and when to activate other emergency actors involved in the event situation.

It coordinates all others provincial agencies. The most important (in case of e-Response) agencies are:

- *The Civilian Protection and Fire* agency, that is the main structure coordinated by the PECC. This agency includes Provincial Fire fighters units (ca. 5886 fire fighters) subdivided into two kinds of organizations:
  - The Permanent Fire Corps: ("Corpo Permanente VVF di Trento"), ca. 169 individuals (at the time of this work).
  - The Volunteers Corps: provincial federation of about 240 corps (at the time of this work), approximative one corps for each municipality within Trento province.

  The activities of this agency are mainly based on the Civilian Protection and Fire database: it contains much information such as data concerning Fire corps in Trento province (general data such as people, phone numbers, photos, addresses, etc), the local military station ("Carabinieri"), the forestry stations, the emergency resources (tanks, rafts, etc) and their location, the hydrants map, etc. At present, such a database is logically distributed but physically centralized. Every Fire Corps can read all the information and must update information related to its corps.
- *Other provincial agencies* that are involved into the e-Response activities, such as weather forecast, geologic survey, prevention of emergency event, dam office, etc.

  - *GIS agencies* that provide geographical information about the emergency area.

- PECC coordinates the municipalities involved in the e-Response. For each municipality a Municipality Coordination Center (MCC) needs to be formed in a very short time. Then, MCC follows activities contained in the municipal emergency plan. Specifically, the MCC is the organization responsible for the e-Response in the municipality of Trento. It has to coordinate its directives with PECC activities. Moreover, it controls the Volunteers corps of the Trento municipality. MCC main actors include:

  - The municipality Mayor (coordinator of the MCC).
  - The Municipality Volunteers Fire Unit.
  - Other municipality resources and structures.

- PECC also coordinates other institutions, including:

  - National ("Italian red cross") and local health agencies ("Trentino Emergenza 118").
  - Other government and military institutions at national level: they collaborate with provincial institutions on explicit call.
  - Military and police station units, including:
    * Alpine rescue unit.
    * Alpine corps volunteers unit ("Nu.Vol.A.").
    * Helicopters unit.
    * Police stations.
  - Transportation units, e.g., buses used to evacuate people in the case of flooding emergency.

- Citizen of the municipalities involved in the emergency event.

We can treat the environment in terms of abstract computational resources. The main resources and services that supports the emergency plan include:

- SDI services which provide GIS services and maps for the Trento province. In our scenario the SDI is a distributed system composed of about ten similar GIS agencies (civilian protection, urban plan, forestry, agriculture, geologic survey, public works, environmental protection, public water management, and cadastral). Every GIS agency is responsible for the management of a subset of PAT GIS data and services.

- The network of hydrographic stations (sensors) reports the status of the hydrography. At the moment of this work the number of the stations is 474 and the update time of each sensor information is around 15 minutes. Different sub-networks compose this network, including:

  - Main controlled network: about 100 stations connected (real time) via radio (radio bridges) and GPRS.
  - Secondary sub-networks: provided by the weather forecast local office ("Meteo Trentino").
  - Electricity agencies sensors (provided by national and local companies: "ENEL", "Edison", "Trentino Servizi") to monitoring water level for each dam.
  - Information provided by boundary regions and institutions ("Veneto" region, "Magistrato della acque di Venezia" institution, "Friuli Venezia Giulia" region, "Bolzano" province).

- The weather forecast service reports the weather conditions and the forecast for the next 6-12 hours. This information, connected to the water level of the river Adige, is used to enact the emergency plan of the Trento province.

- Other local repositories: it is reasonable and realistic to suppose that most actors involved will possess and maintain contextual information about emergency procedures: some of these will be from previous emergencies, others will be stored during the current emergency. The majority of such information is probably a duplication of information existing in other resources (however not entirely); nevertheless it is useful to consider it and share it during the emergency.

5.2 Evacuation use case

In this chapter we focus on the activities related to people evacuation from the areas interested by the flood. In particular, we individuated emergency peers (a subset of the actors we presented previously, e.g., firemen, police, medical, bus/ambulance agents, etc.), the main organization involved (e.g., Provincial Emergency Coordination Center, Fire agency, Civilian Protection unit, Health agencies, etc.), a hierarchy between the actors (e.g., emergency chief, subordinate peers, etc.), service peers (e.g., water level sensors, weather forecast services, SDI services, route service, etc.) and a number of possible interactions among the peers and their assigned tasks.

The peers can be distinguished into two main categories: *service peers* and *emergency peers*. While the formers are basically peers providing services under request, the emergency peers are often acting on behalf of emergency human agents that are in charge of realizing the emergency plan.

Preliminary analysis of the scenario and description of the peers and tasks

can be found in [137, 139, 83]. The work in [135] contains the description of the implemented scenario. Specifically, this work contains an exhaustive evaluation of the OpenKnowledge system both in centralized and decentralized information gathering strategies.

Figure 5.3 recalls the richness of all scenarios and interactions possibly involved in the case of evacuation of people interested by the probable flooded areas. The upper part of the figure represents the *pre-alarm* phase of the emergency plan foreseen by the PAT. The bottom part relates to the *evacuation* phase.

In Figure 5.3 black dots represent the peers involved in our emergency use case, white dots represent activities propagation, and edges represent activities committed by the peers. In the following, we are going to describe a relatively simple possible sequence of activities that can be enacted between the roles we described in the previous sections.

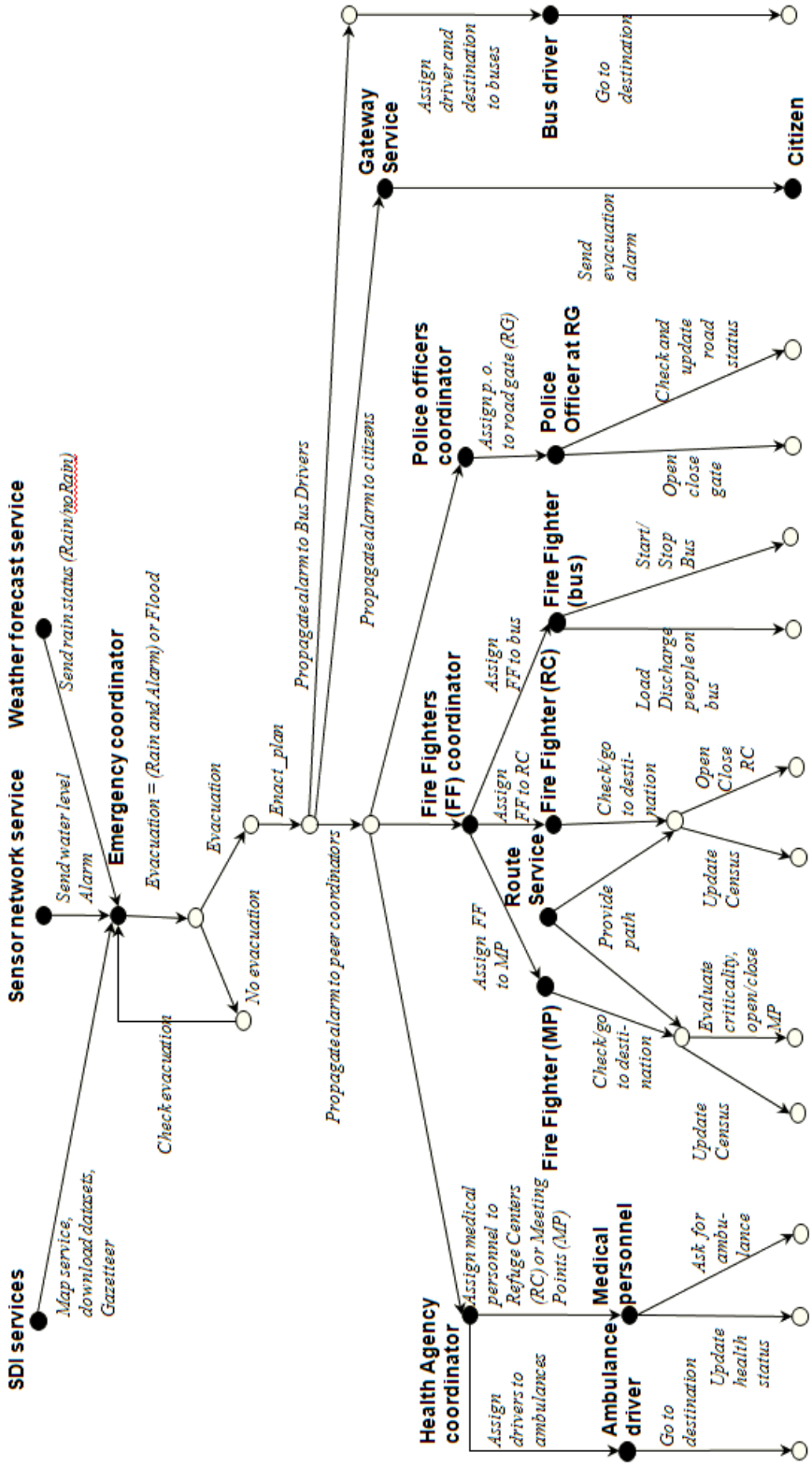


Figure 5.3: The overall e-Response use case

5.2.1 Pre-alarm phase

In the pre-alarm phase are mainly involved service peers which are peers providing all the information needed to enact the emergency plan or not. The pre-alarm phase thus involves mainly service peers which provide information useful for decision making. The pre-alarm phase eventually results in the evacuation phase. Such a phase regards all the activities needed to move people to safe places. In such a phase, the key peers are emergency peers, that is, all the peers in charge of helping in the evacuation of citizen: emergency coordinators, fire fighters, government agencies (e.g., civilian protection department), real-time water level data reporters (e.g., people, sensors). Of course, the emergency peers are supported by service peers such as SDI services, weather forecast services, route services, sensors scattered across the emergency area, etc.

We suppose that the river Adige is near to produce a flooding in Trento town and that the emergency activities are enacted by the emergency coordinator. The emergency coordinator, continuously requires information from the SDI services and evaluate the emergency risk by both evaluating the sensor network alarms and the weather forecast. Based on the information about the status of the area (e.g., by visualizing a map), on the water level (from the sensor network) and on the weather forecast, the emergency coordinator can enact the second phase of the emergency plan: the evacuation phase. Obviously the evacuation phase must be immediately enacted when the river brakes its banks and the water overruns populated areas.

5.2.2 Evacuation phase

To start the evacuation phase the emergency coordinator sends the evacuation directives to other organizations and coordinators. The final goal is to bring the people outside the areas interested by the flood. Specifi-

cally, people are requested to meet at specific meeting points (MP). Then, citizens will be brought from meeting points to refuge centers (RC) outside dangerous areas. Specific means (e.g., buses) are going to be used to fetch the people at MPs and to discharge them at RCs. The emergency plan indicates the routes that an emergency vehicle has to follow when the evacuation phase is enacted. Moreover, along these routes some road gates (RG) are identified. Police officers control RGs in order to check coming and going in flooded areas during the emergency event. Specifically, during the evacuation phase:

- The emergency coordinator propagates the evacuation directive to bus drivers and, at the same time, assigns bus and destinations to each bus driver. Each bus driver must drive to its starting point (usually a MP), load people on the bus, check the path (roads blocked, new roads, etc.) he/she has to follow, drive to a RC, discharge people at RC and then return to a MP.

- The emergency coordinator propagates the alarm to a gateway service. This service, in turn, collects information about individuals and sends the evacuation alarm to them using both traditional (television, radio, megaphones, etc.) and technological (email, SMS, OpenKnowledge messages, etc.) communication means. At the same time, the gateway service sends the list of the MPs that each individual has to reach. As soon as possible, each person has to reach the nearest MP.

- Police officers coordinator, after receiving the evacuation alarm, assigns each RG to a police officer, that, in turn, has the responsibility to open or close the RG and to check and update roads status.

- The emergency coordinator propagates the evacuation alarm to the Fire fighter coordinator. The Fire fighter coordinator assigns destinations (MP, RC, or bus) and tasks to fire fighters.

- The fire fighter that has to reach a MP asks the route service the best path from its location to the MP, checks the path and then reaches the MP. The work in [135] illustrates an exhaustive description of this particular use case and implements a simulated environment of the emergency event. At the MP fire fighter main tasks include: the update of the census (people at the MP) information, the evaluation of the criticality of the meeting point (e.g., imminent flooding, water level, etc.), and the responsibility to open or close the MP.

- The fire fighter whose destination is a RC asks the route service the best path from its location to the RC, checks the path and then reaches the RC. At the RC the fire fighter has to update census information and to decide if the RC has to be closed (e.g., when it is full).

- The main activities of a fire fighter on a bus are related to load people on a bus at a MP and to discharge people at a RC.

- The health agency coordinator must assign ambulances to drivers that, in turn, check and reach their destination. Moreover, the health agency coordinator assigns medical personnel to RCs or MPs that, in turn, health patients, update their health status and, eventually, ask for an ambulance.

5.3 SDI service coordination

Within the e-Response scenario presented in the previous section we present here a subset of the activities related to some selected SDI services. In particular, we have analyzed the organizational model of the distributed GIS Agency infrastructure of Trento province. The framework is represented by a number of specialized GIS agencies, such as, civilian protection, urban plan, forestry, agriculture, geologic survey, public works, environmental

protection, public water management, and cadastral. Each GIS agency is responsible for providing a subset of the geographic information for the local region. To support interoperability among the different GIS agencies the regional information infrastructure is shifting from a traditional GIS to a distributed SDI.

In Trento province the SDI is managed by an institution named Environmental and Geographical Information System (SIAT). It is responsible for the management of all geographic information in Trento province. SIAT is divided into different agencies. As said before, each agency is responsible for a subset of the datasets, so the Geology Survey agency is responsible for geological datasets, the Urban Planning agency is responsible for the urban planning cartography, and so on. Some datasets are defined as *basic cartography*, in the sense that these datasets are the base on which all the other datasets (*thematic* datasets) are built upon. The basic cartography contains, for example, the aerial digital photos, the topographic map, the elevation points, the digital terrain model, the administrative boundaries, etc. Examples of thematic datasets are the geology risk, the natural parks, the location of the hydrants, the location of the schools, etc. Every agency produces its datasets, and provides a number of GIS services[1].

In the following, we focus on some of the most commonly used specific use cases, i.e., gazetteer service, map request service, and download service.

5.3.1 Overall description of the SDI scenario

In [139], we presented our proposed service oriented architecture for the implementation of a concrete cluster of services in order to obtain either a map or a geographical dataset (see Figure 5.4). To this end, the user first inputs a location identifier, usually a string. If the name of the place is recognized by the system, the system returns its (geographical) position

[1] `http://www.territorio.provincia.tn.it`

or a list of possible - similar - locations. Thus, the user can refine his/her (geographical) query and proceed to query either for a map request or a for a download request.

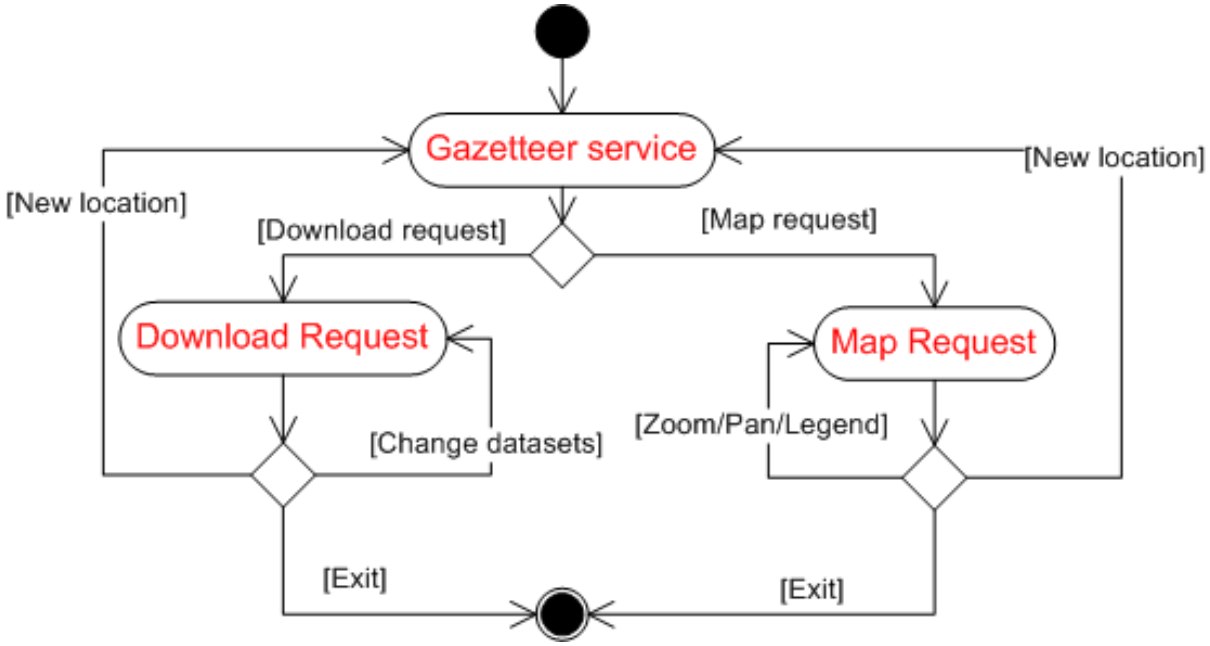


Figure 5.4: Overall Architecture for Map Request Service

We separate the main flow, depicted in Figure 5.4 into three individual flows, the gazetteer service, the map service and the download service:

- **The gazetteer service.** The user searches for a string in the system toponym repository. The gazetteer service returns a list of place names that contain the input string. The user chooses one of the location names and asks the system for the position. The system outputs the geographic coordinates of the toponym. Figure 5.5 shows the activity diagram for the gazetteer Service.

  Note that this is a simplified schema. A gazetteer could be improved by adding additional features such as, for instance:

  – Returning geographical location in a different geographic coordi-

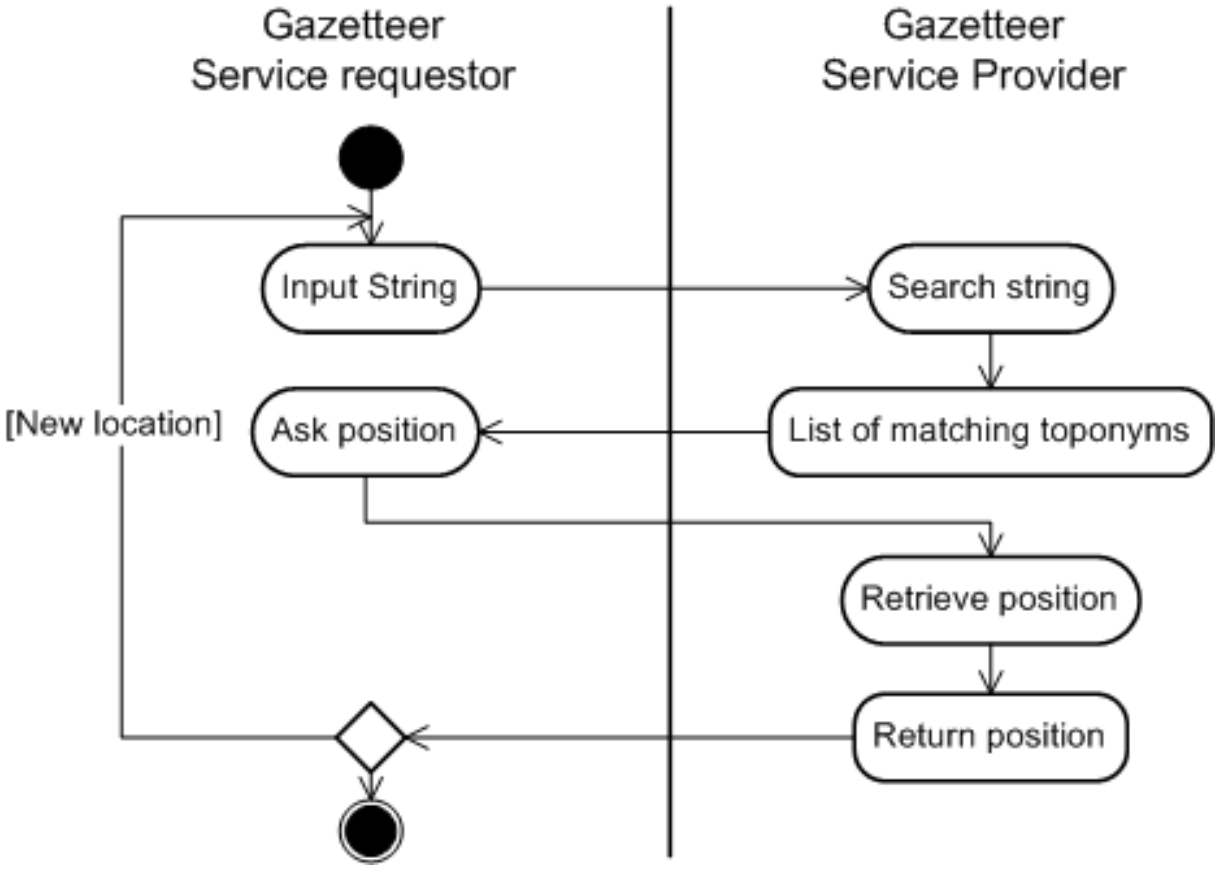


Figure 5.5: Activity Diagram for the Gazetteer Service

nate system. In this case the system should be able to address a web service that transforms geographic coordinates into the final reference coordinate system.

– Extending the search by adding description of places, such as churches, restaurants, public services, shops, etc. In this case the system has to search for general classes of a objects and a semantic matching system could improve the results (e.g., transforming classes into equivalent concepts).

- **The map service.** Usually, a map service requestor needs to visualize a map of a region with geo-referenced information selected by a user.

In this case, the searched map is a composition of different geographic layers offered by a GIS service provider. The user asks for a map. He/she gives the Map Provider Service the coordinates of the center of the map (the toponym position), the precision scale, and the layers he/she wants to visualize. The map provider computes the boundary of the map and builds the digital map. Finally it returns the map to the requestor. Figure 5.6 shows the activity diagram for the map request service.

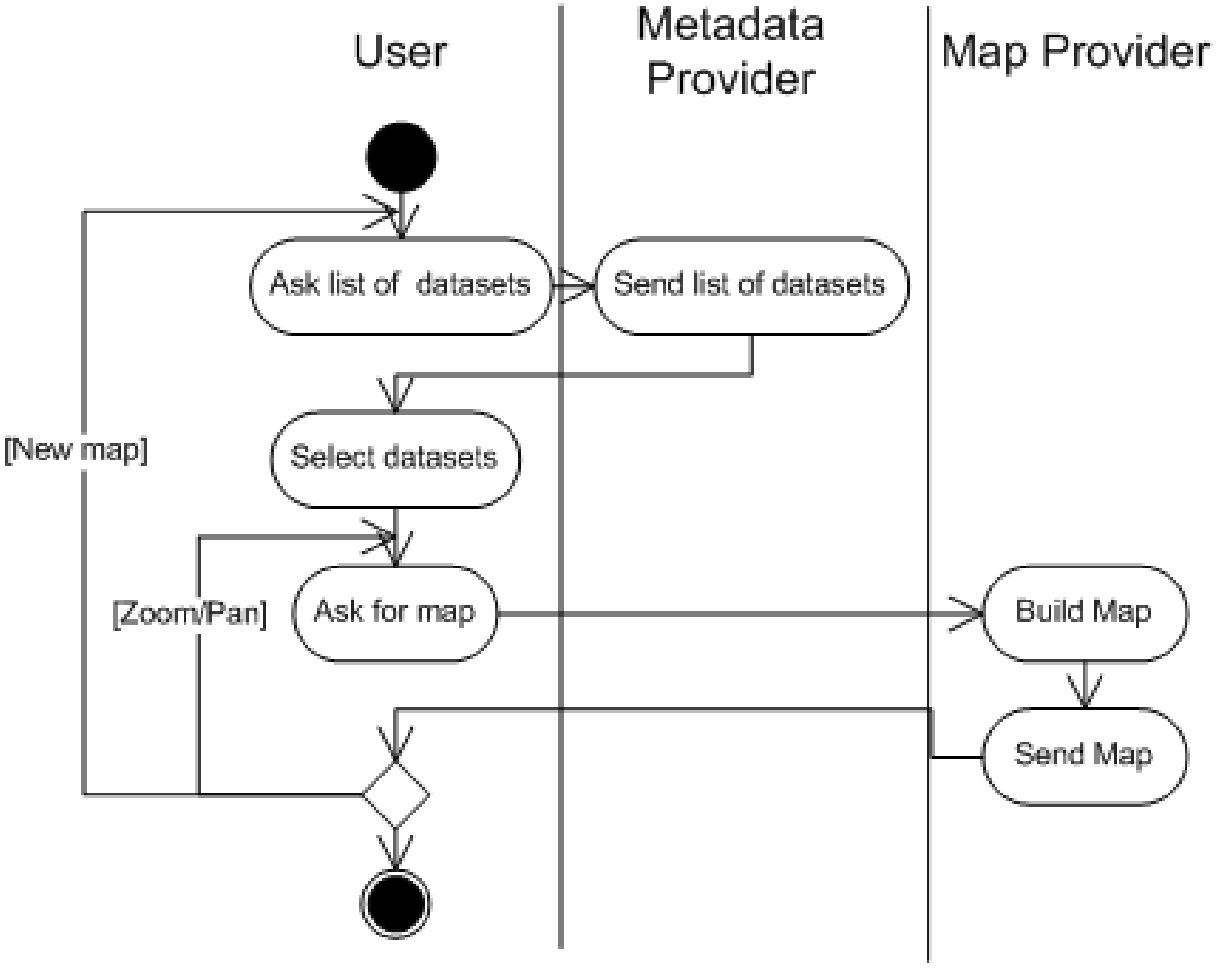


Figure 5.6: Activity Diagram for the Map Request Service

- **The download service.** The user can ask for geographical datasets

stored into the system. The user first selects the layers he/she wants to download, then the dataset provider sends these layers to the users. In our case the interchange format is assumed to be the either in GML, KML[2], or in ESRI Shape[3] formats. Figure 5.7 illustrates the activity diagram for the download service request.

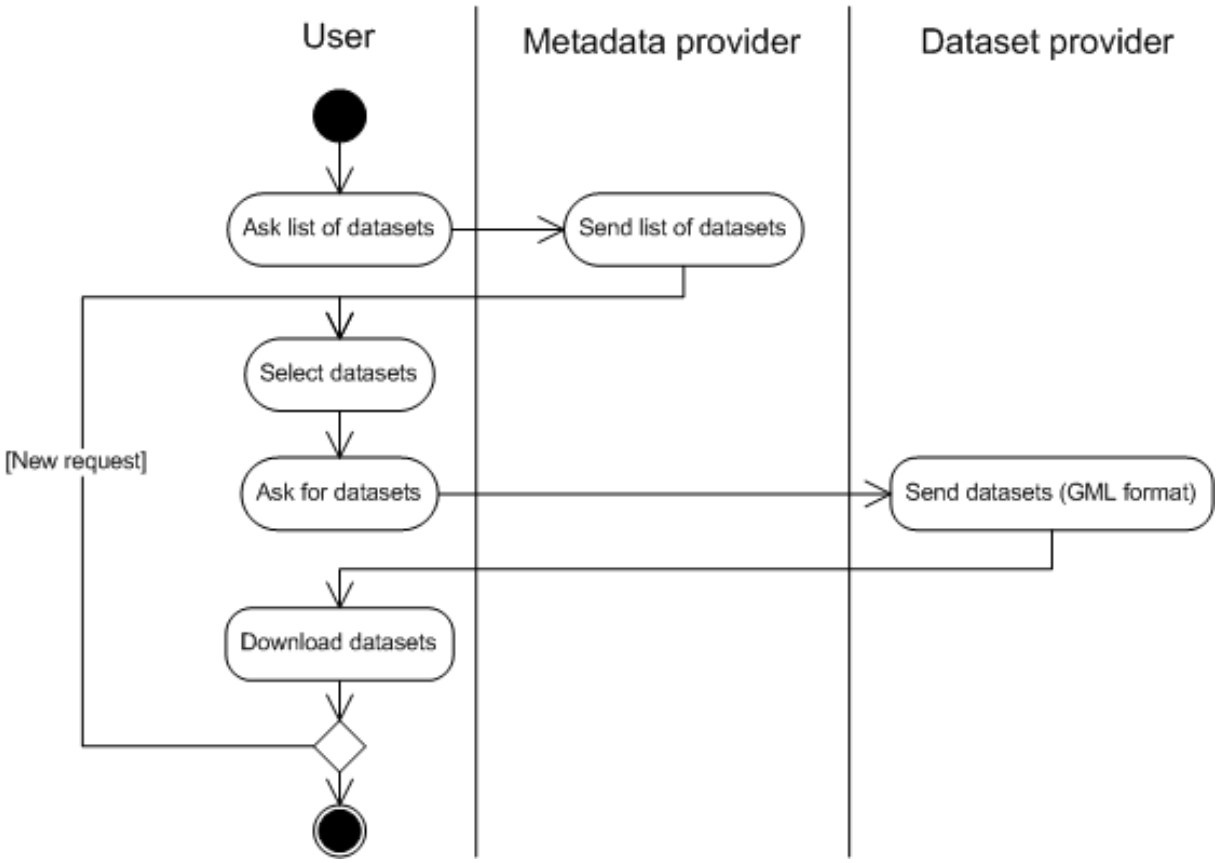


Figure 5.7: Activity Diagram for the Download Request Service

5.4 Summary

In this chapter we gave a presentation of the overall motivating scenario for the P2P semantic matching framework we use in this thesis. In particular,

[2] http://code.google.com/apis/kml/documentation/

[3] http://www.esri.com/library/whitepapers/pdfs/shapefile.pdf

we illustrated a natural disaster scenario, i.e., a possible flooding event in Trento. Within this scenario we described the activities related to people evacuation from probable flooding areas to refuge centers located outside these areas. Moreover, we specified three SDI services relevant for our scenario.

In the next chapter we will describe the framework we used to support our scenario, i.e., the OpenKnowledge system. This system provides a P2P infrastructure needed to run interaction between the peers involved in the e-Response scenario.

Chapter 6

Supporting the scenario: the OpenKnowledge system

In this chapter, we present the main characteristics of the OpenKnowledge system[1] developed within the FP6 OpenKnowledge EU project, which provides the underlying P2P infrastructure needed to solve the specific requirements depicted in the previous chapter.

The OpenKnowledge system allows peers, on an arbitrarily large P2P network, to interact productively with one another without any global agreements or pre-run-time knowledge of who to interact with or how interactions will proceed. Within this environment, we distinguish *functional* knowledge from *content* knowledge. *Content* knowledge is the data that is shared on the network, and that is queried by peers. These may be documents, pictures, music, computational services, etc. *Functional* knowledge is information about the functionality of services, and the mappings and interactions between them required to use the content knowledge on the OpenKnowledge system. OpenKnowledge provides mechanisms and tools that hide the complexity of such *functional* knowledge [125]. Specifically, the core concepts in OpenKnowledge are [116]: (i) the interactions between peers, defined by *interaction models* (the functional knowledge) published

[1] `http://www.openk.org`

by the authors on a P2P infrastructure with a keyword-based description; and (*ii*) a distributed infrastructure, denoted as *OpenKnowledge Kernel*, that supports the publishing, discovery, execution, monitoring and management of the various interaction models.

The OpenKnowledge system has at its core a mechanism for sharing models of activities that require interaction across the Internet. We refer to such models as *interaction models* (IMs). We expect that communities of practice will naturally form around collections of IMs and that these communities can be stabilized by a mechanism for rapidly propagating IMs of common interest across peer groups. Notice that this is explicitly an *interaction-centered* approach to knowledge sharing, as opposed to the traditional *data-centered* approach.

The system has been built with a completely distributed philosophy in mind, using P2P technology. Each peer that participates in the OpenKnowledge system will be running the platform we call the *OpenKnowledge Kernel* [27], that enables the basic functionality of finding interactions of interest as well as service providers for the execution of interactions. More precisely, the system is focused on efficiently sharing, discovering and executing these formally described IMs together with pointers to either the code for the services or peers that can execute the services.

In what follows, we contextualize the OpenKnowledge system by evaluating its similarity to other interaction-oriented approaches such as P2P and multi-agent systems, semantic service-oriented architectures, and grid-service models (§6.1). Then, we present the LCC protocol language used to implements IMs among distributed processes (§6.2). Next, we illustrate the OpenKnowledge system model (§6.3) and its architecture (§6.4). Finally, we describe the matcher module of the OpenKnowledge system (§6.5). This module is fundamental to our work, since it implements semantic matching between invocations of web services and web service descriptions.

6.1 The context

The need for open and reliable knowledge sharing is an issue that many systems have tried to address; in this section we embody the essential characteristics of those that share features with our own approach, as well as highlight their differences. Thus, we discuss some of the approaches that have also taken an interaction-oriented method: P2P systems, multi-agent systems, semantic service-oriented architectures, and grid-services.

P2P systems [70, 93, 5, 155] are very closed to the OpenKnowledge model. The central ideas of distributed storage, decentralized address register and symmetric roles of each peer are fully adopted by OpenKnowledge. The main differences between OpenKnowledge and P2P systems are that (*i*) OpenKnowledge aims at *service sharing* rather than *data-sharing* and (*ii*) that OpenKnowledge is a *semantic* P2P system which uses a meaningful description of the services each peer is providing to the network and uses a semantic matching approach to discover and compose services provided by peers.

Multi-agent systems (MASs) [151] are another class of related systems to OpenKnowledge. They both share the philosophy of a distributed sets of autonomous processes exchanging information. On the other hand, OpenKnowledge is different from MASs because (*i*) agents utilize highly structured architectures while OpenKnowledge peers are meant to be extensible by adding *plugins* and (*ii*) agents main characteristics are their pro-active nature, while an OpenKnowledge peers are reactive, as they only act when prompted by an external event. However, MASs and OpenKnowledge share the model of coordination because they both share the model of institutions and similar languages (i.e., LCC in the case of OpenKnowledge) which is normally used to define multi-agent interactions [149].

Semantic Service-Oriented Architecture (SSOA) [58] is an integrated

platform that facilitates discovery, composition and execution of services in a distributed, scalable and interoperable way, similar to OpenKnowledge. Moreover, they both rely on the assumption that, if services are semantically or syntactically described, they can also be searched and composed. However SSOA model presents some differences from the OpenKnowledge system such as: (*i*) for SSOA service building blocks are Web Services, while in OpenKnowledge are OpenKnowledge Components (OKCs, see §6.3) which invocation protocol is proprietary, (*ii*) SSOA aims at on-line composition of simple services into complex services whereas, OpenKnowledge adopts predefined workflows of services namely, *interaction models*, (*iii*) OpenKnowledge recruitment and execution of services is dynamically performed at runtime, possibly using a approximate semantic matching of services, while in SSOA, the advertisements of service are accompanied with the endpoint of the executor of a service, (*iv*) OpenKnowledge enables users not only to discover appropriate services for a given task execution, but also locate a number of shared task definitions (*IMs*) and (*v*) finally, usually SSOA architecture is centralized (with the exception of Meteor-S [146]) whereas OpenKnowledge architecture adopts a P2P architecture.

Grid-service model [40] often relies on SOA approach and is similar to the OpenKnowledge system in the sense that they both typically organize interactions in fixed workflows. However, OpenKnowledge system differs from Grid-Service model because the latter usually considers aspects that are missed in OpenKnowledge such as long-term stability of services, provenance, quality of service and resource monitoring. Moreover, as in the case of SSOA, Grid-service model advertisement is accompanied with the endpoint identification of the services and finally, in Grid-service advertising systems are usually centralized while in OpenKnowledge they are fully distributed.

In summary, OpenKnowledge includes the following set of selected char-

acteristics:

- OpenKnowledge is an *interaction-centric* approach, as peers share models of activities (*IMs*) that require interaction across the Internet.
- OpenKnowledge relies on a *semantic P2P* approach, because it uses distributed storage, decentralized address register, it provides symmetric roles of each peer, and it uses a semantic matching approach to discover and compose services provided by peers.
- OpenKnowledge provides a service choreography mechanism where protocols are defined by using the LCC language (see next section).
- OpenKnowledge enables users not only to discover appropriate services for a given task execution, but also to locate a number of shared task definitions.

6.2 Lightweight Coordination Calculus

In this section, we describe LCC [115], a choreography language employed to specify peer interactions supported by the OpenKnowledge system.

6.2.1 LCC basics

LCC is a protocol language used to describe interactions among distributed processes, e.g., agents and web services. LCC can be considered as a heavily-sugared variant of the π-calculus [92] with an asynchronous semantics. The extensions to the core calculus are designed to make the language more suited to the concepts found in multi-agent systems and dialogues. The formal basis is the primary reason that we have chosen to use LCC over

more popular languages, such as WS-Coordination[2], BPEL4WS[3], and the OWL-S process model. LCC was designed specifically for expressing P2P style interactions within multi-agent systems, i.e., without any central control; therefore, it is well suited for modeling coordination (choreography) of software components running in an open environment. The abstract syntax of LCC is presented in Figure 6.1.

$$
\begin{array}{rcl}
\mathit{Framework} & := & \{\mathit{Clause}, \ldots\} \\
\mathit{Clause} & := & \mathit{Role} :: \mathit{Dn} \\
\mathit{Agent} & := & a(\mathit{Type}, \mathit{Id}) \\
\mathit{Dn} & := & \mathit{Agent} \mid \mathit{Message} \mid \mathit{Dn\ then\ Dn} \mid \mathit{Dn\ or\ Dn} \mid \mathit{Dn\ par\ Dn} \mid \mathit{null} \leftarrow C \\
\mathit{Message} & := & M \Rightarrow \mathit{Agent} \mid M \Rightarrow \mathit{Agent} \leftarrow C \mid M \Leftarrow \mathit{Agent} \mid C \leftarrow M \Leftarrow \mathit{Agent} \\
C & := & \mathit{Term} \mid C \wedge C \mid C \vee C \\
\mathit{Type} & := & \mathit{Term} \\
M & := & \mathit{Term}
\end{array}
$$

Where *null* denotes an event which does not involve message passing; $Term$ is a structured term and Id is either a variable or a unique identifier for the agent.

Figure 6.1: Abstract syntax of LCC.

Interactions in LCC are expressed as message passing behaviors associated with roles. The most basic behaviors are to send or receive messages, where sending a message may be conditional on satisfying a constraint (precondition) and receiving a message may imply constraints (post-condition) on the peer accepting it.

There are five key syntactic categories in the definition, namely: *Framework, Clause, Agent, Dn* (Definition)*, and Message.* These categories have the following meanings. A *Framework*, which bounds an interaction in our definition, comprises a set of clauses. Each *Clause* corresponds to an agent,

[2] http://docs.oasis-open.org/ws-tx/wscoor/2006/06

[3] http://www.ibm.com/developerworks/library/ws-bpel/

which is the name that we give to an interacting component. Each agent has a unique name *a* and a *Type* which defines the role of the agent. The interactions, that the agent must perform, are given by a definition *Dn*. These definitions may be composed as sequences (*then*), choices (*or*), or in parallel (*par*[4]). The actual interactions between agents are given by *Message* definitions. Messages involve sending ($\Rightarrow$) or receiving ($\Leftarrow$) of terms M from another agent, and these exchanges may be constrained by C.

A basic LCC interaction is shown in Figure 6.2.

$$
\begin{array}{l}
a(r1, A1) :: \\
\quad ask(X) \Rightarrow a(r2, A2) \leftarrow need(X)\ then \\
\quad update(X) \leftarrow return(X) \Leftarrow a(r2, A2) \\
\\
a(r2, A2) :: \\
\quad ask(X) \Leftarrow a(r1, A1)\ then \\
\quad return(X) \Rightarrow a(r1, A1) \leftarrow get(X)
\end{array}
$$

Figure 6.2: LCC example: double arrows ($\Rightarrow, \Leftarrow$) indicate message passing, single arrow ($\leftarrow$) indicates constraint satisfaction.

The peer identified by the value of the variable *A1* playing the role *r1* verifies if it needs the info X (pre-condition *need(X)*); if it does, *A1* asks the peer identified by the value of the variable *A2* for X by sending the message *ask(X)*. *A2* receives the message *ask(X)* from *A1* and then obtains the info X (pre-condition *get(X)*) before sending back a reply to *A1* through the message *return(X)*. After having received the message *return(X)*, *A1* updates its knowledge (post-condition *update(X)*).

The constraints embedded into the protocol express its semantics and could be written as first-order logic predicates (e.g., in Prolog) as well as

[4] `not yet implemented in the OpenKnowledge interpreter`

methods in an object-oriented language (e.g., in Java). The characteristic of modularity allows separating the protocol from the agent engineering. While performing the protocol, peers can therefore exchange messages, satisfy constraints before/after messages are sent/received and jump from one role to another so that a flexible interaction mechanism is enabled still following a structured policy, which is absolutely necessary for team-execution of coordinated tasks.

6.3 Model of the system

OpenKnowledge is a community that any user can join, the only requirement being the use of the OpenKnowledge system. Each user interacts with the system and other users as a *peer* via the OpenKnowledge *Kernel*, a software that provides the low level protocols required for such interactions. Other than the peers there is another important participant in the OpenKnowledge system, this is the *Discovery and Team Formation Service* (DTS). The DTS is a repository of content, and it is used to publish, discover and retrieve IMs and OKCs. Moreover, DTS coordinates subscription, it chooses a coordinator and it provides team formation and interaction initialization. DTS functionalities are elaborated in the following.

Users can download the OpenKnowledge Kernel from the OpenKnowledge website[5] and install it on their computers. The basic installation provides limited functionality but it can be extended by installing more plug-ins or *OpenKnowledge Components* (OKCs). Each OKC contains a set of functions that provides a result when given a particular input. OpenKnowledge users can develop their own OKCs or discover existing ones via the DTS, where OKC developers are allowed to *publish* them.

[5] `http:www.openk.org`

Interactions among peers are regulated: they follow specific protocols which are defined via IMs. IMs are high level protocol definitions in which no reference to specific peers is made so that they may be reused. Usually IMs are specified by the LCC language and define how abstract *roles* may interact with one another (see, for example, Figure 6.2). Interaction amongst roles happens via messages, the specific form of which is specified in the IM. For each role the IM specifies the order and possible options in sending or receiving messages to or from other roles. The IM also specifies how the values in the messages are to be calculated. One way to calculate these values is through *constraints* which must be executed by the given role prior to message sending or on message reception. These IMs are not fixed, any OpenKnowledge user may develop new IMs and publish them to the discovery service.

Peers interested in playing some role in an IM have to initially *subscribe* to do so with the DTS. The DTS registers the peer identifier, the role it wants to play, and the IM in which it wants to play it in. Since all IMs are stored in the DTS, users can search through it in order to find which IM suits them best and which role they want to play. When enough peers have been subscribed to play all the necessary roles in an IM, then the interaction may start. The DTS is responsible for gathering this information, so it is in charge of starting the interaction process.

An IM defines how peers are to interact. These rules are enforced by a peer playing the special *coordinator* role. The kernel is provided with the functionality to become a coordinator. At start up, a peer may subscribe itself as a coordinator with the DTS. When the DTS has enough interaction subscriptions for an interaction to begin, it chooses a coordinator to manage the interaction and the set of all subscriptions to the specific IM is sent to the coordinator. This is where the *interaction bootstrap* begins.

Coordinators rely on an *interpreter* to process and regulate execution of

IMs. When the interaction starts, the coordinator simulates the interaction among peers via proxies. Peers do not communicate with one another directly, instead all communication goes through the coordinator. Messages are sent from one proxy to another within the coordinator. Peers are only involved in the process when the interpreter encounters a constraint which it cannot solve. In that case the coordinator contacts the peer playing the role for the given constraint in order for it to be solved.

A peer, when asked to solve a constraint, must rely on the functionality of its OKCs. It is up to the peer to decide (with help from the user) which of the functionalities provided by the OKCs it manages is best for solving the constraint. The process of interpreting the IM and asking peers to solve constraints goes on until the IM reaches its end. At this point the peers are informed so that they may free resources that are no longer needed.

Figure 6.3 depicts the OpenKnowledge model, showing the relationships between the different parts and concepts that have been presented above. Peers are geared towards two main activities:

- **Interacting** with other peers by playing a role in some IM, which they fulfill by using the functionality of their OKCs.
- **Coordinating** interactions among other peers, defined as IMs that peers can interpret.

Not shown in the figure is the DTS, which stores all the published IMs and OKCs for others to search and download, and acts as a blackboard where peers wanting to interact with others can subscribe to.

6.4 Architecture of the system

The OpenKnowledge system has a P2P architecture, implemented by the *OpenKnowledge Kernel* that runs on every peer. In this section we briefly

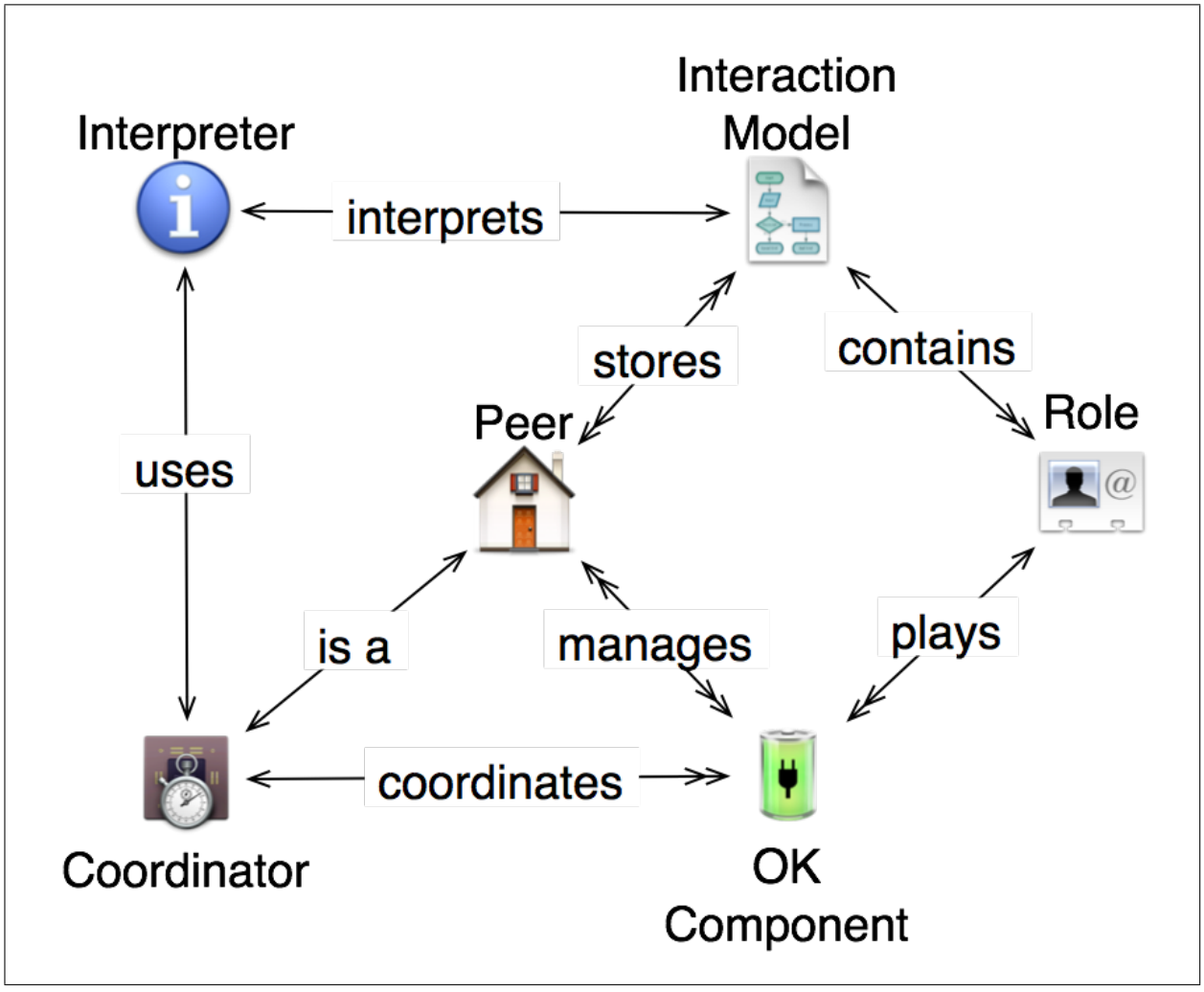


Figure 6.3: OpenKnowledge model

introduce the kernel's architecture. For a more detailed description we refer to [27]. The overall architecture is depicted in Figure 6.4.

Central to this architecture is the *control manager*, that provides execution control over the peer's modules and exposes a set of operations by combining functionalities provided by different modules. A *user-interface* is provided to access the basic OpenKnowledge functionalities: creating IMs and OKCs, searching for IMs, downloading OKCs and subscribing OKCs to roles. All communication is done by asynchronous messaging via the *communication layer*, so an *event tracker* is used in order to be able to track the conversational state for all requests.

Interactions are bootstrapped and run by a *coordinator*, which acts as an orchestrator for the interaction protocol, directing exchanged messages, communicating with involved parties and using the *interpreter* in order to parse and execute the IM at hand. Subscribing OKCs to roles defined in IMs goes through the *subscription negotiator* which in turn uses the *matcher* (see next section) in order to find the most appropriate OKC methods for roles. OKCs are stored locally at the peers in the *component repository*, while subscriptions are stored in the *subscription repository*. Please note that the aforementioned storage refers only to OKCs and subscriptions used locally by peers, but both are also persistently and transparently stored in the *storage service* layer of the DTS, accessed via the *discovery proxy*. Finally, a *trust* module is used in order to rank peers and artifacts based on a model for trust that takes into account previous experience.

As it is central for this thesis, in the next section we will describe the characteristics and implementation of the matcher module (SPSM).

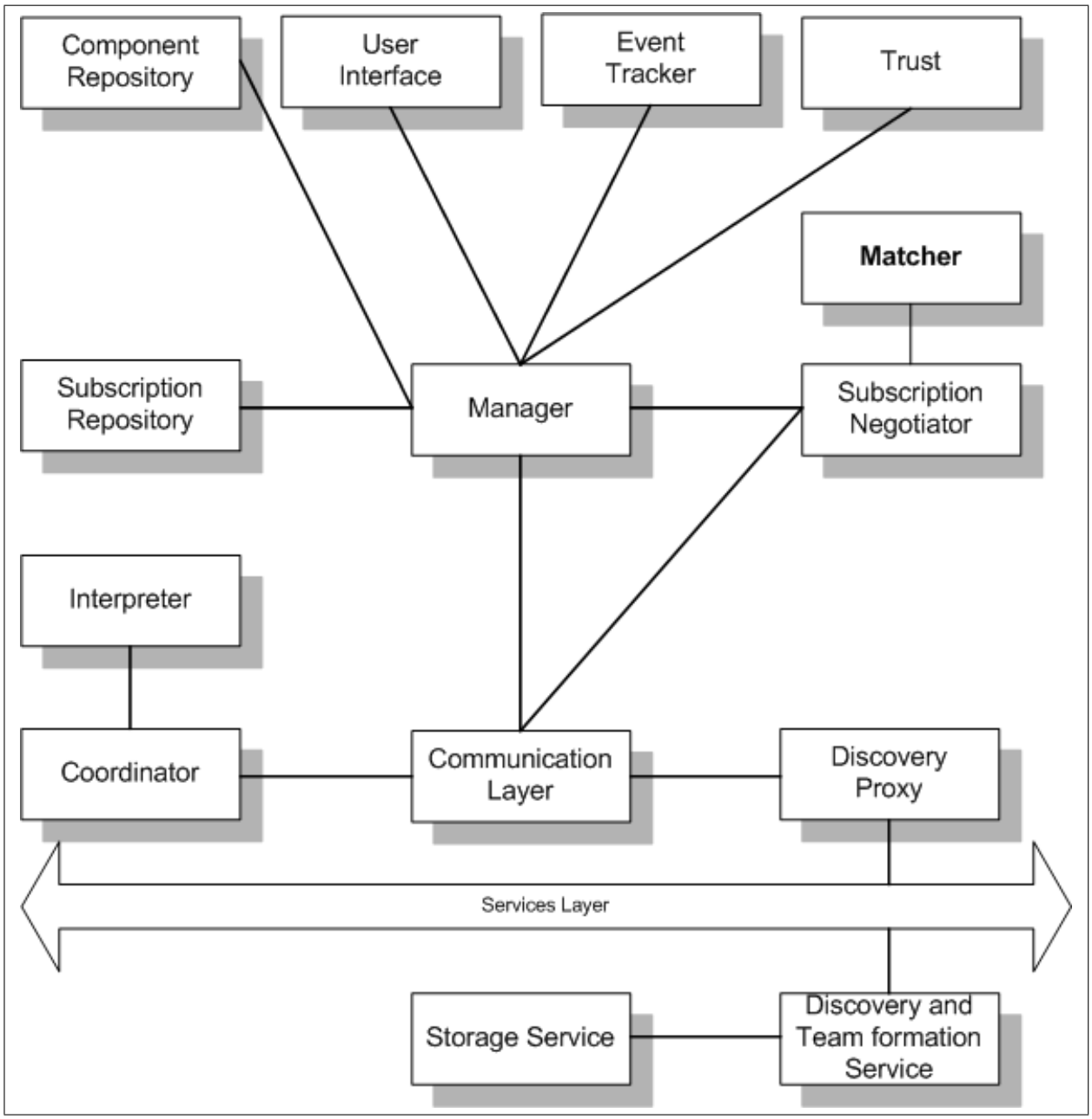


Figure 6.4: OpenKnowledge kernel architecture

6.5 The SPSM implementation

An OKC needs to understand the other OKCs it is interacting with. Chances are that an OKC will not interact with all the other OKCs, therefore defining a priori ontology seems unreasonable, given the complexity of the task. Furthermore, we want to achieve low entry cost, therefore, matching of one OKC's terms to others must be done at runtime. The matcher's aim is to aid in this process.

The matcher is used in the searching and integrating the interaction processes and services. When searching for IMs and OKCs it is used to map the text in their annotations to the query. When interacting, an OKC can use the matcher to map those terms that another OKC is sending to its own terms. The matcher taps into the information gathered from the system use, to provide community-supported mappings. The matcher module has been implemented by using a specific ontology semantic matching solution, namely Structure Preserving Semantic Matching (SPSM) [43]. In the following we will first briefly describe SPSM, then we will explain its behaviour and, finally, we will illustrate its implementation.

6.5.1 SPSM description

In our scenario peers are selected at run time and they can change every time. Let us suppose that we want to match a web service user description, such as: *requestMap(Version, Layers, Width, Height, Format, XMin_BB, YMin_BB, XMax_BB, YMax_BB)*, *T1* in Figure 6.5, with a web service operation description, such as: *requestMap(Dimension(Width, Height), Edition, Layers, DataFormat, Request, Xmin, Ymin, Xmax, Ymax)*, *T2* in Figure 6.5. These descriptions can be represented as tree-like structures.

As shown in Figure 6.5 the first description requires the second argument of *requestMap* operation (*Layers*) to be matched to the fourth one (*Layers*)

of *requestMap* operation in the second description. The value of *Version* in the first description must be passed to the second web service operation as the *Edition* argument. Moreover, *Request* (this parameter indicates which web service operation e.g., map service, download service, is being invoked) in *T2* has no corresponding term in *T1*.

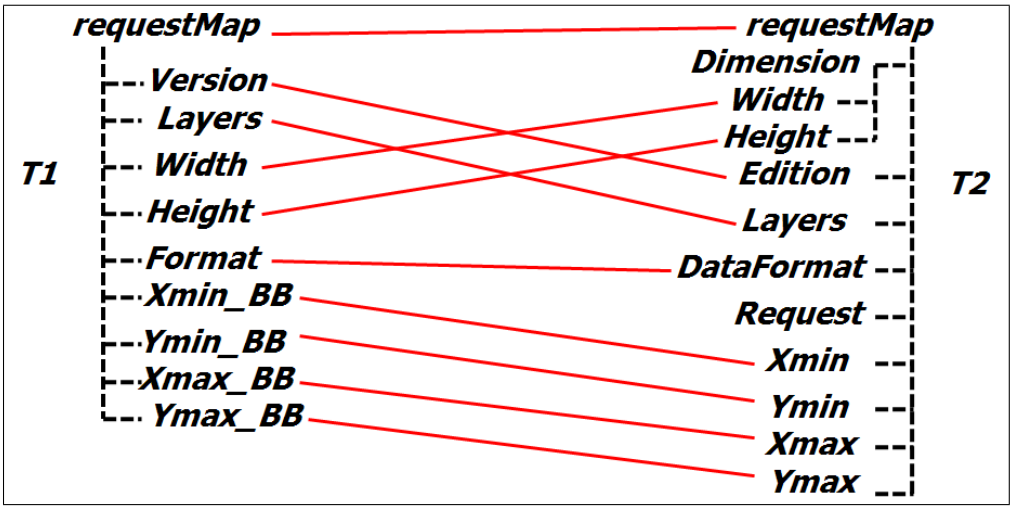


Figure 6.5: Two web service descriptions (trees) and correspondences (lines) between them.

The purpose of SPSM is to reduce semantic heterogeneity in web service user descriptions. Specifically, a semantic similarity measure is used to estimate similarity between web service user descriptions under consideration. This scenario poses additional constraints on conventional ontology matching. In particular, we need to compute the correspondences holding among the full tree structures and preserve certain structural properties of the trees under consideration. Thus, the goal here is to have a structure-preserving semantic matching operation. This operation takes two tree-like structures and produces a set of correspondences between those nodes of the trees that correspond semantically to one another, *(i)* still preserving a set of structural properties of the trees being matched, namely that functions are matched to functions and variables to variables; and *(ii)* only in

the case if the trees are globally similar to one another, e.g., $tree_1$ is 0.54 similar to $tree_2$ according to some measure.

6.5.2 SPSM approach

We briefly report here the SPSM approach (for completeness purpose see [43]) and present it with the help of examples from the GIS domain. We focus on tree-like-structures, see Figure 6.5. The SPSM matching process is organized in two steps: *(i)* node matching and *(ii)* tree matching.

Node matching tackles the semantic heterogeneity problem by considering only labels at nodes and domain specific contextual information of the trees. SPSM uses the *S-Match* system [48]. Technically, two nodes *n1* and *n2* in trees *T1* and *T2* match if and only if: *c@n1 R c@n2* holds based on S-Match. *c@n1* and *c@n2* are the concepts, that represent entities of the local ontologies, at nodes *n1* and *n2* and $R \in \{=, \sqsubseteq, \sqsupseteq, \textit{not related}\}$. In particular, in semantic matching [44] as implemented in the *S-Match* system the key idea is that the relations (e.g., $=, \sqsubseteq$) between nodes are determined by *(i)* expressing the entities, that is the concepts, of the ontologies as logical formulas and *(ii)* reducing the matching problem to a logical validity problem.

Specifically, concepts are translated into logical formulas which explicitly express the concept descriptions as encoded in the ontology structure and in external resources, such as WordNet [91]. This allows for a translation of the matching problem into a logical validity problem, which can then be efficiently resolved using sound and complete state-of-the-art satisfiability solvers [47]. Notice that the result of this stage is the set of correspondences holding between the nodes of the trees. For example, that *requestMap* and *Version* in *T1* correspond to *requestMap* and *Edition* in *T2*, respectively.

Tree matching, in turn, exploits the results of the node matching and the

structure of the trees to find if these globally match each other. Technically, two trees *T1* and *T2* approximately match if and only if there is at least one node $n1_i$ in *T1* and node $n2_j$ in *T2* such that: *(i)* $n1_i$ matches $n2_j$, *(ii)* all ancestors of $n1_i$ are matched to the ancestors of $n2_j$, where $i = 1 \ldots N1; j = 1 \ldots N2; N1$ and $N2$ are the number of nodes in *T1* and *T2*, respectively.

6.5.3 SPSM implementation

The implementation of SPSM is based on *(i)* a formal theory of abstraction [45] and *(ii)* a tree edit-distance [145].

Abstraction operations

The work in [45] categorizes the various kinds of abstraction operations, including:

- **Predicate (Pd):** two or more predicates are merged, typically to the least general generalization in the predicate type hierarchy, e.g., $Height(X)+Dimension(X) \rightarrow Dimension(X)$. We call $Dimension(X)$ a predicate abstraction of $Height(X)$, namely $Dimension(X) \sqsupseteq_{Pd} Height(X)$. Conversely, we call $Height(X)$ a predicate refinement of $Dimension(X)$, namely $Height\ (X) \sqsubseteq_{Pd} Dimension(X)$.

- **Domain (D):** two or more terms are merged, typically by moving constants to the least general generalization in the domain type hierarchy, e.g., $Xmin\_BB + Xmin \rightarrow Xmin$. We call $Xmin$ a domain abstraction of $Xmin\_BB$, namely $Xmin \sqsupseteq_D Xmin\_BB$. Conversely, we call $Xmin\_BB$ a domain refinement of $Xmin$, namely $Xmin\_BB \sqsubseteq_D Xmin$.

- **Propositional (P):** one or more arguments are dropped, e.g., $Layers(L1) \rightarrow Layers$. We call $Layers$ a propositional abstraction of $Layers(L1)$,

namely $Layers \sqsubseteq_P Layers(L1)$. Conversely, $Layers(L1)$ is a propositional refinement of $Layers$, namely $Layers(L1) \sqsubseteq_P Layers$.

Let us consider the following example: $Height(X)$ and $Dimension$. In this case there is no abstraction/refinement operation that makes those first-order terms equivalent. However, consequent applications of propositional and domain abstraction operations make the two terms equivalent: $Height(X) \sqsubseteq_P Height \sqsubseteq_D Dimension$.

The abstraction/refinement operations discussed above preserve the desired properties: that functions are matched to functions and variables to variables. For example, predicate and domain abstraction/refinement operations do not convert a function into a variable. Thus, for instance, the correspondences between $Height$ (variable) and $Width$ (variable) in $T1$ and $Dimension$ (function) in $T2$, although returned by the node matching, should be further discarded, and therefore, are not shown in Figure 6.5.

Global similarity measurement

The key idea is to use abstractions/refinements as allowed tree edit-distance operations in order to estimate the similarity of two tree structures. Tree edit-distance is the minimum number of tree edit operations, namely node *insertion*, *deletion*, *replacement*, required to transform one tree to another. The goal is to: *(i)* minimize the editing cost, i.e., computation of the minimal cost composition of abstractions/refinements, *(ii)* allow only those tree edit operations that have their abstraction theoretic counterparts in order to reflect semantics of the first-order terms. A uniform proposal here is to assign a unit cost (see Table 6.1) to all operations that have their abstraction theoretic counterparts, while operations not allowed by definition of abstractions/refinements are assigned an infinite cost.

The following three relations between trees are considered: $T1 = T2$, $T1 \sqsubseteq T2$, and $T1 \sqsupseteq T2$. A global similarity score (*TreeSim*) between two

Table 6.1: The correspondence between abstraction operations, tree edit operations and costs.

Abstractions	**Operation**	**Preconditions**	$Cost_{T1=T2}$	$Cost_{T1 \sqsubset T2}$	$Cost_{T1 \sqsupset T2}$
$t_1 \sqsupseteq_{Pd} t_2$	$replace(a, b)$	$a \sqsupseteq b$; a and b correspond to predicates	1	∞	1
$t_1 \sqsupseteq_{D} t_2$	$replace(a, b)$	$a \sqsupseteq b$; a and b correspond to functions or constants	1	∞	1
$t_1 \sqsupseteq_{P} t_2$	$insert(a)$	a corresponds to predicates, functions or constants	1	∞	1
$t_1 \sqsubseteq_{Pd} t_2$	$replace(a, b)$	$a \sqsubseteq b$; a and b correspond to predicates	1	1	∞
$t_1 \sqsubseteq_{D} t_2$	$replace(a, b)$	$a \sqsubseteq b$; a and b correspond to functions or constants	1	1	∞
$t_1 \sqsubseteq_{P} t_2$	$delete(a)$	a corresponds to predicates, functions or constants	1	1	∞
$t_1 = t_2$	$a = b$	$a = b$; a and b correspond to predicates, functions or constants	0	0	0

trees $T1$ and $T2$ ranges in $[0 \ldots 1]$ and is computed as follows:

$$TreeSim(T1, T2) = 1 - \frac{min \sum_{i \in S} n_i \cdot Cost_i}{max(N1, N2)} \tag{6.1}$$

where S is the set of allowed tree edit operations, n_i is the number of i^{th} operation necessary to convert one tree into the other, and $Cost_i$ is the cost of the i^{th} operation. The minimal edit-distance is normalized by the size of the biggest tree. Finally, a normalized distance (denoting dissimilarity) is converted into a similarity score. When $Cost_i$ is infinite (see Table 6.1), $TreeSim$ is estimated as zero.

The highest value of $TreeSim$ among $T1 = T2$, $T1 \sqsubseteq T2$, and $T1 \sqsupseteq T2$ is returned as the final similarity score. For the example of Figure 6.5, 10 node-to-node correspondences, namely 6 equivalence and 4 abstraction/refinement relations, were identified by the node matching algorithm. The biggest tree is $T2$ with 12 nodes. Then, these are used to compute $TreeSim$ between $T1$ and $T2$ by exploiting the above mentioned formula. In our example $TreeSim$ is 0.54 for both $T1 = T2$ and $T1 \sqsubseteq T2$ (while it is 0 for $T1 \sqsupseteq T2$). The tree similarity value is used to select trees whose sim-

ilarity value is greater than a cut-off threshold. In our example $TreeSim$ is higher than the cut-off threshold of 0.5, and, therefore, the two trees globally match as expected and the correspondences connecting the nodes of the term trees can be further used for data translation purposes.

6.6 Summary

In this chapter we described OpenKnowledge, an operational system that uses models of interaction as the focus for knowledge exchange. First, we defined and contextualized the OpenKnowledge system by comparing it to the approaches that have also taken an interaction-oriented method such as P2P and multi-agent systems, Semantic Service-Oriented architectures and Grid-service models. Then, we described LCC, a choreography language employed by OpenKnowledge to specify protocols between peers.

Next, we presented the OpenKnowledge system model which is based on IMs, OpenKnowledge Components, and on the Discovery Team Formation Service, a module that coordinates distributed participants to IMs. Also, we illustrated the P2P oriented architecture of the OpenKnowledge system by describing the main modules of the system and their functionalities. Finally, we focused on the matcher module, which is used when searching and integrating IMs and services provided by peers.

In the following chapter we will present how we implemented, within the motivating scenario depicted in §5, the SDI use cases we described in §5.3.

Chapter 7

SDI services implementation

As mentioned in the previous chapter, OpenKnowledge peers interact with other peers through specific protocols. The language used to specify these protocols is a modification of LCC which we presented in §6.2 and which is normally used to define multi-agent interactions. In this chapter the goal is to show how we implemented the SDI services illustrated in §5.3. Thus, we first present the overall architecture of the peers involved in the use cases (§7.1). Each peer participates to specific IMs which we present in the following sections, namely the *gazetteer* service IM (§7.2), the *map* service IM (§7.3), and the *download* service IM (§7.4). Finally, we will give a description of each OKC which we implemented by using the Java language.

7.1 The OKCs architecture

Figure 7.1 illustrates the main *OK enabled* components that implement the use cases we depicted in §5.3.

Note that the SDI service provider is represented as a single peer and that client peers are represented as separated peers (one client for each

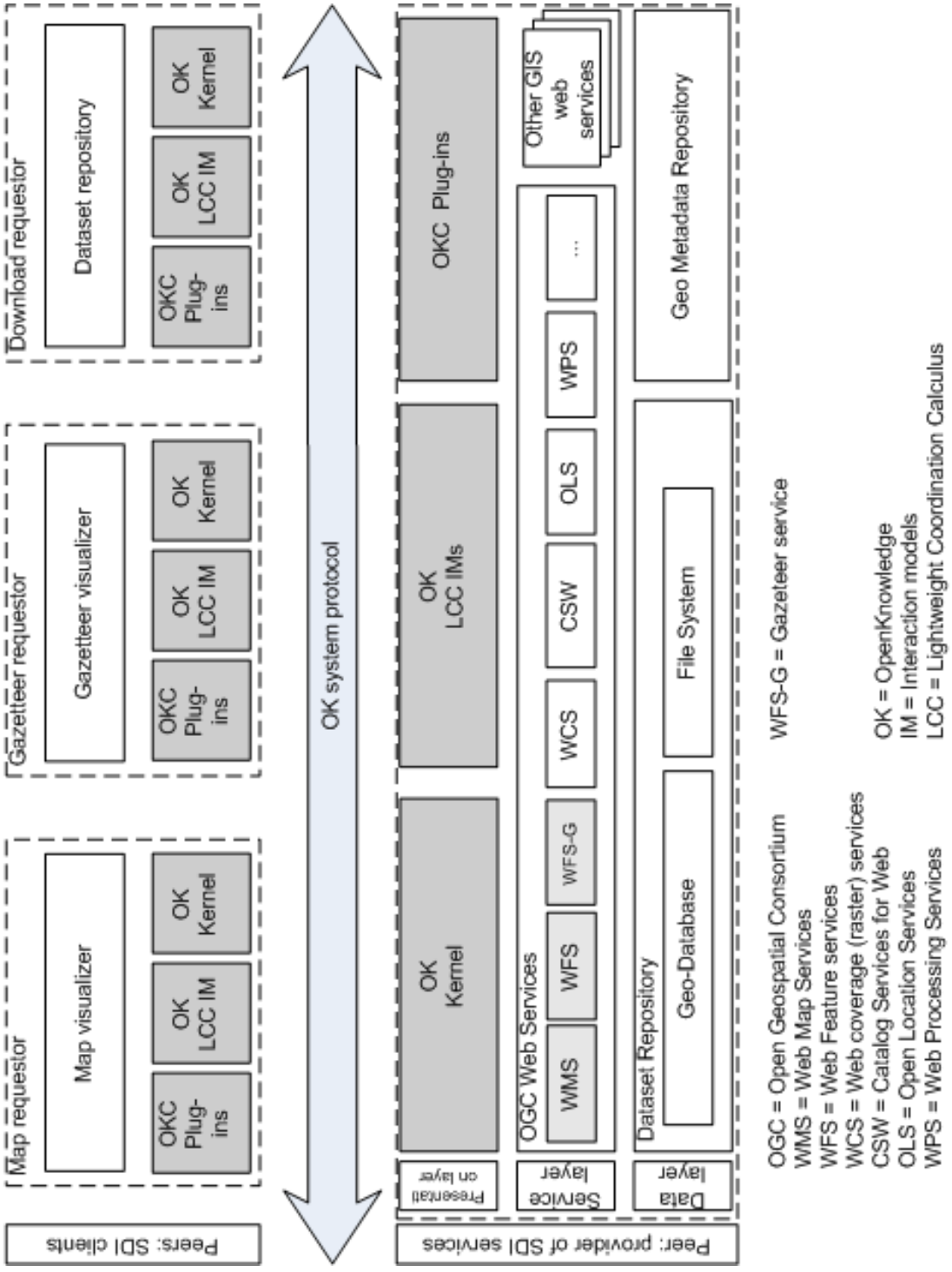


Figure 7.1: OK enabled SDI services

use case). This is a particular case of the more general case in which different peers (e.g., one for each provincial GIS agency) can provide GIS services, thus implementing a distributed SDI. Moreover, the three client peers can be grouped into an unique peer that requests all the services. For example, in our e-Response scenario, the emergency GUI (see §7.5) collects GIS services in the same graphic interface. Thus, either the emergency coordinator peer can ask for one of the GIS services we implemented, or three different fire fighters can ask for three different GIS services.

The main goal of the server peer (*provider of SDI services*) is to provide geographic data and geographic services. Its logical architecture can be subdivided into the following layers:

Data layer. The data layer represents the lowest layer of the service provider. This layer contains two kinds of objects: the geographic data (dataset) repository and the metadata repository. Geographic data can be stores using essentially two different methods: geo-database or file system. Usually a geo-database is implemented by an RDBMS plus an additional a spatial module (e.g., Oracle Spatial or PostGIS). A geo-database contains the geometrical representation (points, lines, polygons) of discrete features of the real world (e.g., buildings, roads, and residential areas). Continuous events (e.g., aerial and satellite photos and 3D terrain models) are usually stored by using the file system method. The geo metadata repository contains documentation about published geographic datasets and services.

GIS Service Layer. Different kinds of services can be provided by the server. OGC services are the standard way to implement GIS system functionalities, including the ones shown in Figure 7.1 (i.e., *Web Map Service, Web Feature Service, Gazetteer Service, Web Coverage Service, Catalog Service for Web, Open Location Service*, and *Web*

Processing Services). Other non-standard web services (*Other GIS Web services*) can be provided by the system (e.g., specific, non standard, local services).

Presentation layer. In our case, all the SDI functionalities can be accessed through the OpenKnowledge system interface. Each *OK enabled* peer can communicate with other peers when: (*i*) the OpenKnowledge kernel modules (coordinator, interpreter, matcher, trust and discovery) are locally installed (see §6.4), (*ii*) the IMs are downloaded, and (*iii*) when the OKC plug-in components, that satisfy the role constraints of IMs, are implemented.

All the (*OK enabled*) peers (OpenKnowledge client and server peers) interact by using LCC language IMs, and the underlying OpenKnowledge system architecture (*OK system protocol*).

7.2 The gazetteer service

The goal of the gazetteer service is to provide the geographical location of a name selected by a peer. In this section, we first describe the gazetteer use case (§7.2.1), then we provide its LCC formalization (§7.2.2), and finally we provide the description of the OpenKnowledge components that implement the gazetteer IM constraints (§7.2.3).

7.2.1 Description of the gazetteer use case

Figure 7.2 shows the sequence of the messages between a gazetteer service requestor and a gazetteer service provider. Usually, a gazetteer service requestor needs to find a geographical name (e.g., place name, river name,

mountain name) by using a name (string) indicated by a user (*getTopRequest(Top)*. In this case, the gazetteer service provider returns a list of the geographical names, and of their identifiers (*LT*) that corresponds to the term requested by the user (*getTopResponse(LT)*).

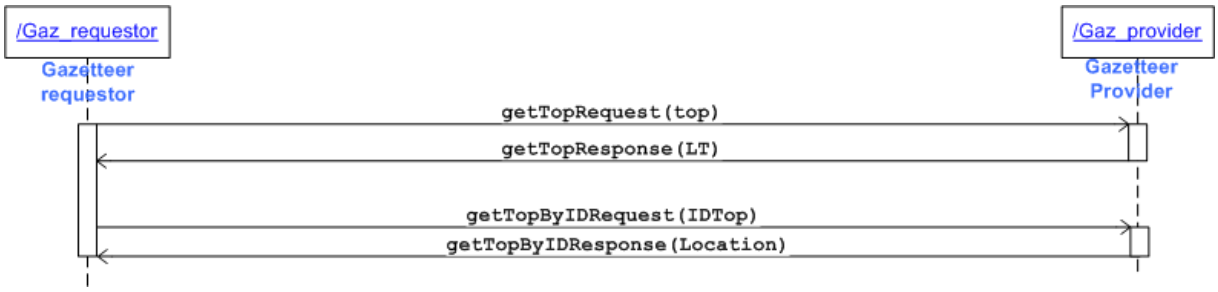


Figure 7.2: Gazetteer service sequence diagram.

The user selects a geographical name from the list returned by the gazetteer service provider. Then, the identifier (*IDTop*) of the correspondent geographical name is selected and sent to the gazetteer service provider (*getTopByIDRequest(IDTop)*). Finally, the service provider returns the geographical position (*Location*) of the requested geographical name (*getTopByIDResponse(Location)*).

7.2.2 The gazetteer Interaction Model

Figure 7.3 and Figure 7.4 show the LCC formalization for the IM depicted in Figure 7.2. The IM contains the interactions between a requestor of a gazetteer service (*gaz_requestor*, Figure 7.3) and a gazetteer service provider (*gaz_provider*, Figure 7.4).

Specifically, in Figure 7.3:

- The gazetteer requestor takes the role *gaz_requestor*. Here, it asks the user the geographical name (*locate(Top)*) and, if the user does not stop the interaction ($null \leftarrow endProt(Top)$), it sends the geographical name to the service provider (*getTopRequest(Top)*). Then,

$$
\begin{array}{l}
a(gaz\_requestor, R) :: \\
\quad (getTopRequest(Top) \Rightarrow a(gaz\_provider, Z) \leftarrow locate(Top)\ then \\
\left(\begin{array}{l}
getTopResponse(LT) \Leftarrow a(gaz\_provider, Z)\ then \\
\quad getTopByIDRequest(IDTop) \Rightarrow a(gaz\_provider, Z) \\
\qquad \leftarrow selectToponym(LT, IDTop)\ then \\
\quad \left(\begin{array}{l} getTopByIDResponse(Loc) \Leftarrow a(gaz\_provider, Z)\ then \\ null \leftarrow showToponymLocation(Loc)\ then \\ \qquad a(gaz\_requestor, R) \end{array} \right) or \\
\quad \left(\begin{array}{l} error(IDTop) \Leftarrow a(gaz\_provider, Z)\ then \\ \qquad a(gaz\_requestor, R) \end{array} \right) or \\
\quad notFound(Top) \Leftarrow a(gaz\_provider, Z)\ then \\
\qquad null \leftarrow notFoundTop(Top)\ then \\
\qquad\quad a(gaz\_requestor, R)
\end{array} \right) or \\
null \leftarrow endProt(Top)
\end{array}
$$

Figure 7.3: Gazetteer requestor IM

it waits for the list of the toponyms (LT) that corresponds to the user geographical name (*getTopResponse(LT)*).

- After receiving the list of the geographical names, if it is not null (*notFound(Top)*), the user selects a toponym (*IDTop*) from that list (*selectToponym(LT,IDTop)*) and then it requests the geographical location (*Loc*) of the toponym (*getTopByIDRequest(IDTop)*).

- After receiving the geographical location (*getTopByIDResponse(Loc)*), if there are no errors (*error(IDTop)*), the gazetteer requestor returns it to the user (*showToponymLocation(Loc)*).

In Figure 7.4 the gazetteer service provider acts as follows:

- The gazetteer provider peer takes the role *gaz_provider*. When it receives a request for a toponym position (*getTopRequest(Top)*), it searches for the geographical names (*searchFor(Top,LT)*) and if it does not find the toponym, it sends a *notFound(Top)* message to the re-

$$
\begin{array}{l}
a(gaz\_provider, Z) :: \\
\quad getTopRequest(Top) \Leftarrow a(gaz\_requestor, R)\ then \\
\left(\begin{array}{l}
\quad getTopResponse(LT) \Rightarrow a(gaz\_requestor, R) \leftarrow searchFor(Top, LT)\ then \\
\qquad\quad getTopByIDRequest(IDTop) \Leftarrow a(gaz\_requestor, R)\ then \\
\qquad\qquad getTopByIDResponse(Loc) \Rightarrow a(gaz\_requestor, R) \\
\qquad\qquad\quad \leftarrow transform(IDTop, Loc)\ then \\
\qquad\qquad\quad a(gaz\_provider, Z) \\
\qquad\qquad or \\
\qquad\qquad\quad error(IDTop) \Rightarrow a(gaz\_requestor, R)\ then \\
\qquad\qquad\qquad a(gaz\_provider, Z)
\end{array}\right) or \\
\quad notFound(Top) \Rightarrow a(gaz\_requestor, R)\ then \\
\qquad a(gaz\_provider, Z)
\end{array}
$$

Figure 7.4: Gazetteer requestor IM

questor. Otherwise, it sends a list of toponyms (LT) that correspond to the requested name (*getTopResponse(LT)*).

- After that, it waits for a location request (*getTopByIDrequest(IDTop)*) and, when it receives the request, if there are no errors (*error(IDTop)*), it transforms the toponym into a location position (*transform(IDTop, Loc)*) and passes it to the requestor (*getTopByIDResponse(Loc)*).

7.2.3 The gazetteer OKCs

For each role (*gaz_requestor* and *gaz_provider*) of the gazetteer IM an OKC has been developed. Figure 7.5 shows the Java class diagrams of the OKCs. For the *gaz_requestor* role the OKC implements all the constraints requested by the role (i.e., *locate(Top), selectToponym(LT, IDTop), showToponymLocation(Loc), notFoundTop(Top),* and *endProt(Top)*). All the OKCs were implemented as Java methods in a single Java class, namely *GazRequestorWSList*.

For the *gaz_provider* role another Java class has been implemented

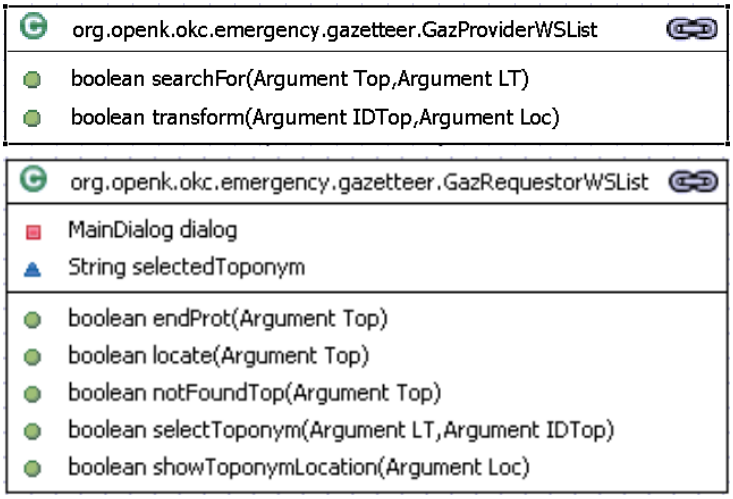


Figure 7.5: Java class diagrams of gazetteer service.

(*GazProviderWSList*). This class contains the Java methods (i.e., *searchFor(Top,LT)* and *transform(IDTop,Loc)*) that implement the constraints of the *gaz_provider* role.

In particular, the *gaz_provider* OKC invokes a gazetteer service built on the Deegree Java framework[1]. Geographical names of the Trento province were collected from the SIAT data repository. The gazetteer implemented by Deegree complies the WFS-G [104] OGC specification which, in turn, provides the following functionalities:

- **GetCapabilities.** When a client requests a capabilities document from the WFS-G provider, the provider returns a document that contains: *(i)* a description of all the operations that the WFS-G supports, *(ii)* a list of all feature types (layers that represent geographical names) that it can service, and *(iii)* a description of the structure of the underlying gazetteer data store. Roughly speaking, by using this functionality, the client asks a GIS service provider: *Do you implement a gazetteer?*

[1] `http://www.deegree.org`

- **DescribeFeatureType.** A client application (optionally) requests the set of gazetteer metadata objects (e.g., attributes) to identify the feature types that implement the gazetteer data model. The *DescribeFeatureType* operation allows gazetteer clients to retrieve schema descriptions which define how the gazetteer server will generate feature instances on output (in response to *GetFeature* requests, see below). Basically, by using this functionality, the client asks a gazetteer: *Which kinds of geographical names do you provide?*

- **GetFeature.** The *GetFeature* operation allows retrieval of features from a gazetteer service. A *GetFeature* request is processed by a gazetteer and, when the set of geographical names that corresponds to the request are found, an instance document, containing the result set, is returned to the client. The *GetFeature* operation supports the following behaviour:

  1. Get all entries in a gazetteer (empty filter).
  2. Get all entries in each separate gazetteer (a WFS-G can support *Multiple Gazetteers*).
  3. Get entry by name.
  4. Get entry by id.
  5. Get entries within a bounding box.
  6. Get entries within a polygon geometry.
  7. Each of the above queries for a specified feature.

  Finally, with this functionality, the client asks a gazetteer: *Could you find a particular geographical name?*

7.3 The map service

In this use case, the map requestor needs to visualize a map of a region with geo-referenced information selected by a user. Usually, the searched map is a composition of different geographic layers offered by a GIS service provider. In this section, we first describe the map service use case (§7.3.1), then, we provide its formalization with LCC (§7.3.2), and finally, we discuss the implementation of the OKCs that implement the map requestor and the map provider constraints (§7.3.3).

7.3.1 Description of the map request use case

Figure 7.6 shows the interaction for the Map request service.

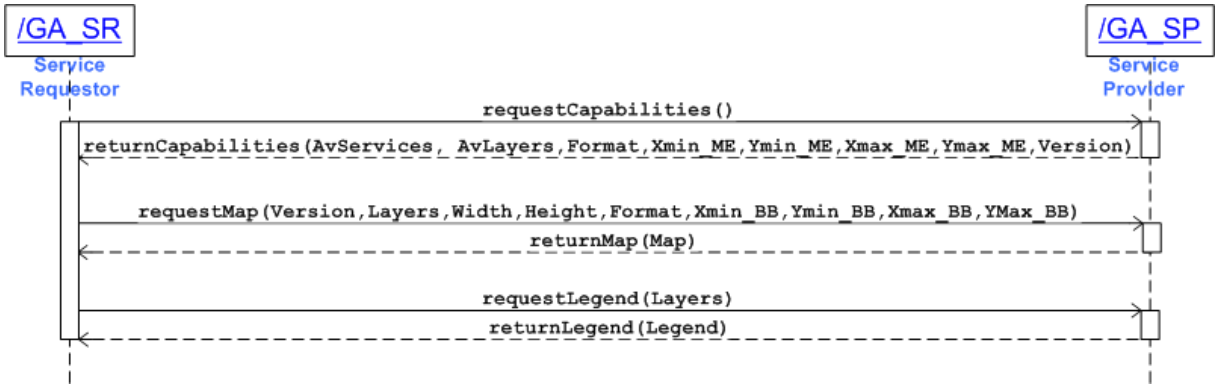


Figure 7.6: Map request service.

Interactions between a map service requestor and a map service provider are described as follows:

- The requestor (GIS Agency Service Requestor, *GA_SR*) asks the provider (GIS Agency Service Provider, *GA_SP*) for the characteristics of the provided services (*requestCapabilities()*).

- *GA_SP* returns its characteristics (*returnCapabilities*), in particular: the list of available services (*AvServices*), the list of geographic datasets managed by the server (*AvLayers*), the file format of the returned services (*Format*), the geographic bounds of the available services (*XMin_ME, YMin_ME, XMax_ME, YMax_ME*) and the software version (*Version*) of the adopted service.

- Then, *GA_SR* asks for the map service (*requestMap*) using the information received from the previous step. This message contains the software version of the adopted service (*Version*), the requested geographic layers (*Layers*, a subset of the available layers), the dimension of the map image (*Width, Heigth*), the format of the map image (*Format*) and the spatial coverage of the map (*XMin_BB, YMin_BB, XMax_BB, YMax_BB*).

- *GA_SP* provides the map (*return(Map)*) requested by the requestor.

- Finally, *GA_SR* asks for the graphic legend that describes the previous map (*requestLegend(Layers)*) and *GA_SP* returns the legend (*returnLegend(Legend)*) to *GA_SR*.

7.3.2 Formalization with LCC

Figure 7.7 and Figure 7.8 show the LCC code for the IM depicted in Figure 7.6.

The IM contains the interactions from the viewpoint of a GIS agency (map) service requestor (*ga_sr*, Figure 7.7) and of a GIS agency (map) service provider (*ga_sp*, Figure 7.8). Specifically:

- In Figure 7.7, the GIS agency service requestor (*ga_sr*) asks the service provider (*ga_sp*) its capabilities (*requestCapabilities()*).

$$
\begin{array}{l}
a(ga\_sr, R) :: \\
\quad requestCapabilities() \Rightarrow a(ga\_sp, P)\ then \\
\quad returnCapabilities(AvailableServices, AvailableLayers, Format, \\
\qquad XMin\_ME, YMin\_ME, XMax\_ME, YMax\_ME, Version) \\
\qquad \Leftarrow a(ga\_sp, P)\ then \\
\quad requestMap(Version, Layers, Width, Height, Format \\
\qquad XMin\_BB, YMin\_BB, XMax\_BB, YMax\_BB) \Rightarrow a(ga\_sp, P) \\
\qquad\quad \leftarrow selectLayers(AvailableLayers, Layers) \wedge needMap(Width, Height) \wedge \\
\qquad\quad selectBoundingBox(XMin\_ME, YMin\_ME, XMax\_ME, YMax\_ME, \\
\qquad\qquad XMin\_BB, YMin\_BB, XMax\_BB, YMax\_BB)\ then \\
\quad returnMap(Map) \Rightarrow a(ga\_sp, P)\ then \\
\quad requestLegend(Layers) \Rightarrow a(ga\_sp, P)\ then \\
\quad returnLegend(Legend) \Leftarrow a(ga\_sp, P)
\end{array}
$$

Figure 7.7: LCC fragment for the GIS agency service requestor role.

$$
\begin{array}{l}
a(ga\_sp, P) :: \\
\left(\begin{array}{l}
requestCapabilities() \Leftarrow a(ga\_sr, R)\ then \\
returnCapabilities(AvailableServices, AvailableLayers, Format, \\
\qquad XMin\_ME, YMin\_ME, XMax\_ME, YMax\_ME, Version) \\
\qquad \Rightarrow a(ga\_sr, R) \\
\qquad \leftarrow getCapabilities(Version, AvailableServices, AvailableLayers, \\
\qquad\quad Format, XMin\_ME, YMin\_ME, XMax\_ME, YMax\_ME)\ then \\
\qquad\quad a(ga\_sp, P)
\end{array}\right) \ or \\
\left(\begin{array}{l}
requestMap(Version, Layers, Width, Height, Format, \\
\qquad XMin\_BB, YMin\_BB, XMax\_BB, YMax\_BB) \Leftarrow a(ga\_sr, R)\ then \\
returnMap(Map) \Rightarrow a(ga\_sr, R) \\
\qquad \leftarrow \underline{getMap(Version, Layers, Width, Height, Format,} \\
\qquad\quad \underline{XMin\_BB, YMin\_BB, XMax\_BB, YMax\_BB, Map)}\ then \\
\qquad\quad a(ga\_sp, P)
\end{array}\right) \ or \\
\left(\begin{array}{l}
requestLegend(Layers) \Leftarrow a(ga\_sr, R)\ then \\
returnLegend(Legend) \Rightarrow a(ga\_sr, R) \\
\qquad \leftarrow getLegend(Layers, Legend)\ then \\
\qquad\quad a(ga\_sp, P)
\end{array}\right)
\end{array}
$$

Figure 7.8: LCC fragment for the GIS agency service provider role.

After that, the service requestor waits (*returnCapabilities(...)*)until the service provider returns the list of the available services (*AvailableServices*), the list of the available layers (*AvailableLayers*), the format of the returned file (*Format*), and the geographic coverage (map extent) of the available services (*XMin_ME, YMin_ME, Xmax_ME, YMax_ME*). Then the map requestor asks the service provider for a map (*requestMap(...)*). It selects some of the available geographic layers (*selectLayers(AvailableLayers, Layers)*), defines the map dimension (*needMap(Width, Height)*) and selects an area from the available geographic extension (*selectBoundingBox(XMin_ME, YMin_ME, XMax_ME, YMax_ME, XMin_BB, YMin-_BB, XMax_BB, YMax_BB)*). Finally, it requests the map legend of the selected layers (*requestLegend(Layers)*).

- In Figure 7.8 the GIS agency service provider (*ga_sp*) waits for one of the following requests: *requestCapabilities*, *requestMap* and *requestLegend*. After receiving one of them, it performs, respectively, the following actions:
  - It builds its capabilities (*getCapabilities(MapFile, Version, AvailableServices, AvailableLayers, Format, Xmin_ME, YMin_ME, Xmax_ME, YMax_ME)*) and passes them to the requestor (*returnCapabilities(...)*).
  - It builds a digital map (*getMap(Version, Layers, Width, Height, Format, XMin_BB, YMin_BB, XMax_BB, YMax_BB, Map)*) and sends it to the service requestor (*returnMap(Map)*).
  - It builds a legend of the requested layers (*getLegend(Layers, Legend)*) and returns it to the service requestor (*returnLegend(Legend)*).

  Note that in §6.5 we used the *getMap* constraint (underlined in Figure 7.8) as part of the motivating example of the employed matching

approach.

7.3.3 The map service OKCs

It has been developed an OKC for each role (*ga_sr* and *ga_sp*) of the map service IM. Figure 7.9 shows the Java class diagrams of the OKCs. For the *ga_sr* role the OKC implements all the constraints requested by the role (i.e., *selectLayers(...)* and *needMap(...)*). We implemented the constraints as Java methods in a single Java class, namely *WmsRequestor*.

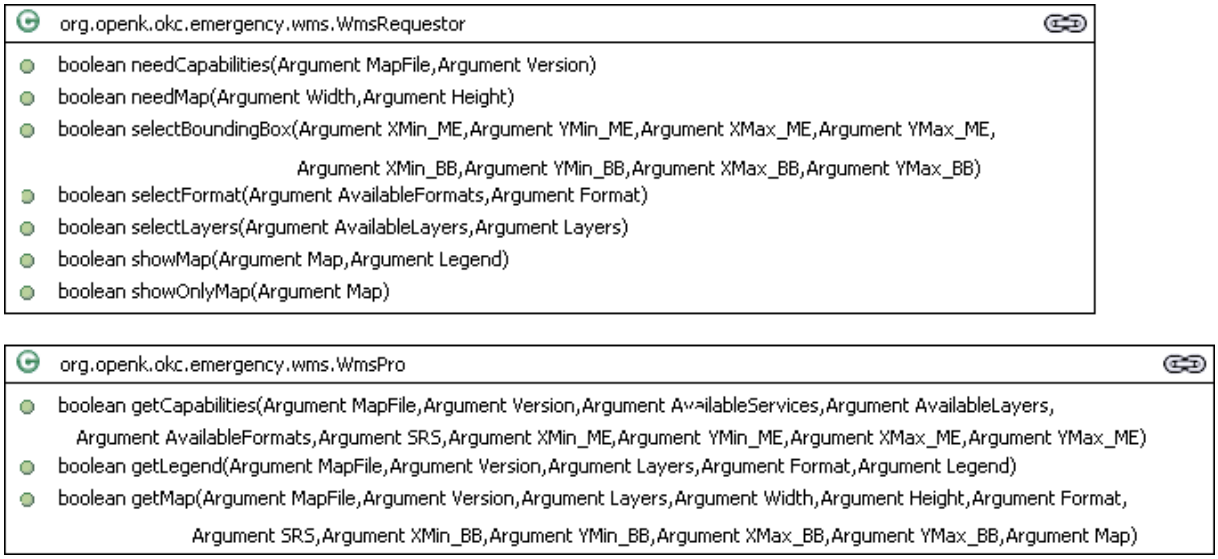


Figure 7.9: Java class diagrams of map service.

For the *ga_sp* role another Java class has been implemented (*WmsPro*). This class contains the Java methods that implement the constraints of the *ga_sp* role (i.e., *getCapabilities(...)*, *getMap(...)*, and *getLegend(Layers,Legend)*).

In particular, the class *WmsPro* invokes a WMS service built on the MapServer framework[2]. Geographical layers of the Trento province were collected from the SIAT data repository. The map service implemented by

[2] `http://mapserver.org/`

MapServer complies the WMS OGC specification which, in turn, provides the following functionalities:

- **GetCapabilities.** When a client requests a capabilities document from a WMS server, it returns a service-level metadata, which is a machine-readable (and human-readable) description of the WMS's information content and acceptable request parameters. In the particular case of a WMS, the response to a *GetCapabilities* request contains general information about the service itself and specific information about the available maps.

- **GetMap.** The server returns a map image whose geospatial and dimensional parameters are well defined. The *GetMap* operation is designed to produce a map, which is defined to be either a pictorial image or a set of graphical elements. Upon receiving a Map request, a Map Server shall either satisfy the request or throw an exception in the requested format.

- **GetFeatureInfo.** In this case the server returns information about particular features shown on a map. The *GetFeatureInfo* operation is designed to provide clients of a WMS with more information about features in the pictures of maps that were returned by previous *GetMap* requests. The canonical use case for *GetFeatureInfo* is that a user sees the response of a *GetMap* request and chooses a point on that map for which to obtain more information. The basic operation provides the ability for a client to specify which pixel is being asked about, which layer(s) should be investigated, and what format the information should be returned in.

7.4 The download service

This interaction models a protocol between a peer, that requests to download some geographical data, and a provider of geographical data. In this use case, the aim of the requestor is to obtain geographical data and then use them for some operations such as: perform data analysis, execute topological operations, pass them to other services, etc. In this section, we first describe the download service use case (§7.4.1), then we provide its formalization with LCC (§7.4.2), and finally, we discuss the implementation of the OKCs (§7.4.3).

7.4.1 Description of the download request use case

Figure 7.10 shows the sequence of the messages between a download service requestor and a download service provider.

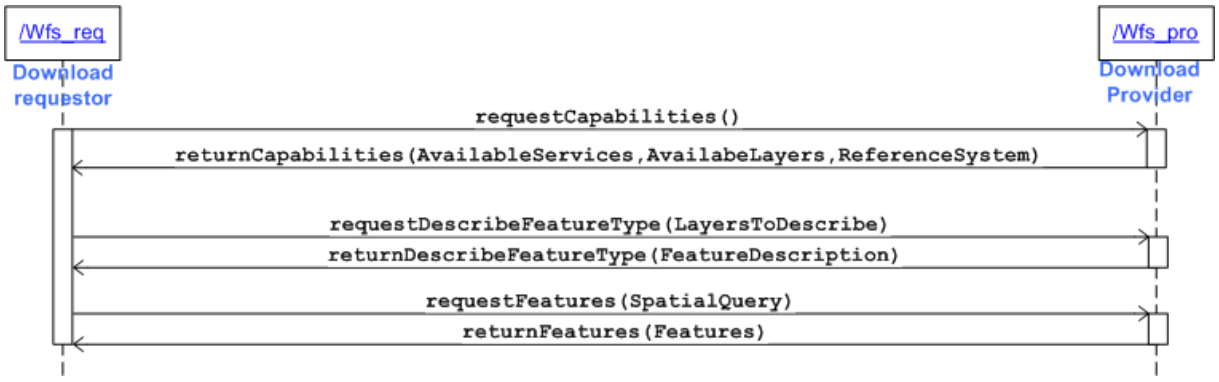


Figure 7.10: Sequence diagram of the download service.

Interactions between a download service requestor and a download service provider are briefly described as follows:

- The requestor (*Wfs_req*) asks the provider (*Wfs_pro*) for the characteristics of the provided services (*requestCapabilities()*).

- *Wfs_pro* returns the following characteristics: the list of available services (*AvailableServices*), the list of geographic datasets managed by the server (*AvailableLayers*), and the geographical reference system (*ReferenceSystem*) used by the provider.

- Then, *Wfs_req* can optionally asks for the description (i.e., the structure of the feature schemas) of the layers (*requestDescribeFeatureType(LayersToDescribe)*).

- If so, *Wfs_pro* provides the schema description of the features contained into each layer requested by the requestor (*returnDescribeFeatureType(FeatureDescription)*).

- Finally, if *Ws_req* asks the provider to download some selected geographical features (*requestFeatures(SpatialQuery)*), *Wfs_pro* returns the requested features (*returnFeatures((Features)*).

7.4.2 Formalization with LCC

Figure 7.11 and Figure 7.12 show the LCC code for the IM depicted in Figure 7.10. The IM contains the interactions between a requestor of a download service (*wfs_req*, Figure 7.11) and a download service provider (*wfs_pro*, Figure 7.12).

Specifically, in Figure 7.11:

- The download requestor takes the role *wfs_req*. Here, it asks the service provider (*wfs_pro*) its capabilities (*requestCapabilities()*). After that, the service requestor waits until the service provider returns the list of the available services (*AvailableServices*), the list of the available layers (*AvailableLayers*), and the coordinate reference system of the geographical features (*ReferenceSystem*).

$$
\begin{array}{l}
a(wfs\_req, R) :: \\
\quad requestCapabilities() \Rightarrow a(wfs\_pro, P)\ then \\
\quad\quad returnCapabilities(AvailableServices, AvailableLayers, ReferenceSystem) \\
\quad\quad\quad \Leftarrow a(wfs\_pro, P)\ then \\
\quad\quad\quad\quad \left(\begin{array}{l}
requestDescribeFeatureType(LayersToDescribe) \\
\quad \Rightarrow a(wfs\_pro, P) \\
\quad \leftarrow needDescribe(AvailableLayers, LayersToDescribe)\ then \\
\quad\quad returnDescribeFeatureType(FeatureDescription) \\
\quad\quad\quad \Leftarrow a(wfs\_pro, P)\ then \\
\quad\quad\quad a(wfs\_req\_fea(AvailableLayers), F)
\end{array} \right) \\
\quad\quad or \\
\quad\quad\quad a(wfs\_req\_fea(AvailableLayers), F)
\end{array}
$$

$$
\begin{array}{l}
a(wfs\_req\_fea(AvailableLayers), F) :: \\
\quad requestFeatures(SpatialQuery) \Rightarrow a(wfs\_pro, P) \\
\quad\quad \leftarrow buildSpatialQuery(AvailableLayers, FeatureDescription, SpatialQuery)\ then \\
\quad\quad returnFeatures(Features) \Leftarrow a(wfs\_pro, P)\ then \\
\quad\quad\quad null \leftarrow showFeatures(Features)
\end{array}
$$

Figure 7.11: Download service requestor role IM.

- Optionally, the download requestor (*needDescribe(AvailableLayers, LayersToDescribe)*) asks the service provider for the structure of the data schema (*requestDescribeFeatureType(featureDescription)*) and waits for the description from the service provider (*returnDescribeFeatureType-(FeatureDescription)*).

- Finally, the requestor assumes the role *a(wfs_req_fea(AvailableLayers),F)*, builds a spatial query (*buildSpatialQuery(AvailableLayers, Feature-Description, SpatialQuery)*, sends its request to the service provider (*requestFeatures(SpatialQuery)*), waits for the data from the provider (*returnFeatures(Features)*) and shows the final result to the user (*show Features(Features)*).

In Figure 7.12 the download service provider acts as follows:

$$
\begin{array}{l}
a(wfs\_pro, P) :: \\
\left(\begin{array}{l}
\quad requestCapabilities() \Leftarrow a(ga\_sr, R)\ then \\
\qquad returnCapabilities(AvailableServices, AvailableLayers, \\
\qquad\quad ReferenceSystem) \Rightarrow a(ga\_sr, R) \\
\qquad\quad \leftarrow getCapabilities(AvailableServices, \\
\qquad\qquad AvailableLayers, ReferenceSystem)\ then \\
\qquad\qquad a(wfs\_pro, P)
\end{array}\right) \ or \\
\left(\begin{array}{l}
\quad requestDescribeFeatureType(LayersToDescribe) \\
\qquad \Leftarrow a(ga\_sr, R)\ then \\
\qquad returnDescribeFeatureType(FeatureDescription) \Rightarrow a(ga\_sr, R) \\
\qquad\quad \leftarrow getDescribeFeatures(LayersToDescribe, \\
\qquad\qquad FeatureDescription)\ then \\
\qquad\qquad a(wfs\_pro, P)
\end{array}\right) \ or \\
\left(\begin{array}{l}
\quad requestFeatures(SpatialQuery) \Leftarrow a(ga\_sr(AvailableLayers), R)\ then \\
\qquad returnFeatures(Features) \Rightarrow a(ga\_sr, R) \\
\qquad\quad \leftarrow getFeatures(SpatialQuery, Features)\ then \\
\qquad\qquad a(wfs\_pro, P)
\end{array}\right)
\end{array}
$$

Figure 7.12: Download service provider role IM.

- It takes the role *wfs_pro*.
- When it receives a capabilities request (*requestCapabilities()*), it first builds its characteristics (*getCapabilities(...)*), i.e., the available services (*AvailableServices*), the available geographical layers (*AvailableLayers*), and the geographical coordinate reference system it uses to exchange the features (*ReferenceSystem*). After that, it returns (*returnCapabilities(...)*) the characteristics to the requestor.
- When it receives a request about the description of the geographical features it provides (*requestDescribeFeatureType(LayersToDescribe)*), it selects the features that correspond to the requested geographical layers (*getDescribeFeatures(LayersToDescribe, FeatureDescription)*) and sends the schema structure to the requestor (*returnDescribeFeatureType(FeatureDescription)*).

- When it receives a spatial query about the geographical feature it provides (*requestFeatures(SpatialQuery)*), it builds the query result (*getFeatures(SpatialQuery,Features)*) and returns the result to the requestor (*returnFeatures(Features)*).

7.4.3 The download service OKCs

It has been developed an OKC for each role (*wfs_req* and *wfs_pro*) of the download service IM. Figure 7.13 shows the Java class diagrams of the OKCs. For the *wfs_req* role the OKC implements all the constraints requested by the role (i.e., *needDescribe(AvailableLayers, LayersToDescribe)*, *buildSpatialQuery(AvailableLayers, FeatureDescription, SpatialQuery)*), and *showFeatures(Features)*. We implemented the OKCs as Java methods of a single Java class, namely *WfsRequestor*.

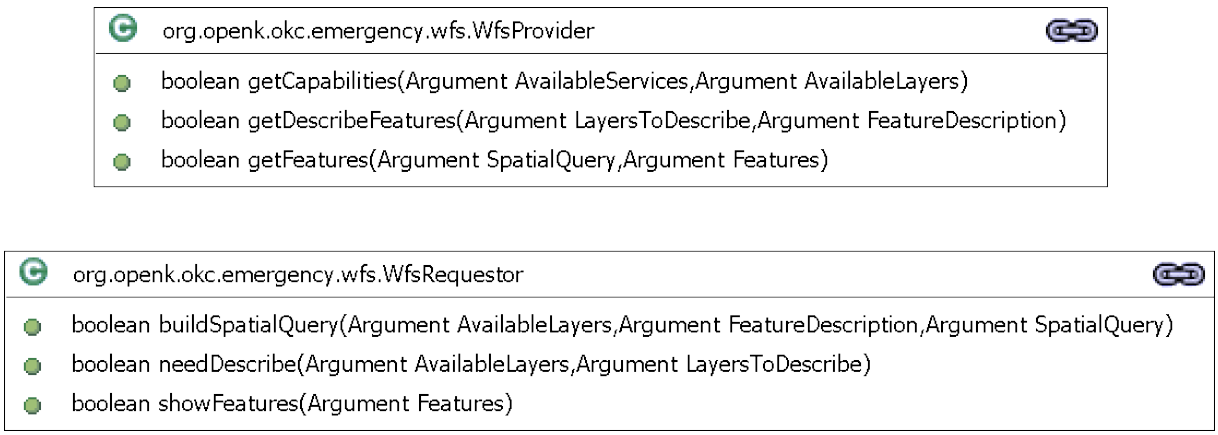


Figure 7.13: Java class diagrams of map service.

For the *wfs_pro* role another Java class has been implemented (*WfsProvider*). This class contains the Java methods that implement the constraints of the *wfs_pro* role (i.e., *getDescribeFeatures(LayersToDescribe, FeatureDescription)* and *getFeatures(SpatialQuery, Features)*).

In particular, the class *WfsProvider* invokes a WFS service built on

the MapServer framework[3]. Geographical layers of the Trento province were collected from the SIAT data repository. The download service implemented by MapServer complies the WFS OGC specification which, in turn, provides the following functionalities:

- **GetCapabilities.** A web feature service provider must be able to describe its capabilities. Specifically, it must indicate which feature types it can serve and what operations are supported on each feature type.
- **DescribeFeatureType.** The function of the *DescribeFeatureType* operation is to generate a schema description of feature types served by a WFS implementation. The schema descriptions define how a WFS implementation expects feature instances to be encoded on input (via *Insert* and *Update* requests) and how feature instances will be generated on output (in response, i.e., to a *GetFeature* request).
- **GetFeature.** The *GetFeature* operation allows retrieval of features from a WFS. A *GetFeature* request is processed by a WFS and usually a GML [107] instance document, containing the result set, is returned to the client.

7.5 The emergency GUI.

In order to complete the scenario, let us to describe how the internals discussed previously are actually used by final users. We developed an e-Response testbed in which the coordination of the web services between the network peers can be executed, visualized and analyzed [81]. In this application, the ongoing simulation of an emergency situation and the results acquired by the IMs proposed in the previous subsections, together

[3] `http://mapserver.org/`

with movements of the emergency peers, are visualized through a GUI as shown in Figure 7.14.

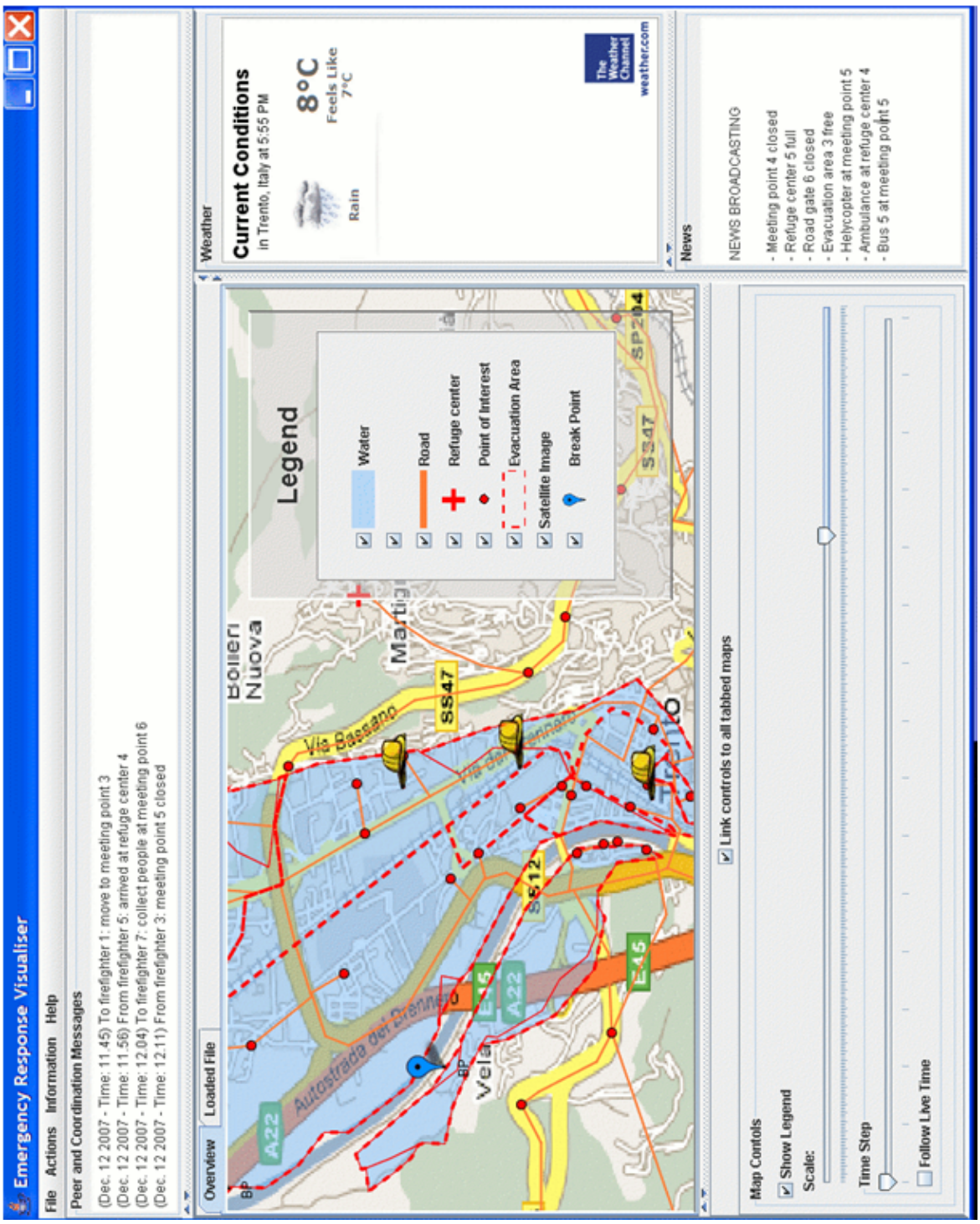


Figure 7.14: e-Response visualizer.

The GUI represents a control panel used by final users, e.g., by the emergency coordinators. Through the emergency GUI users can visualize the emergency area and the events of the emergency situation. SDI services provide geographical data to the users by using the interactions previously formalized with LCC. The GUI shows static geographic datasets (topographic map, probable flooding areas, escape roads, meeting points, refuge centers and sensor networks) as well as dynamic datasets (e.g., the position of the firefighters and of the citizens involved in the simulation). Moreover, through the GUI, users can perform actions (e.g., enact the emergency plan, recall digital services, search for other GIS datasets, locate a geographical name, send statements to the emergency actors, etc.) as well as ask information about the emergency situation (e.g., evacuated people, list of the emergency participants, blocked roads, situation of the meeting points and of the refuge centers, etc.).

7.6 Summary

In this chapter, we presented the implementation of three SDI use cases by the means of the OpenKnowledge system, namely the gazetteer, the map and the download use cases. For each use case, we presented:

- A description of the protocol between the requestor and the provider roles.
- A detailed explanation of the resulting IMs that implement the above mentioned protocols.
- An illustration of the OKCs and related Java classes which we used to implement the message constraints as requested by the IMs.

Moreover, we described the e-Response GUI used by the emergency peers to invoke the aforementioned services.

In the following chapter we will present a detailed analysis of the evaluation of the OpenKnowledge matcher module, that implements semantic ontology matching between service invocations (represented by the constraints of each IM) and service descriptions (represented, in our case, by the Java methods of each OKC).

Part IV

SPSM Evaluation

Chapter 8

The GIS web service evaluation dataset

In this chapter and in the following chapters of this part, we present an evaluation of the structure-preserving semantic matching (SPSM) approach, which we applied, as OpenKnowledge semantic matching module, within an emergency response scenario for geographic service coordination. Specifically, we evaluate the SPSM solution on real world GIS ESRI ArcWeb services[1] by conducting two kinds of experiments: (*i*) the first experiment (*evolution experiment*) includes matching of original web service signatures to synthetically altered ones, and (*ii*) in the second experiment (*classification experiment*) we compare a manual classification of our dataset to the unsupervised one produced by SPSM.

In particular, in this chapter, we describe the evaluation dataset, which is represented by the ESRI ArcWeb set of WSDL operations (§8.1). Then, we illustrate the evaluation setup, both for the evolution experiment (§8.2) and the classification experiment (§8.3).

[1] `http://www.esri.com/software/arcwebservices/`

8.1 Evaluation dataset

The SPSM solution allows to match web services that are described in the corresponding WSDL files and eventually in other formats, such as OWL-S and WSMO. However, until actual services with such semantic specifications are commonly published and available, we limit our evaluation to the names of the WSDL SOAP methods (operations) and of their parameters as carriers of meaningful information about the behavior and the semantics of the services. The SPSM approach thus assumes that the web services described in WSDL will be specified with some kinds of meaningful descriptions of: *(i)* what the operations are (e.g., *find_Address_By_Point*); *(ii)* what the inputs and outputs are: i.e., that arguments are labeled descriptively and not merely as *input1*, *var1*, and so on. Any additional mark-up that is used to provide semantics for web services outside of the WSDL files can also be amenable to the techniques, provided, as is usually the case, that descriptions of inputs and outputs can be captured in a tree structure.

In our experiments we base our test cases on ESRI ArcWeb WSDL operations and we compare labeled trees that correspond to the signature of the operations. ArcWeb is a rich and well documented set of web services which specifies an application programming interfaces (APIs) for integrating mapping functionality and GIS content into browser, desktop, mobile, and server applications. In the following list, we present a brief description of the ArcWeb SOAP API that has been used to build our evaluation dataset:

- **Address Finder Web Service:** performs geocode and reverse geocode.
- **Address Manager Web Service:** performs batch geocode and keeps the results.

- **Authentication Web Service:** creates authentication tokens to access other ArcWeb Services.
- **Content Finder Web Service:** searches metadata.
- **Data Manager Web Service:** uploads and stores data.
- **Map Image Web Service:** creates map images and thematic maps.
- **Place Finder Web Service:** finds place names.
- **Report Web Service:** creates demographic and site analysis reports.
- **Route Finder Web Service:** creates routes and driving directions.
- **Spatial Query Web Service:** finds nearest points, lines, and areas.
- **Utility Web Service:** calculates drive-time polygons and changes coordinate systems.
- **Wireless Location Web Service:** finds locations of wireless devices.

We conducted two different kinds of experiments. The first one has been inspired by the work on systematic benchmarks of the Ontology Alignment Evaluation Initiative (OAEI) [33]. In this experiment we matched original labeled trees to synthetically altered trees. Moreover, we compared the performance of the SPSM algorithm against the performance of a baseline solution, such as edit-distance[2]. In the second experiment we compared a manual classification of our GIS ArcWeb services dataset, the so-called reference alignment, to the unsupervised one produced by SPSM.

Finally, we evaluated efficiency and quality of the results of SPSM matching solution on these test cases. The evaluation was performed on a

[2]The edit-distance between two strings is given by the minimum number of operations needed to transform one string into the other, where an operation is an insertion, deletion, or substitution of a single character.

standard laptop Intel Centrino Core Duo CPU-2Ghz, 2GB RAM, with the Windows Vista (32bit, SP1) operating system, and with no applications running but a single matching system.

8.2 Evolution experiment setup

Ontology and web service engineering practices suggest that often the underlying trees to be matched are derived or inspired from one another. The result is equivalent to using different kinds of operations to change the syntax and the semantics of the original tree [43]. Therefore, it is reasonable to compare a tree with another one derived from the original tree. We evaluated SPSM following this experiment in which we performed syntactic and semantic alterations (*alteration operations*) to the nodes in trees, with a random probability (*alteration probability*) ranging in $[0.1 \ldots 0.9]$[3]. The evaluation dataset was composed of trees that are alterations of several original trees.

Initially, 80 original trees were extracted from the ESRI ArcWeb services collection. Some examples include:

- *find_Address_By_Point(point, address_Finder_Options, part)*,
- *get_Distance(location1, location2, num_Points, return_Geometry, token, units)*, and
- *convert_Map_Coords_To_Pixel_Coords(map_Area, map_Size, map_Coords, token)*.

Then, 20 altered trees were automatically generated for each original tree and for each alteration probability. Pairs, composed of the original tree and

[3]Probability values outside this range produce either too similar (< 0.1) or too different (> 0.9) trees to the original ones, and hence, are out of our interest.

one varied tree, were then used as input to SPSM. The alteration operations were applied to node names (node names being composed of labels) and correspond to the following four alteration categories (the underscored labels indicate modifications):

1. **Replace a node name with an unrelated node name**: a node name was replaced with an unrelated node name randomly selected from a generic dictionary. In our test we used the Brown corpus[4], a standard corpus of present-day American English. Some examples include:

   - Original tree:
     find_Address_By_Point(point, address_Finder_Options, part)
   - Modified tree:
     find_Address_By_Point(atom_firmer, discussion, part)

2. **Add or remove a label in a node name**: the label of a node name was either dropped or added. A label was dropped only if the node name contained more than one label. Label addition in node names was performed by using words extracted from the Brown corpus. Some examples include:

   - Original tree:
     find_Address_By_Point(point,address_Finder_Options,part)
   - Modified tree:
     find_By_Point(toast_point, address_Milledgeville_Finder_Options, part)

[4] http://icame.uib.no/brown/bcm.html

3. **Alter syntactically a label**: this test aimed at mimicking potential misspellings of the node labels. First, we decided randomly whether or not to modify a node name. Then, we randomly selected the set of labels to be modified and, for each word, we randomly decided how to modify it by using three types of alterations: character dropped, added, or changed. Some examples include:

   - Original tree:
     find_Address_By_Point(point,address_Finder_Options,part)
   - Modified tree:
     finm_Address_By_Poioat(einqt,ddress_Finder_Optxions,vparc)

4. **Replace a label in a node name with a related (i.e., synonyms, hyponyms, and hypernyms) one**: this test aimed at simulating the selection of an operation whose meaning was similar (equivalent, more general or less general) to the original one. In the implementation of these types of alterations we used a number of generic sources like WordNet 3.0 and Moby[5]. Some examples include:

   - Original tree:
     find_Address_By_Point(point,address_Finder_Options,part)
   - Modified tree:
     locate_Address_By_Point(place, address_Finder_Options, part)

We implemented evaluation tests to explore the robustness of the SPSM approach towards both typical syntactic alterations (i.e., replacements of node names, modification of node names and misspellings) and typical semantic alterations (i.e., usage of related synonyms, hyponyms, hypernyms) of node names.

[5] http://www.mobysaurus.com

8.3 Classification experiment setup

In this experiment, we aimed at investigating the capability of the SPSM algorithm in the unsupervised clustering of a set of meaningfully related web service operations. The evaluation setup corresponds to a manual classification (reference alignment) of a selected set (50) of ArcWeb service operations. These 50 operations are a subset of the operations considered in the evaluation experiment. The subset was obtained as described in step 2 of the construction procedure (see next). The construction of the reference alignment included the following steps:

1. Manual classification of the initial set of operations conforming to the WSDL file description of the operations used in the evaluation experiment.

2. Deletion of some general (valid for all the group) operations, e.g., *get_Info (data_Sources, token)*; which do not contribute to operation-specific information of the classification process.

3. Refinement of the classification by logically regrouping some operations, e.g., *find _Place(place_Name, place_Finder_Options, token)* was grouped together with the *address_finder* set of operations.

Table 8.1 summarizes, for each original ArcWeb WSDL file (rows), the number of operations of each group of the reference alignment (columns). We compare each operation with all the other operations in the dataset.

Table 8.2 summarizes, for both the experiments, the evaluation parameters. Specifically, we report number of operations, number of levels, maximum and average number of nodes and labels of the evaluation datasets.

Table 8.1: Reference alignment of ArcWeb services.

	GeoCoding and routing	**Map pixel conversion**	**Data manager**	**Spatial query**	**Map image**	**Coordinate graphic transformation**	**Map transformation**
Address finder	4	-	-	-	-	-	-
Address manager	-	4	-	-	-	-	-
Data manager	-	-	12	-	-	-	-
Map image	-	2	-	-	11	-	-
Place finder	1	-	-	-	-	-	-
Route finder	1	-	-	-	-	-	-
Spatial query	-	-	-	3	-	-	-
Utility	2	-	-	-	-	5	3
Wireless location	2	-	-	-	-	-	-

Table 8.2: Summative statistics for the test cases.

Test case number	**Number of WSDL operations**	**Maximum number of levels**	**Maximum number of nodes**	**Average number of nodes**	**Maximum number of labels**	**Average number of labels**
1	80	1	7	3.8	16	8
2	50	1	7	4.1	16	9

8.4 Summary

In order to evaluate the semantic matching approach we conducted two kinds of experiments: in the first experiment the goal was to match original web services signatures to synthetically altered ones, in the second experiment we compared a manual classification of our dataset to the unsupervised one produced by SPSM.

Specifically, in this chapter, we first illustrated the set of GIS ArcWeb WSDL operations which we adopted as evaluation dataset. Then, we presented the experiment setup for both the aforementioned experiments. In the first experiment we initially extracted 80 original signatures from the ESRI ArcWeb services collection. Then, we synthetically generated al-

tered signatures from the original ones by using four alteration operations, namely (*i*) *replace a node name with an unrelated node name*, (*ii*) *add or remove a label in a node name*, (*iii*) *alter syntactically a label*, and (*iv*) *replace a label in a node name with a related (e.g., synonyms, hyponyms, and hypernyms) one*. Moreover, we applied these syntactic and semantic alterations to the original signatures, with a random probability ranging in $[0.1 \ldots 0.9]$.

In the second experiment we first performed a manual classification of the set of operations conforming to the WSDL file description of the operations. Then, we deleted some general (valid for all the group) operations, and finally, we refined the classification by logically regrouping some operations.

In the following chapter we will present the method which we adopted in the evaluation of both the experiments.

Chapter 9

Evaluation method

In this chapter we first define standard quality measures we used in our experiments (§9.1). Then, we describe the method which we adopted in the evaluation of both the evolution experiment (§9.2) and the classification experiment (§9.3). Finally, we present the number of matching tasks we use to evaluate the SPSM solution (§9.4).

9.1 Evaluation measures

We used standard measures such as precision, recall and F-measure to evaluate quality of the SPSM matching results [34]. Specifically, for both the experiments, we based calculation of these measures on the comparison of the correspondences produced by a matching system (R) with the reference correspondences considered to be correct (C). We also define the sets of true positives (TP), false positives (FP) and false negatives (FN), as, respectively, the set of the correct correspondences which have been found, the set of the wrong correspondences which have been found and the set of the correct correspondences which have not been found. Thus:

$$R = TP \cup FP \tag{9.1}$$

$$C = TP \cup FN \tag{9.2}$$

Precision, recall and F-measure are defined as follows:

- *Precision:* varies in the $[0 \ldots 1]$ range; the higher the value, the smaller the set of false positives which have been computed. Precision is a measure of correctness and it is computed as follows:

$$Precision = \frac{\mid TP \mid}{\mid R \mid} \tag{9.3}$$

- *Recall:* varies in the $[0 \ldots 1]$ range; the higher the value, the smaller the set of true positives which have not been computed. Recall is a measure of completeness and it is computed as follows:

$$Recall = \frac{\mid TP \mid}{\mid C \mid} \tag{9.4}$$

- *F-measure:* varies in the $[0 \ldots 1]$ range; it is global measure of the matching quality, which increases if the matching quality increases. The version presented here was computed as the harmonic mean of precision and recall:

$$F - measure = \frac{2 * Recall * Precision}{Recall + Precision} \tag{9.5}$$

9.2 Evolution experiment: the evaluation method

Since the generated tree alterations were known, these provided the ground truth (or the expected similarity score, see next equation 9.6), and hence, the reference results were available by construction (see also [33]). This allowed for the computation of the matching quality measures. In particular,

we computed the standard matching quality measures, such as precision, recall, and F-measure for the similarity between trees.

We assigned to each node a label that described the type of the relation with the original one. Initially, we set the value of similarity to 1 and the value of the relation to *equivalent*. Then each alteration operation, applied sequentially to each node, reduced the similarity value and changed the relation value. We changed the rate of the reduction and the value of the relation according to the following empirical rules.

1. **Replace a label with an unrelated label**: when applied, we classified the two nodes as *not related* and we set the node score to 0.

2. **Add or remove a label in a node name**: when applied, we reduced the current node score by 0.5. If the parent node was still related, we considered the initial node either *more general* (when the label was added) or *less general* (when the label was removed) than the modified node. Some examples include:

   - *the initial node is more general than the modified one*:
     - Original node:
       find_Address_By_Point(part)
     - Modified node (label added):
       find_disturbed_Address_By_Point(part)

   - *the initial node is less general than the modified one*:
     - Original node:
       find_Address_By_Point(address_Finder_Options)
     - Modified node (label removed):
       find_Address_By_Point(address_Finder)

3. **Alter syntactically a label**: when applied, for each letter dropped, added or changed, we empirically decreased the similarity value by $(0.5/(total\ number\ of\ letters\ of\ the\ node\ label))$. We did not change the relation value between the original node and the modified one.

4. **Replace a label in a node name with a related one**: when applied, if the two nodes were related, we did not change the score if the new label was a synonym. If the new label was a hypernym or a hyponym of the original node, we changed the relation value to, respectively, *less general* and *more general* and therefore we applied to the similarity value an empirical reduction of 0.5.

When all the alteration operations were applied, the expected similarity score (*ExpScore*) between two trees $T1$ (the original one) and $T2$ (the modified one) that ranges between $[0 \dots 1]$ was computed as follows:

$$ExpScore(T1, T2) = \frac{\sum_{i \in N} Score_i}{N} \tag{9.6}$$

where $Score_i$ is the resulting similarity value assigned to each node of $T2$ and the expected similarity score is normalized by the size N of the trees.

The reference correspondences, used to compute true positive and false positive correspondences, were the altered trees whose expected similarity scores were higher than an empirically fixed threshold ($corrThresh$). This empirically fixed threshold separates the trees that a human user would, on average, consider as still similar to the original from those that are too different. Of course, this is the source of subjectivity, though we set it based on our previous experience with ontology matching evaluation in OAEI[1] campaigns.

We computed recall, precision and F-measure values as shown by the

[1] http://oaei.ontologymatching.org/2006

equations 9.3, 9.4 and 9.5. We calculated the correspondences produced by the SPSM solution (R) and the reference correspondences considered to be correct (C) as follows:

$$R = \{T2 \in Res \mid TreeSim(T1, T2) \geq cutoffThresh\} \tag{9.7}$$

$$C = \{T2 \in Res \mid ExpScore(T1, T2) \geq corrThresh\} \tag{9.8}$$

where $ExpScore$ was computed for each modified tree ($T2$), $TreeSim$ (see equation 6.1) was the similarity score returned by the SPSM solution, $cutoffThresh$ was the $TreeSim$ cut-off threshold and Res was, for each original tree $T1$, the set of the modified trees.

The set of true positive, false positive and false negative correspondences were computed as follows:

$$TP = \{T2 \mid T2 \in R \wedge T2 \in C\} \tag{9.9}$$

$$FP = \{T2 \mid T2 \in R \wedge T2 \notin C\} \tag{9.10}$$

$$FN = \{T2 \mid T2 \in C \wedge T2 \notin R\} \tag{9.11}$$

To exemplify the equations above, Table 9.1 shows the results for the alteration operation *Add or remove a label in a node name* with an evaluation probability of 0.7.

In addition, for a fixed probability, we compared SPSM recall, precision and F-measure values with the ones obtained from a baseline matcher, namely edit-distance, for our evaluation. Also, we evaluated recall and precision using combined results obtained by varying the *add or remove a label in a node name with a related one* (semantic) alteration operation combined with the *alter syntactically a label* (syntactic) alteration operation.

We repeated all experiments described above 10 times in order to obtain

Table 9.1: Example of quality measures results.

Cut-off threshold	$\|C\|$	$\|R\|$	$\|TP\|$	$\|FP\|$	$\|FN\|$	**Recall**	**Precision**	**F-measure**
0.1	593	1598	593	1005	0	1.000	0.371	0.541
0.2	593	1585	593	992	0	1.000	0.374	0.545
0.3	593	1568	593	975	0	1.000	0.378	0.549
0.4	593	1496	593	903	0	1.000	0.396	0.568
0.5	593	1391	593	798	0	1.000	0.426	0.598
0.6	593	758	588	170	5	0.992	0.776	0.871
0.7	593	642	513	129	80	0.865	0.799	0.831
0.8	593	397	315	82	278	0.531	0.794	0.636
0.9	593	143	112	31	481	0.189	0.783	0.304

statistically significant results and the presented results (§10) correspond to the average values. The maximum value of standard deviation was 0.013.

9.3 Classification experiment: the evaluation method

As described in the evaluation set-up, we first classified a selected set of ArcWeb operations, in order to obtain the truth classification set for our evaluation (reference alignment). This classification was mainly based on the WSDL description of the operations. We built $n \times n$ matrix (where n was the number of the selected WSDL operations) that contained the reference alignment, i.e, the manual classification of each pair of operations, which we considered to be correct.

Let $OP = \{T_1, T_2, \ldots, T_n\}$ be the set of the trees that corresponds to the selected operations. We defined the correspondences (C) considered to be correct as the subset of the cartesian product $OP^2 = OP \times OP$ that corresponded to our reference alignment ($RefAlign$):

$$C = \{(T_i, T_j) \in OP^2 | (T_i, T_j) \in RefAlign, 1 \leq i \leq n, 1 \leq j \leq n\} \quad (9.12)$$

In this test we compared the constructed manual classification of the selected web service operations with the one automatically obtained by the SPSM approach. Specifically:

- We compared each operation signature with all the other signatures.
- We computed a similarity measure between each signature and all the other signatures.
- We classified the pairs of operations by comparing their similarity score to a given cut-off threshold.

We calculated the correspondences produced by the SPSM solution (R) as follows:

$$R = \{(T_i, T_j) \in OP^2 | TreeSim(T_i, T_j) \geq cutoffThresh, 1 \leq i \leq n, 1 \leq j \leq n\} \quad (9.13)$$

where $TreeSim$ (see equation 6.1) was the similarity score returned by the SPSM solution and $cutoffThresh$ was the $TreeSim$ cut-off threshold.

We used the SPSM algorithm to independently classify same operations in an automatic way. For each pair of operations, the SPSM algorithm returned a similarity measure ($TreeSim$) that was compared with a cut-off threshold ($cutoffThresh$) in the range $[0.1 \ldots 0.9]$. If the similarity measure was higher than the cut-off threshold then the pair was said to be similar. Finally, we compared the reference alignment with the automatic classification performed by SPSM.

We computed recall, precision and F-measure comparing the set of the relevant (manual) classifications and the set of the retrieved (automatic) correspondences as shown by the equations 9.3, 9.4 and 9.5. The set of true positives (TP) contained the pairs of operations which were manually classified in the same group and which similarity calculated by SPSM ($TreeSim$) was greater than the cut-off threshold (see Equation 9.14).

$$TP = \{(T_i, T_j) | (T_i, T_j) \in R \wedge (T_i, T_j) \in C, 1 \leq i \leq n, 1 \leq j \leq n\} \quad (9.14)$$

The set of false positives (FP) contained the pairs of operations that were not manually classified into the same group and whose ($TreeSim$) similarity score was greater than the cut-off threshold (see Equation 9.15).

$$FP = \{(T_i, T_j) | (T_i, T_j) \in R \wedge (T_i, T_j) \notin C, 1 \leq i \leq n, 1 \leq j \leq n\} \quad (9.15)$$

The set of false negatives (FN) contained the pairs of the operations that were manually classified into the same group but which ($TreeSim$) similarity score was lower than the cut-off threshold (see Equation 9.16).

$$FN = \{(T_i, T_j) | (T_i, T_j) \in C \wedge (T_i, T_j) \notin R, 1 \leq i \leq n, 1 \leq j \leq n\} \quad (9.16)$$

For example, we manually classified the following pair of operations into the same group:

find_Address_By_Point(point, address_Finder_Options)

and

find_Location_By_Phone_Number(phone_Number, address_Finder_Options).

In this case, SPSM returned a $TreeSim$ similarity score of 0.67. Then, if we set the cut-off threshold at 0.6 the correspondence returned by SPSM is a true positive, if we set the cut-off threshold at 0.7 the correspondence returned by SPSM is a false negative.

9.4 Number of matching tasks

A significant number of matching tasks has to be performed in order to evaluate the SPSM approach.

Specifically, for the evaluation experiment the number of the matching tasks is calculated as follows:

$$MatchTasks_1 = Op * Changes * Prob * AltOp * Rep \tag{9.17}$$

where Op is the number of initial operation signatures (80), $Changes$ is the number of the variations for each signature (20), $Prob$ is the number of probabilities which we applied to each alteration operation (i.e., 9, from 0.1 to 0.9, with step of 0.1), $AltOp$ is the number of alteration operations (4), and Rep is the number of repetitions of the experiment (10). We repeated this experiment 10 times given the sporadic nature of alterations; the resulting number of matching tasks here is 576.000.

In turn, the number of matching tasks we made when we compared the SPSM approach to the edit-distance matching algorithm, within the evolution experiment, is as follows:

$$MatchTasks_2 = Op * Prob^{AltOp} * Changes \tag{9.18}$$

where Op and is the number of initial operation signatures (80), $Prob$ is the number of probabilities which we applied to each alteration operation (9), $AltOp$ is the number of alteration operations which we combined in order to obtain both syntactic and semantic alterations (2), and $Changes$ is the number of the variations for each signature (20). Thus, in this case, the resulting number of the matching task is 129.600.

Finally, for the classification experiment we compared each operation signature with all the other signatures (50). Thus the resulting number of matching tasks $MatchTasks_3$ is 2.500. Therefore, the overall number of

matching tasks performed in all the experiments is as follows:

$$\sum_{i=1}^{3} MatchTasks_i = 708.100 \tag{9.19}$$

9.5 Summary

In this chapter we presented the evaluation method which we adopted in the experiments we made to evaluate SPSM. Specifically, for both the experiments, we used standard measures such as precision, recall and F-measure to evaluate quality of the SPSM matching results.

For the first experiment, we showed how we obtained the reference correspondences. First, we computed the *expected similarity score*, a resulting similarity score based on the modifications we applied to the original signatures in order to produce the synthetically alterated ones. Then, we used the expected similarity score and an empirically fixed threshold to calculate the reference correspondences.

For the second experiment, we first selected a subset of the evaluation dataset which we built in the first experiment. Then, we manually classified the set of WSDL operations by meaningfully grouping these operations into a number of collections (*reference alignment*). Next, we computed the similarity score between each pair of operations by using the SPSM approach. Finally, we compared the reference alignment to the automatic one performed by the SPSM approach.

Moreover, we calculated the total number of matching tasks we performed to evaluate the SPSM approach.

In the following chapter we will illustrate quality evaluation results and performance results for both the experiments.

Chapter 10

Evaluation results

In this section we first present the quality evaluation results for the SPSM evaluation experiments, namely, the evolution experiment and the classification experiment. In the first experiment we represented quality measures as function of applied cut-off threshold and alteration probability for each alteration operation (§10.1). Moreover, we compare the SPSM approach to a baseline matcher.

In the second experiment we also performed quality evaluation measures and we obtained best overall quality value (F-measure) around 55% for the given GIS operation set (§10.2). Next, we present the performance evaluation results (§10.3), and finally, we summarize evaluation results (§10.4).

10.1 Evolution experiment: the results

For each alteration operation, quality measures are functions of the $TreeSim$ cut-off threshold values and of the alteration probability. In all 3D graphs, we represent the variation of the probability of the alteration operation on Y axis, the used $TreeSim$ cut-off threshold on X axis and the resulting

measures of recall, precision and F-measure on Z axis. Moreover, in all reported graphs, we used an empirically fixed threshold $corrThresh = 0.6$ (see §9.2).

1. **Replace a node name with an unrelated node name:** this alteration operation replaced an entire node name with an unrelated one, randomly selected from a thesaurus. Graphs in Figures 10.1, 10.2 and 10.3 show the relationship between the variation of the probability of the alteration operation, the variation of the used $TreeSim$ cut-off threshold and the resulting measures of recall, precision and F-measure.

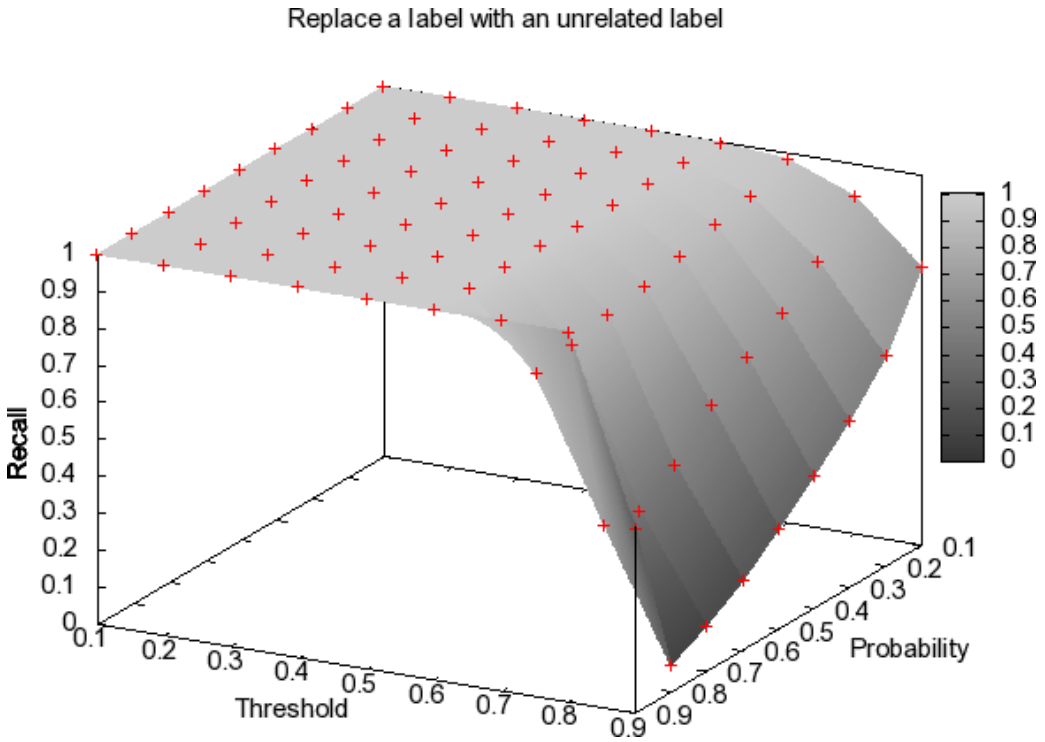


Figure 10.1: Recall of *Replace a node name with an unrelated one* alteration operation.

Figures 10.1, 10.2 and 10.3 indicate that for all alterations' probability the value of recall is very high up to a $TreeSim$ cut-off threshold (around 0.6), after which it drops rapidly. Thus, we can say

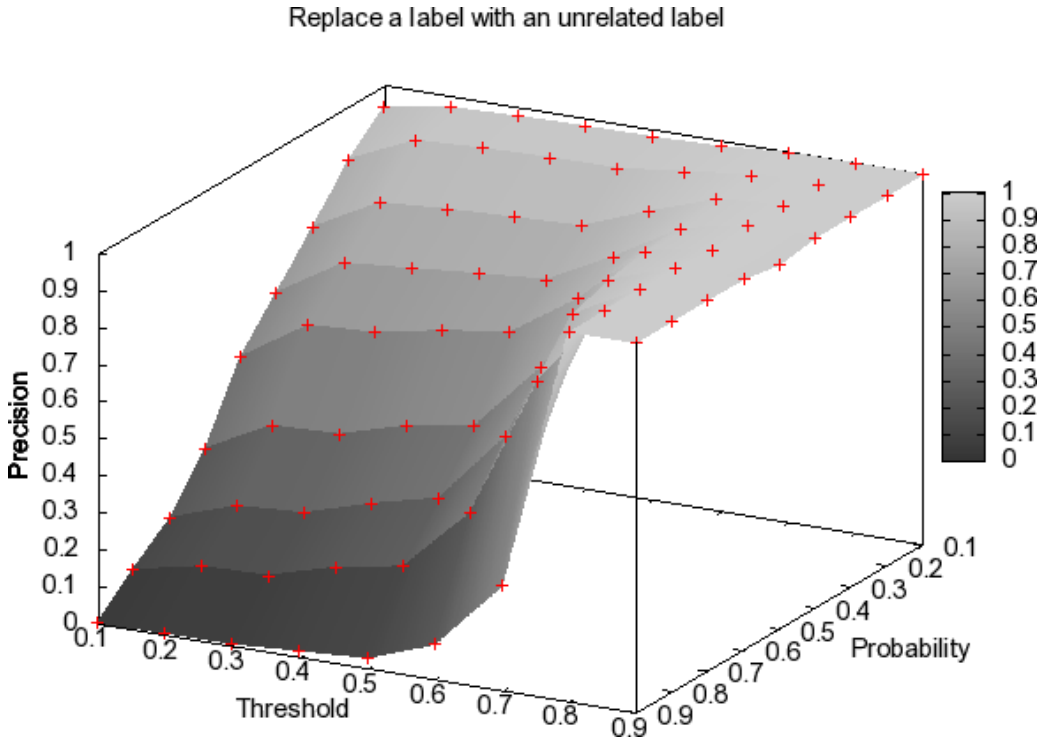


Figure 10.2: Precision of *Replace a node name with an unrelated one* alteration operation.

that, in our experiments, the SPSM approach retrieves all the expected (relevant) correspondences until the empirically fixed threshold ($corrThresh = 0.6$), that mimics the user's tolerance to errors, is reached.

The behavior of the precision is complementary: precision improves rapidly as the $TreeSim$ cut-off threshold exceeds the empirically fixed threshold. On the other hand, precision decreases steadily as a function of the alterations' probability while the $TreeSim$ cut-off threshold is below the empirically fixed threshold. We observed that this behavior, when we increased the probability of the alteration operation, depended on the decreasing number of true positives, while the number of false positives remained stable.

Figure 10.3 summarizes the overall quality performance for the SPSM algorithm in terms of F-measure: the best global measures of match-

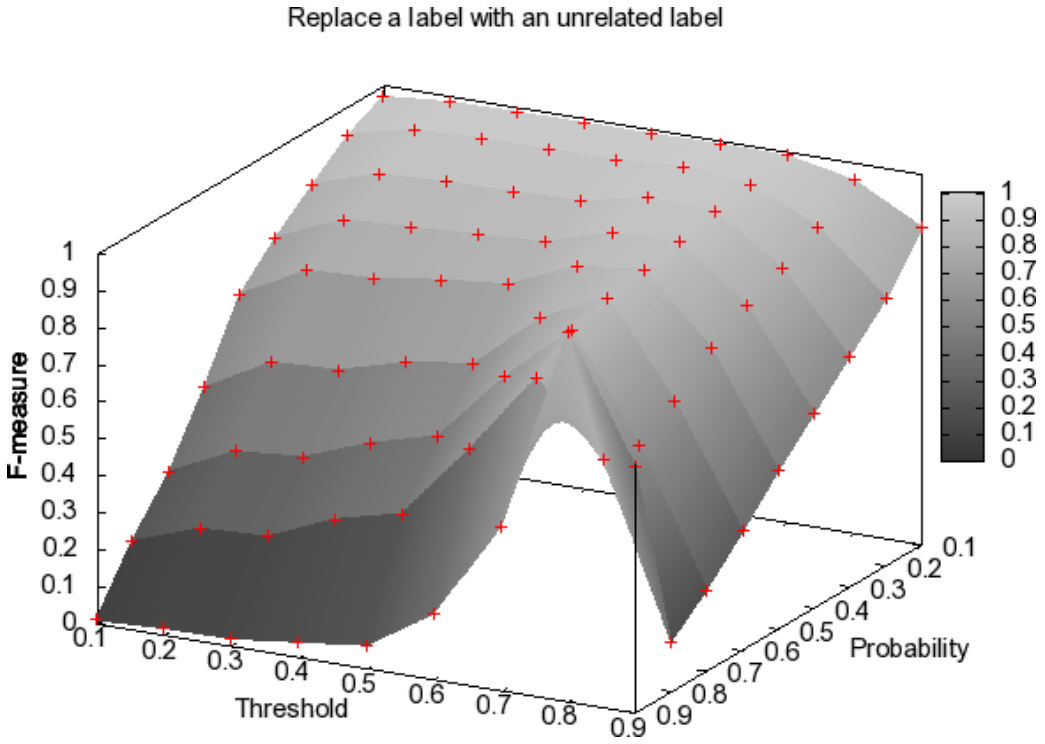


Figure 10.3: F-measure of *Replace a node name with an unrelated one* alteration operation.

ing quality are obtained around a cut-off threshold of 0.6, i.e., around the empirically fixed threshold ($corrThresh$) used to calculate the set of true positives, false positive and false negative correspondences (see equations 9.9, 9.10 and 9.11). Analyzing the data, we observe that this is, in fact, the threshold where we can find a good balance between the number of the true positive correspondences and the number of the false positive correspondences. Even when the probability of the alteration is very high the balance between correctness and completeness is good. For instance, at the optimal $TreeSim$ cut-off threshold (0.6), for an important alteration probability of 80%, F-measure is higher than 74%. These data prove the robustness of the SPSM approach up to significant syntactic modifications in the node names.

2. **Add or remove a label in a node name:** this alteration operation

added or removed a label in a node name. Figures 10.4, 10.5 and 10.6 show the relationship between the variation of the probability of the alteration operation, the applied *TreeSim* cut-off threshold and the resulting measures for recall, precision and F-measure.

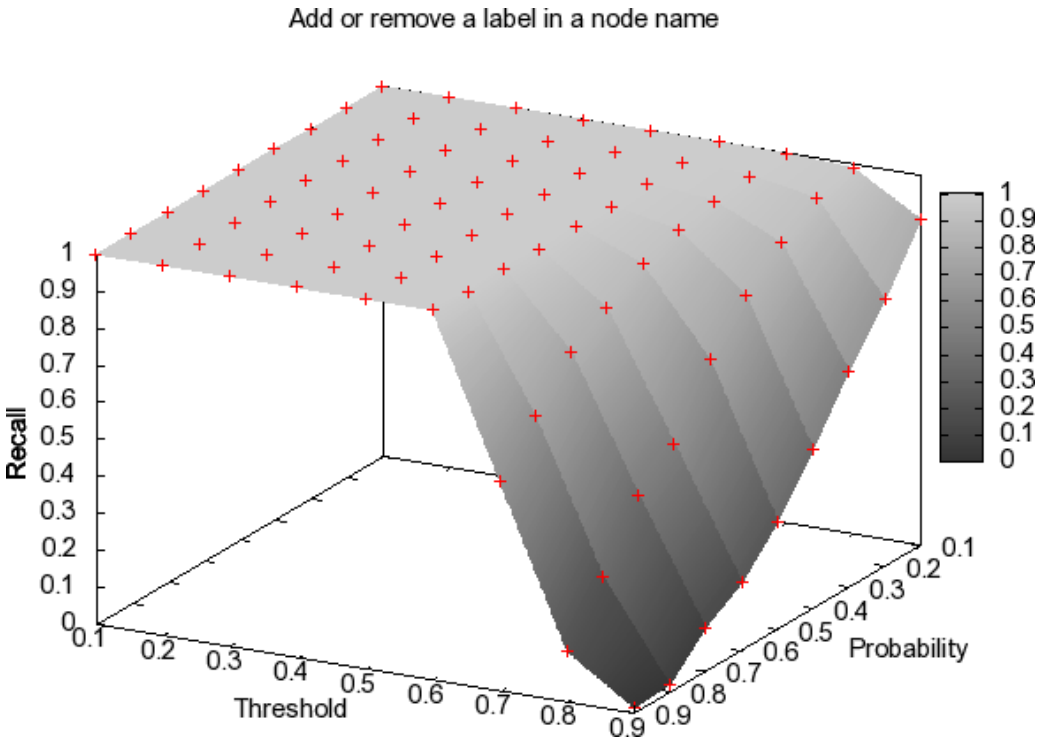


Figure 10.4: Recall of *Add or remove a label in a node name* alteration operation.

The behavior is similar to the one of the previous test. For instance, at the optimal *TreeSim* cut-off threshold (0.6), for an alteration probability of 80%, F-measure is higher than 75%. Thus, the previous arguments hold also here and we can conclude equally in this case that the SPSM approach is robust up to significant alteration (probability=80%) of node names.

3. **Alter syntactically a label in a node name:** this alteration operation altered syntactically a label in a node name, by modifying

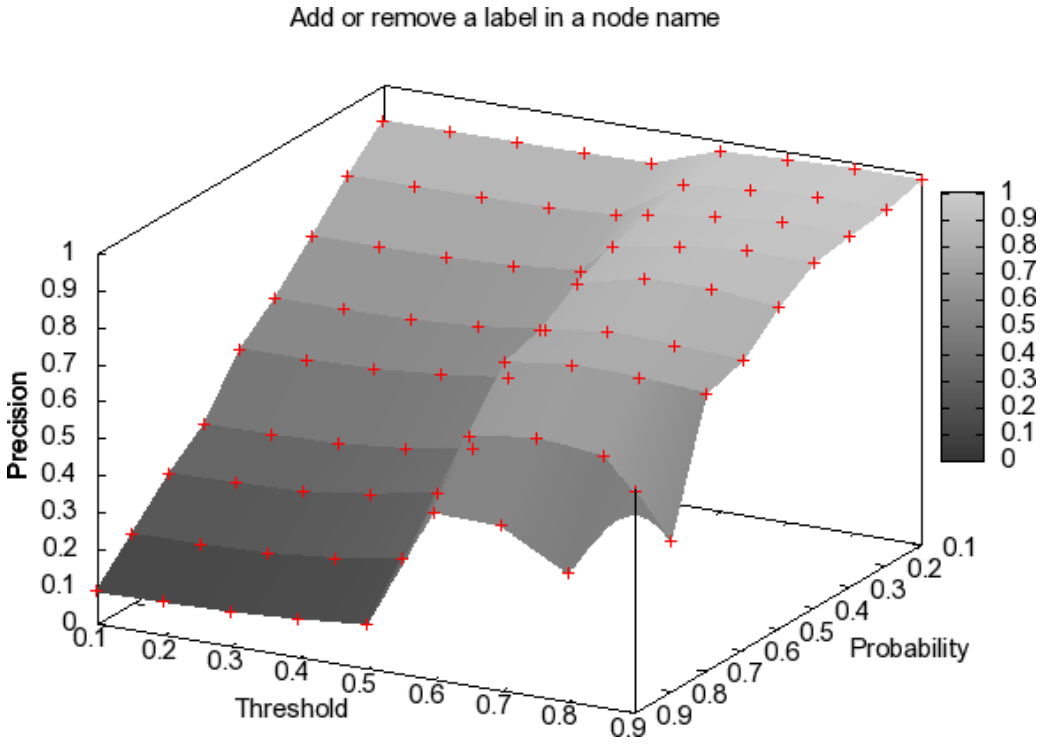


Figure 10.5: Precision of *Add or remove a label in a node name* alteration operation.

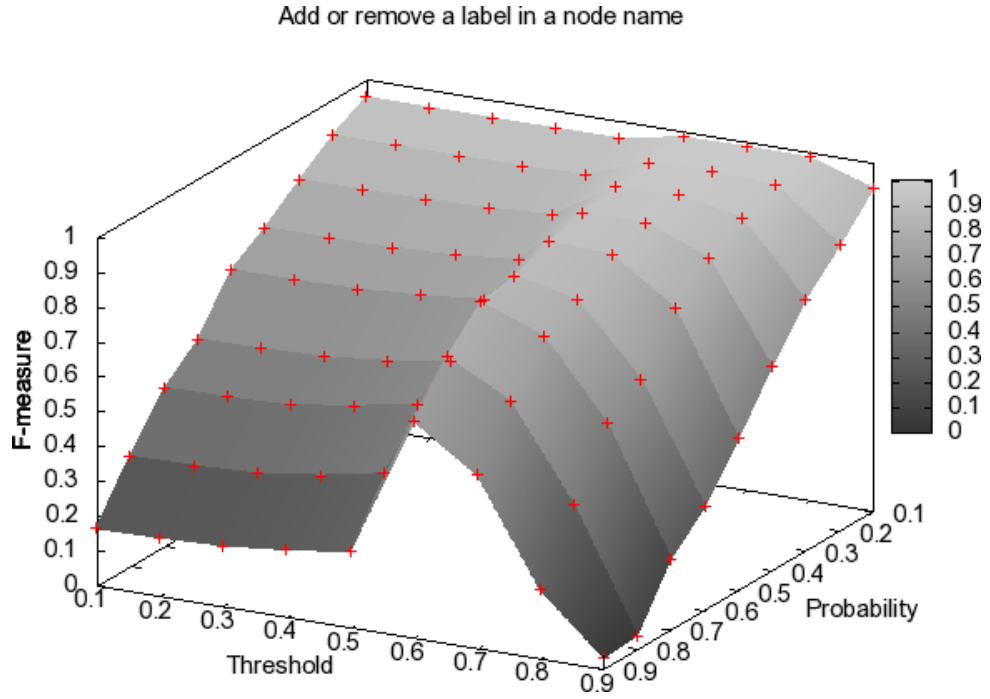


Figure 10.6: F-measure of *Add or remove a label in a node name* alteration operation.

(drop, add, delete) its characters. Figures 10.7, 10.8, and 10.9 show the evaluation results.

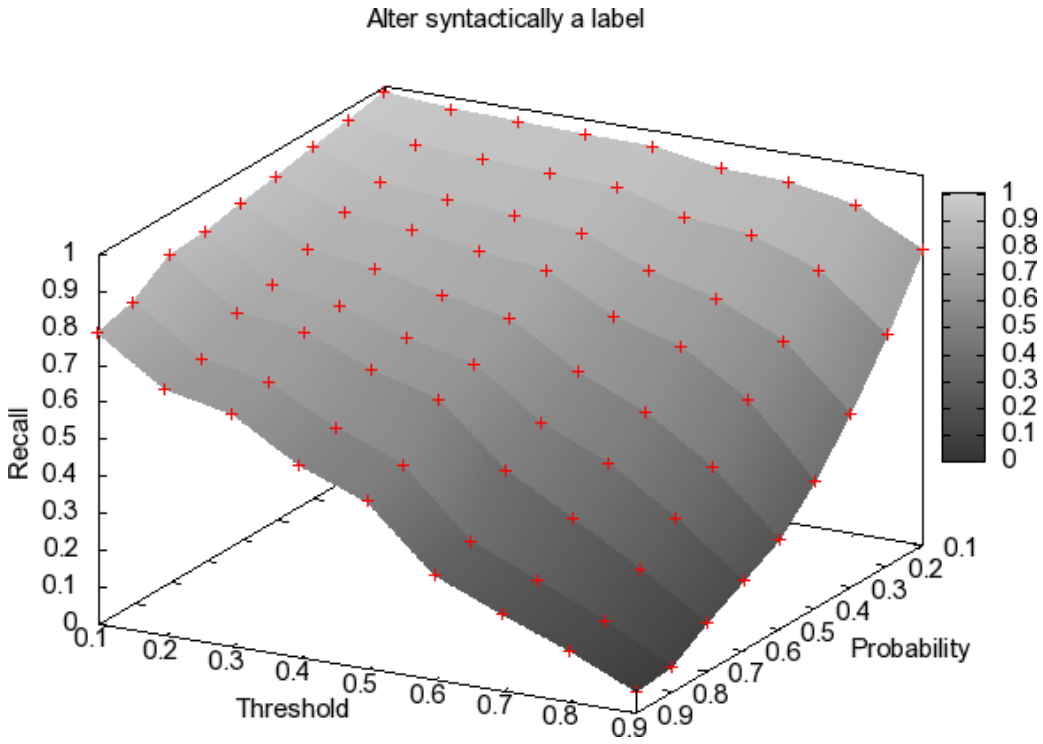


Figure 10.7: Recall of *Alter syntactically a label* alteration operation.

This test evaluated the robustness of the SPSM approach simulating errors and alterations that a programmer could make while writing the service operation signatures. In this test, recall decreases steadily as a function of increasing both probability of the alteration and *TreeSim* cut-off threshold. Precision is always high, in the range $[0.87 \ldots 1.0]$. This is due to a high number of true positive correspondences and to a simultaneously low number of false positive correspondences.

Therefore, F-measure graph (Figure 10.9) essentially reproduces the recall graph (Figures 10.7). F-measures values of ~ 0.7 were obtained for alterations' probability up to 70% and *TreeSim* cut-off thresholds up to 0.6.

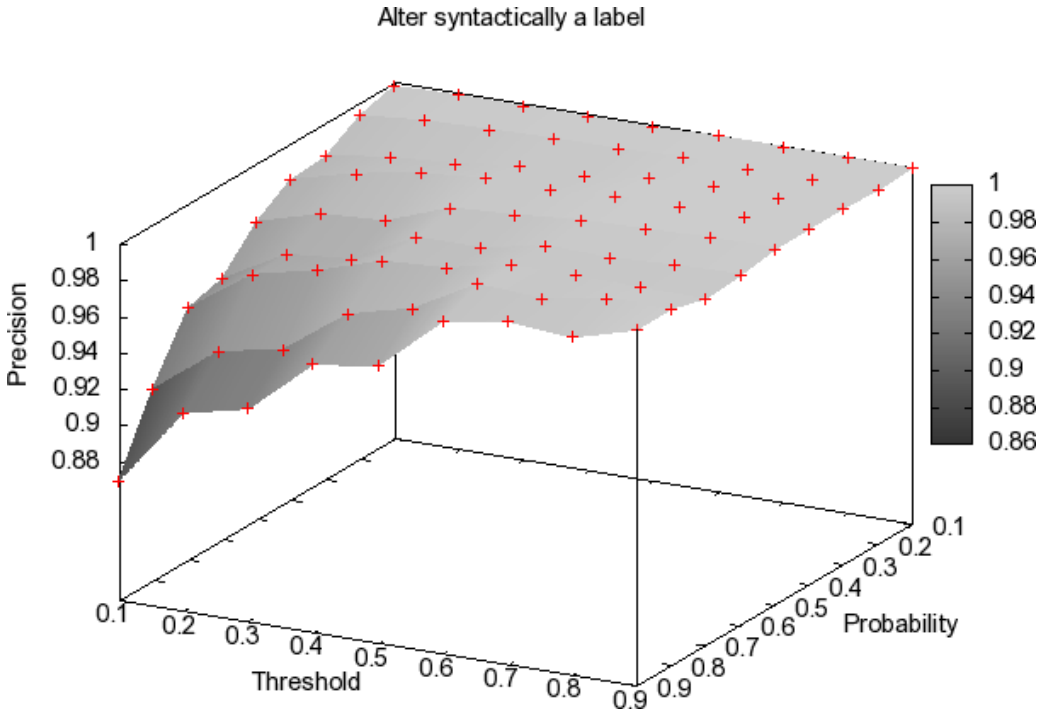


Figure 10.8: Precision of *Alter syntactically a label* alteration operation.

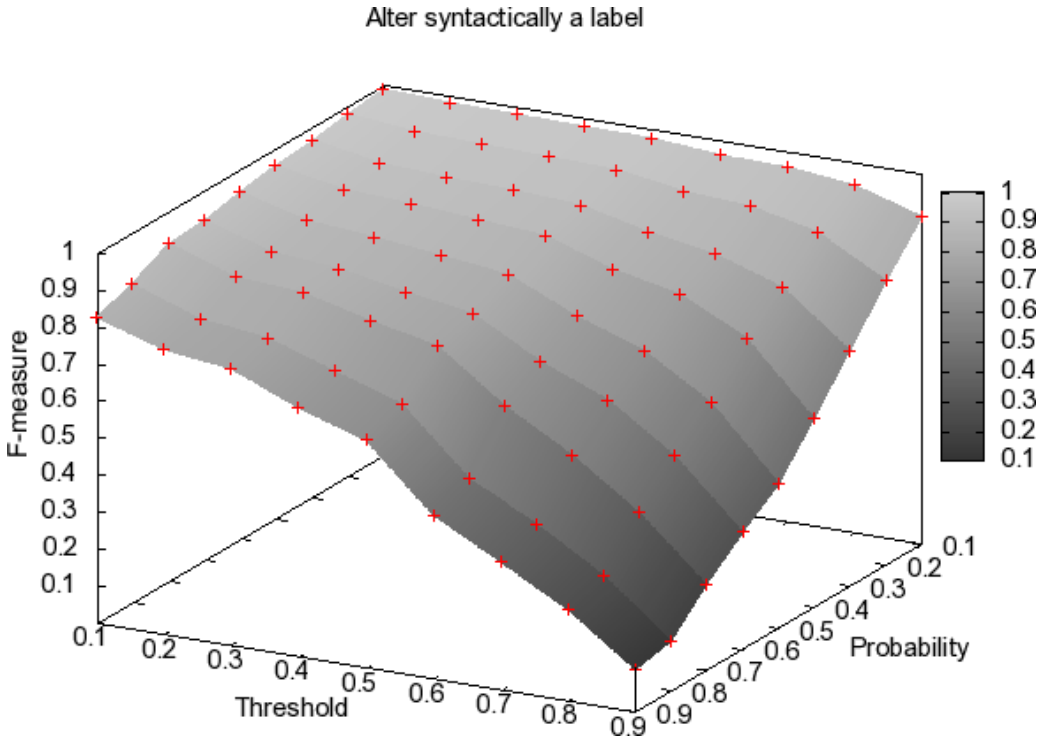


Figure 10.9: F-measure of *Alter syntactically a label* alteration operation.

4. **Replace a label in a node name with a related one:** this alteration operation replaced a label in a node name with a related one, by using synonyms, hyponyms, and hypernyms from a number of generic thesauri. Graphs in Figures 10.10 and 10.11 report on the resulting measures of recall and F-measure. Precision results are not shown as the values were always close to 1. In fact we always used related (i.e., synonyms, hyponyms, and hypernym) terms in the alteration operations. Therefore, almost all the semantic correspondences between the labels were found by SPSM (by construction of the altered set). Thus, a very small number of false positive correspondences were found and precision was always close to 1.

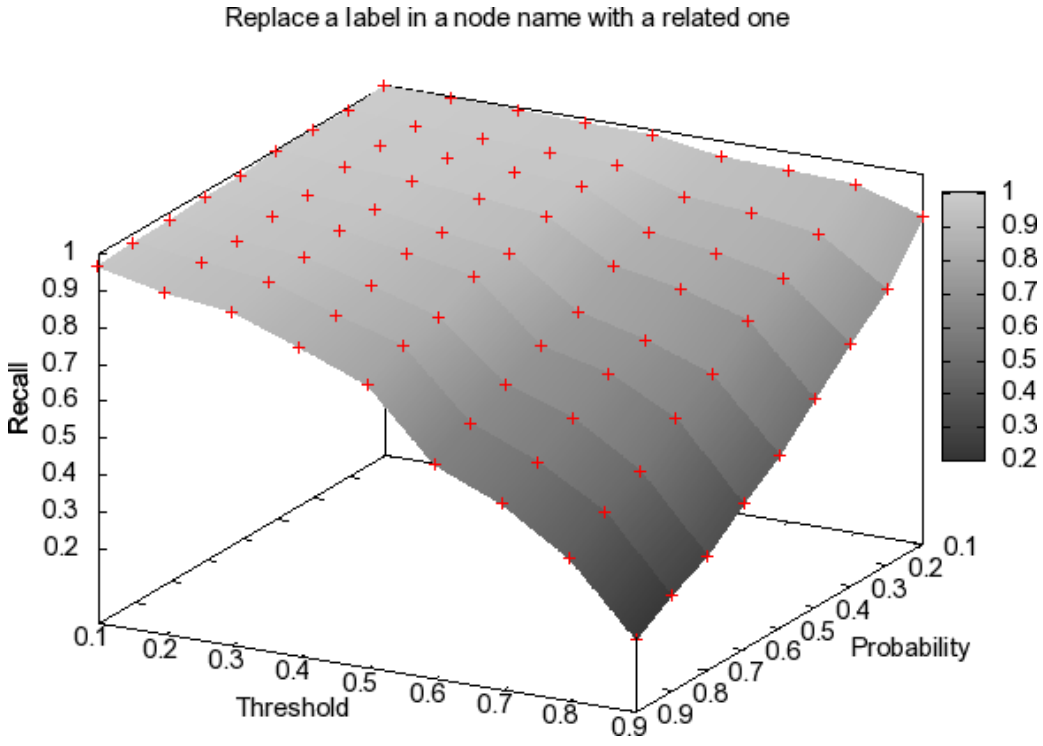


Figure 10.10: Recall of *Replace a label in a node name with a related one* alteration operation.

In this experiment, we evaluated the robustness of the SPSM approach to semantic alterations of the nodes: we did not change the core con-

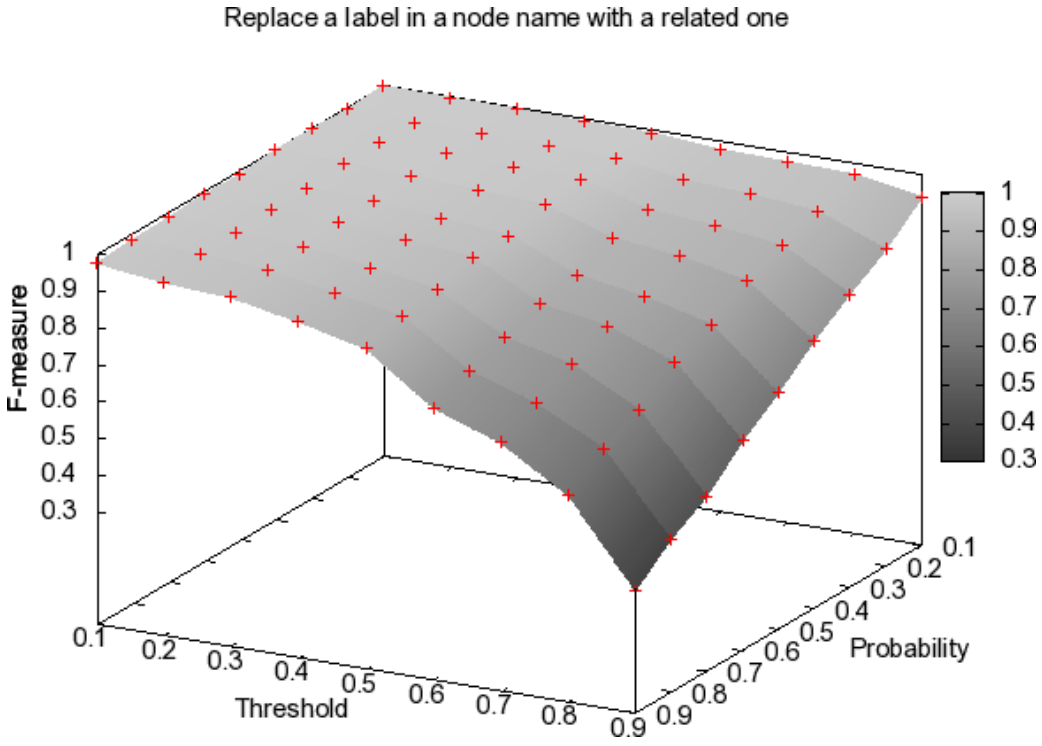


Figure 10.11: F-measure of *Replace a label in a node name with a related one* alteration operation.

cept of the node name, but we used either an *equivalent* or a *more general* or a *less general* label in a node name. In this case recall decreases slowly when both the alteration operation probability and the *TreeSim* cut-off threshold increase.

10.1.1 Comparison of SPSM with baseline matcher.

The goal of this experiment was to compare the SPSM results to a baseline matcher. In order to appropriately compare the two series of results, we used the same evaluation method of the previous experiment. Thus: *(i)* we used the same alteration operations, described in the previous section, to modify the original trees, and *(ii)* we used the results of the previous experiments to identify the best alteration probability to make the comparison between the best results. We made the comparison using all the alteration

operations: *Replace a node name with an unrelated node name*, *Add or remove a label in a node name*, *Alter syntactically a label*, and *Replace a label in a node name with a related one*.

Results for the syntactic modification (*Replace a node name with an unrelated node name*, *Add or remove a label in a node name*, *Alter syntactically a label*) are, as expected, very similar. Therefore, we focused our analysis on the comparison between the node's name syntactic alteration and the the node's names semantic alteration (*Replace a label in a node name with a related one*). Figure 10.12 shows the results when *Replace a node name with an unrelated node name* is applied and Figure 10.13 shows the results when *Replace a label in a node name with a related one* is applied. We plot the results for the most interesting alteration operation probability (60%) for both the syntactic and semantic alterations.

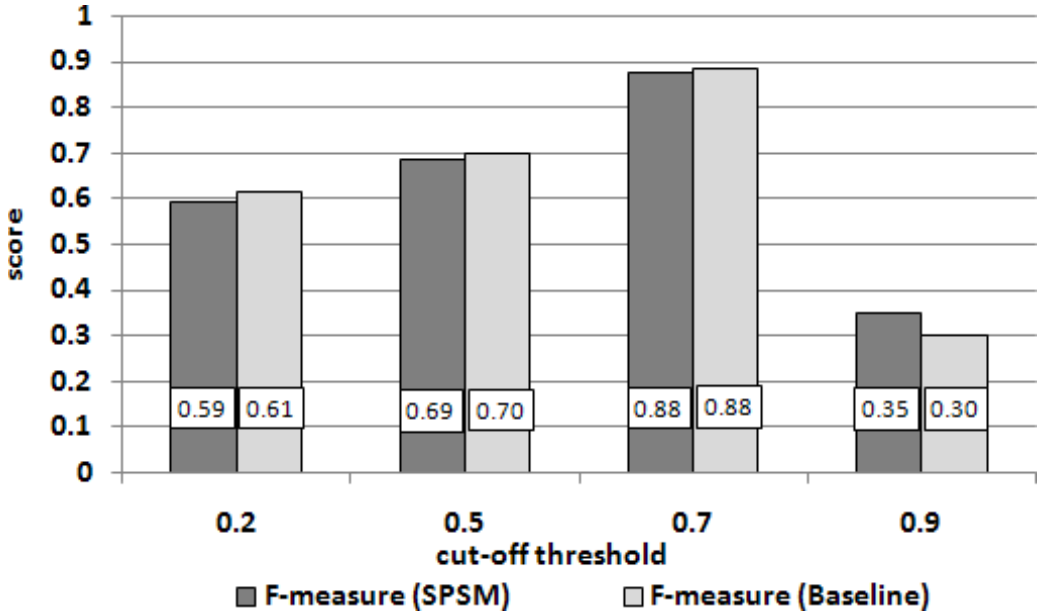


Figure 10.12: SPSM vs. baseline on *Replace a label with an unrelated one* alteration operation.

As Figures 10.12 and 10.13 show, the SPSM approach is always comparable with the baseline matcher when we made syntactic alterations (Figure 10.12). In turn, its results are significantly better than the baseline

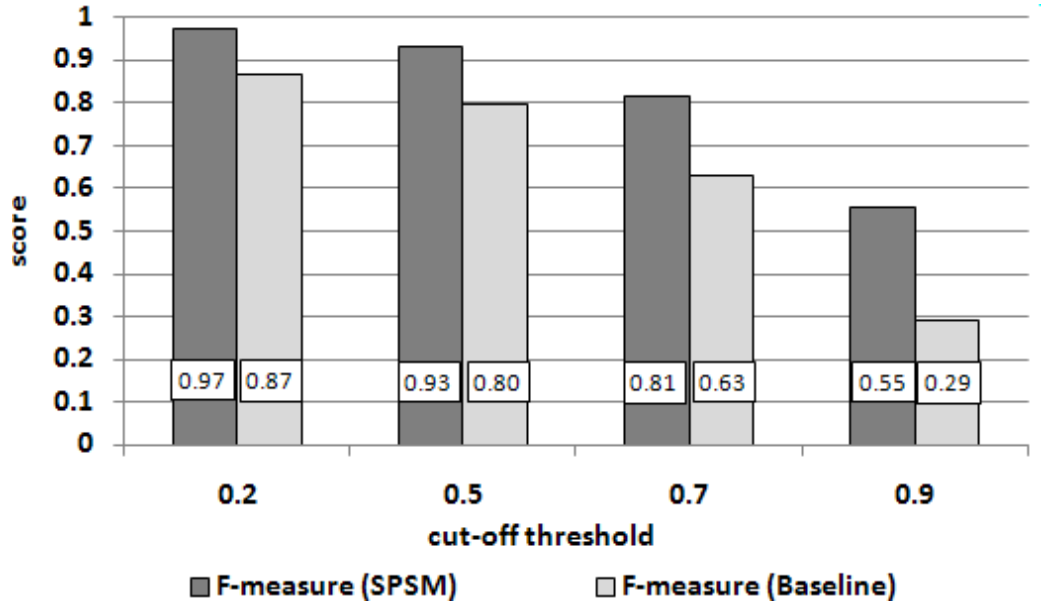


Figure 10.13: SPSM vs. baseline on *Replace a label in a node name with a related one* alteration operation.

matcher (more than 20%, Figure 10.13) when we made semantic alterations.

Figures 10.14 and 10.15 show the comparison between our approach and the baseline when both *Replace a label in a node name with a related one* and *Alter syntactically a label* alterations were combined together. The graphs show the scores of F-measure (we selected an alteration probability of 70%) for both SPSM and edit-distance (baseline) matchers.

Again, the graphs suggest the same conclusion: the SPSM approach behavior is similar to the one of the baseline matcher when syntactic alterations were made, while its performance is constantly better than the baseline when the semantics of the label was modified.

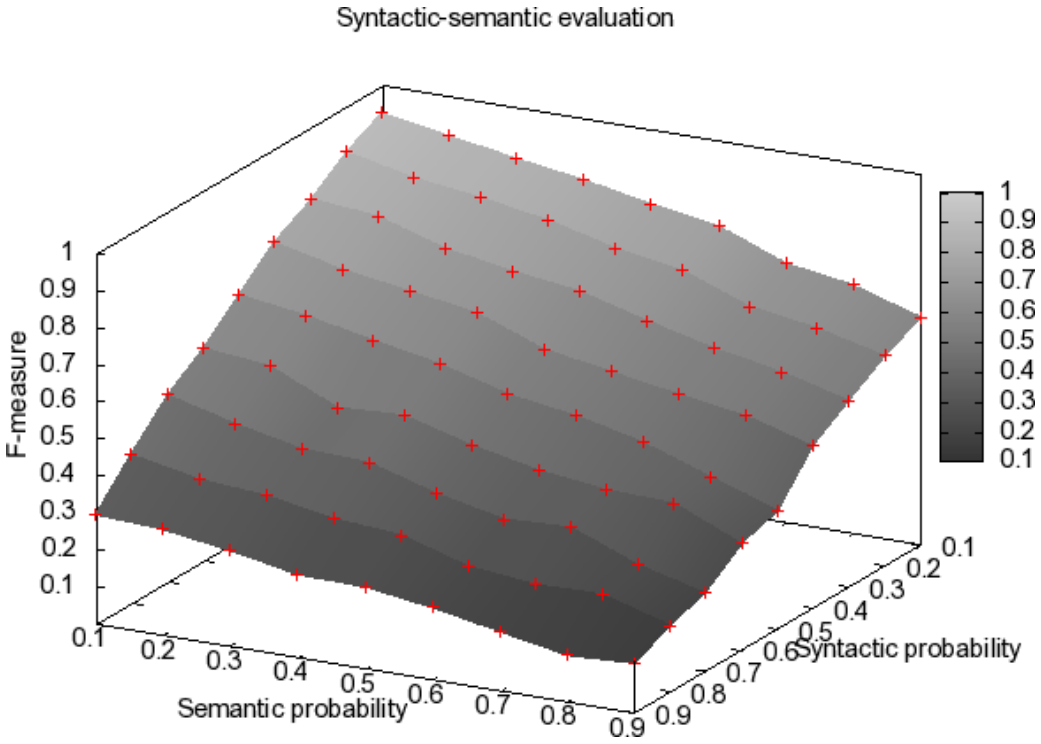


Figure 10.14: F-measure values for SPSM matcher.

10.2 Classification experiment: the results

In this experiment we investigated whether the SPSM approach can be used in determining (in an unsupervised way) the class of a specific GIS operation based on its signature. Figure 10.16 shows, for each *TreeSim* cut-off threshold, recall, precision and F-measure scores. Classification quality measures depend on the cut-off threshold values and the SPSM solution demonstrates good overall matching quality (F-measure) on the wide range of these values. In particular, the best F-measure values exceed 55% for the given GIS operations set (see Figure 10.16: for the *TreeSim* cut-off threshold of 0.5, precision is 0.46, recall is 0.66, and F-measure is 0.55).

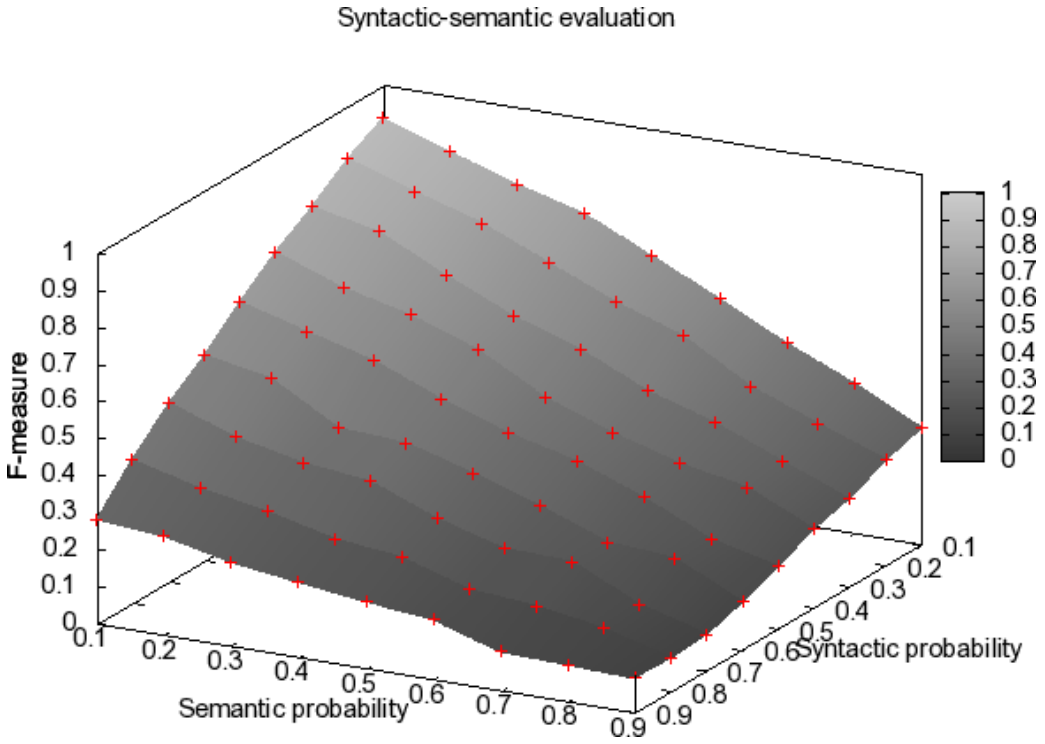


Figure 10.15: F-measure values for edit-distance (baseline) matcher.

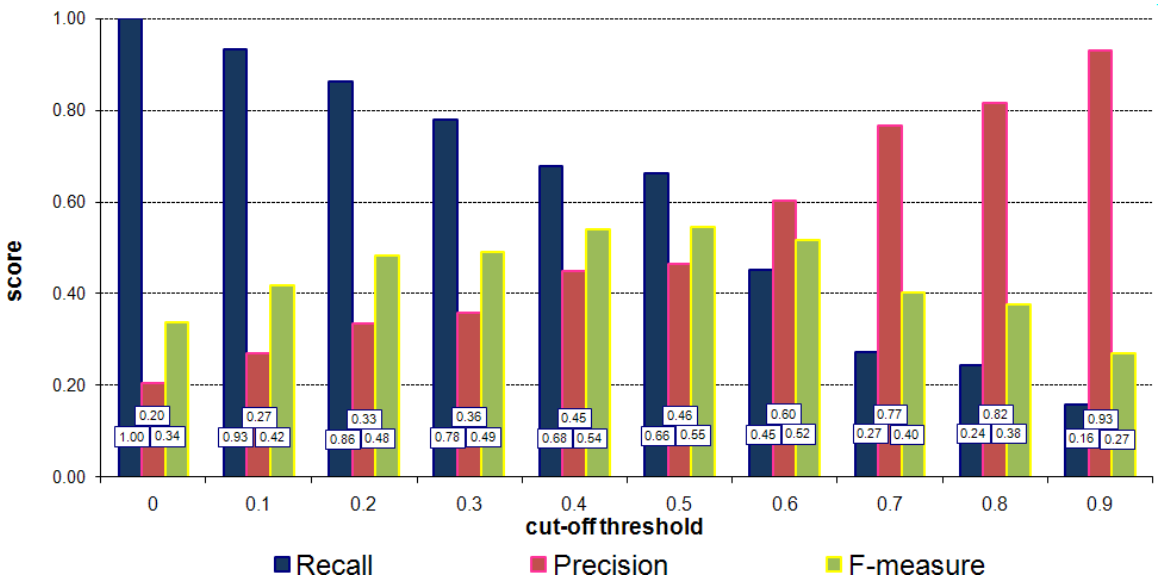


Figure 10.16: Recall, precision and F-measure values for the classification experiment.

10.3 Performance evaluation

The efficiency of the SPSM solution is such that the average execution time per matching task in the evaluation under consideration was 43ms. The

quantity of main memory used by SPSM during matching did not rise more than 2.3MB higher than the standby level. These performance measures suggest that the SPSM is an efficient solution when services have to be dynamically discovered and integrated.

10.4 Evaluation summary

We summarize the evaluation of the SPSM matching approach on the selected set of real-world GIS web services as follows:

SPSM behavior and robustness. We developed evaluation tests to explore the overall behavior and robustness of the SPSM approach towards both typical syntactic and semantic alterations of the GIS service operation signatures. The results showed the robustness of the SPSM algorithm over significant ranges of parameters' variation (cut-off thresholds and alteration operations' probability); while maintaining relatively high (over 50-60%) overall matching relevance quality (F-measure).

Comparison with a baseline matcher (edit-distance) showed how the SPSM approach is always comparable with the baseline when only syntactic alteration are considered, whereas SPSM results were always better (in average more than 20%) when semantic alterations were introduced. This is exactly what one would expect, since the SPSM approach includes a number of state-of-the-art syntactic matchers (that are first used in the internal matching algorithm) plus a number of semantic matchers that enter into play for the alterations in the meaning of nodes labels [43].

SPSM unsupervised clustering capabilities. In this experiment, we investigated how the proposed SPSM approach could be used in de-

termining (in an unsupervised manner) the class of a specific GIS operation directly from the information present in its WSDL operation signature. Classification quality measures depended on the cut-off threshold values and the SPSM solution demonstrated overall good matching quality (i.e., F-measure) on the wide range of these values. In particular, the best F-measure values exceeded 55% for the given GIS operations set. Although, the results are encouraging, still 45% of GIS operation were incorrectly classified, due to the limited knowledge presented in the signatures only. In this case, the presence of more informative and semantically structured annotation would improve significantly the automatic classification at the expense obviously of a greater effort from the designer/programmer.

SPSM performance. Based on all our experiments, the efficiency of the SPSM solution is promising, since the average execution time per matching task was 43 ms and the quantity of main memory was less than 2.3 MB than the standby level.

10.5 Summary

In this chapter we presented an extensive evaluation of the SPSM approach as a practical solution to the semantic heterogeneity problem between different implementations of required geo-services. In the scenario under consideration, we conducted an extensive set of empirical tests to evaluate quality and efficiency indicators of the SPSM approach. We based our tests on a set of real ArcWeb WSDL operations.

Main results are summarize as follows. In the first experiment a high overall matching relevance quality (F-measure) was obtained (over $50 - 60\%$). Moreover, a comparison with a baseline matcher showed how the

SPSM approach is always better (in average by 20%) when semantic alterations are introduced. In the second experiment the best F-measure values exceeded 50% for the given GIS operations set. SPSM performance is good, since the average execution time per matching task was 43 ms. The evaluation results demonstrate robustness and good performance of the SPSM approach on a large (ca. 700.000) number of matching tasks.

Also, we obtained small memory footprint and good matching speed for the given tasks. This suggests that SPSM could be employed to find and integrate similar web service implementations at runtime.

In the following part we will summarize the content of this thesis, its main innovative features, and we will outline future work and application scenarios on the topic of the proposed approach.

Part V

Conclusions

Chapter 11

Summary and future work

In this thesis we have provided a detailed account of the state of the art of interoperability among distributed and heterogeneous geographic information systems. We applied the P2P OpenKnowledge system to a geographic service coordination scenario. The OpenKnowledge system adopts a structure preserving semantic matching solution to discover, integrate and coordinate heterogeneous web services. We evaluated the SPSM solution by conducting experiments on real world ESRI ArcWeb services. In what follows, we present the main contributions of this thesis by summarizing each of the previous chapters.

SDIs are complex information systems whose aim is to support the interoperability between different kinds of GIS service providers and users. In Chapter 1 we discussed motivations and initiatives that are behind the needs of adopting an SDI. We defined the SDI model, we presented a potential SDI technological implementation, and we identified its main components: institutional arrangements, technologies/application development, procedures to access geo-information, and applications to manage fundamental datasets.

A primary issue in the development of SDIs is the advancement of *inter-*

operability, that is one of the key conditions for GIS integration. In Chapter 2 we identified the main dimensions of information systems' interoperability (i.e., autonomy, distribution, dynamics, scalability, and heterogeneity), and we focused on semantic heterogeneity issues of geo-information. Semantic interoperability between heterogeneous geographic data and services is one of the main challenges of modern GIS distributed architectures.

Integrating semantically data and services among heterogeneous geographic providers and users is the fundamental task in order to enable value added services. A key role is played by ontologies, which are machine-accessible representations of knowledge. In Chapter 3 we analyzed the state of the art of the proposed solutions for the semantic heterogeneity issues we identified in the previous chapter. Most of these solutions, both for geographic data and services, adopt a single (top-level) ontology approach. In contrast, in our approach, we assume that geo-data and geo-services are described using terms from different ontologies. Moreover, in this chapter, we presented recent advances in ontology matching and related evaluation approaches.

In our approach we used a P2P technology to coordinate GIS services within an e-Response scenario. P2P technology represents a novel approach in the architecture and system design of collaborative *geo-applications*. In Chapter 4 we defined the main characteristics of the P2P model and we presented developments in this research field w.r.t. geo-applications.

Geographic information acquisition, use, and integration are crucial activities in all the phases of an emergency situation (i.e., assessment and prevention, preparation, response, and recovery management). In Chapter 5 we showed a natural disaster scenario as an example of such activities and as the motivating overall scenario of our approach. Specifically, within this scenario, we focused on the coordination scenario among different SDI services: the gazetteer, the map, and the download services.

We developed coordination between e-Response services by using a P2P interaction-oriented approach, based on the LCC language and on the OpenKnowledge system. In Chapter 6 we presented the main characteristics of this system and of the SPSM module. In Chapter 7 we developed specific coordination protocols in order to support the e-Response scenario and, specifically, we illustrated how we implemented the SDI services we introduced in Chapter 5.

The adopted SPSM technique is used by the OpenKnowledge system to match invocation of web services and web service descriptions. In order to evaluate the behavior, robustness, unsupervised clustering capabilities, and the performance of the SPSM approach, we conducted two kinds of experiments on real world GIS ESRI ArcWeb services. The first experiment included matching of original web service signatures to synthetically altered ones. In the second experiment we compared a manual classification of our dataset to the unsupervised one produced by SPSM. In Chapters 8, 9, and 10 we described the evaluation dataset, we presented the experimental setup, we illustrated the adopted evaluation method, and we discussed the results of the experiments.

In what follows, we will first underline how the used P2P semantic matching approach and its evaluation fulfil the requirements of the dimensions of interoperability (§11.1). Then, we will discuss the applicability of the evaluated SPSM approach (§11.2). Finally, we will outline future work (§11.3).

11.1 Dimensions of interoperability

In this thesis we applied a P2P semantic matching framework to the coordination of three geographic web services, namely the gazetteer, the download, and the map services. Moreover, we provided a detailed evaluation

of the semantic matching solution, used by the framework and which we consider fundamental to discover, integrate and coordinate geographic web services. Specifically, we evaluate the semantic matching approach on a real set of ArcWeb GIS Web services by conducting two kinds of experiments, i.e., the evolution experiment and the classification experiment.

We consider the framework proposed in this thesis and its evaluation novel in the field of heterogeneous distributed systems and, specifically, for distributed SDI model. We underline that the P2P semantic matching approach and its evaluation fulfil the requirements of the dimensions we presented in §2.1. In particular:

- **Autonomy.** The OpenKnowledge system supports different types of autonomy including: *communication autonomy*, i.e., peers participating to OpenKnowledge IMs decide whether to communicate with other components by satisfying IM constraints; *association autonomy*, i.e., peers have the ability to decide how and how much to share their functionalities and resources; and *participation autonomy*, i.e., peers have the ability to associate or disassociate themselves from one or more distributed systems.

- **Distribution.** The OpenKnowledge system supports *physical distribution*, i.e., data can be stored in a central peer or reside on different peers which are geographically distributed and connected, and *operational distribution*, i.e., by the means of DTS there exists a global shared register. The DTS is a repository of content, and it is used to publish, discover and retrieve IMs and OKCs. Moreover, DTS coordinates subscription, it chooses a coordinator and it provides team formation and interaction initialization. DTS can be either centralized [27] or distributed [68].

- **Dynamics.** The OpenKnowledge system supports dynamics and it is

capable to take autonomously actions in order to compensate changes in peers behavior, by means of its subscription mechanism, the trust module, and the matcher module [109].

- **Scalability.** Within the OpenKnowledge project, we conducted a number of tests, that demonstrated: (*i*) the overall scalability of the OpenKnowledge kernel on realistic use cases, and (*ii*) the capability of the OpenKnowledge system to support centralized as well as decentralized architectures for information gathering in e-Response management [135].

- **Heterogeneity.** One of the main goals of the OpenKnowledge system is to support both *syntax heterogeneity* and *semantic heterogeneity* when the invocation of web services and their descriptions are expressed in a different way. The extensive evaluation of these aspects on real GIS services is the main focus of this thesis work. The robustness and the unsupervised clustering capabilities of the evaluation results on the SPSM approach demonstrated that it can be effectively applied when syntactic and semantic discovering and chaining of web services are required. Moreover, SPSM performances suggest that SPSM could be employed to find similar web service implementations at runtime.

11.2 Application scenarios

The proposed approach can can be fruitfully applicable in the following GIS-specific application scenarios.

- **GIS Web service discovery and integration.** Reusing existing web services such as, e.g., WSDL-specified services for building web-based applications, is a very important issue in modern web applications. Discovery of web services based on a classification method (like

the one proposed, e.g., by the UDDI standard) is insufficient, because providers classify services on the shared common-sense understanding of their application domain. SPSM can be used to support a more automated discovery and use of web services, by distinguishing among the potentially useful and the likely irrelevant services, and by ordering the potentially useful ones according to their relevance. SPSM can be effectively applied both during services discovering, by considering how close is the numerical similarity between two signatures, and during the composition or the coordination of web services, by using the correspondences between signatures of input and output parameters. Specifically, for geo-information catalogs [106], the presented SPSM approach could be easily and effectively applied to discovery and chaining geo-services from catalogs of geospatial information and related resources.

- **WPS service composition.** The proposed approach and, specifically, the SPSM solution, could be applied in semi-supervised discovery and composition of geo-processing services which follow the WPS specifications (see §3). The WPS standard defines an interface that facilitates the publishing of geospatial processes and makes it easier to write software clients that can discover and bind to those processes. The WPS specification includes guidelines on how to publish processing services that perform modeling, calculation and elaboration of both vector and raster geo-data[1]. The number of services that implements the WPS specification is increasing day by day. Figure 11.1[2] shows a GUI of a 3D geo-browser that allows the user to select and execute WPS services. In such a case, manual discovery and composition of WPS services can be very difficult (e.g., the initial prototype

[1] http://www.opengeospatial.org/pressroom/pressreleases/843
[2] http://www.graphitech.it/

of this geo-web application, provides more than 200 WPS services).

Figure 11.1: WPS services catalog. Courtesy of Fondazione Graphitech, Trento, Italy.

- **Geo-sensor networks.** The proposed approach can be applied to discovery and chaining geo-sensor services. Geo-sensors can be defined as any device receiving and measuring environmental stimuli that can be geographically referenced. Geo-sensors include different

kinds of sensors like satellite-based sensors, air-borne sensors, and sensors near, on, or under the Earth's surface measuring anything from physical characteristics (pressure, temperature, humidity) and phenomena (wind, rain, earthquakes), human sensors, tracking sensors, smart dust sensors. Large-scale networks of sensors have been in existence for several decades. New opportunities to use this spatially referenced information is now given by the increasing availability of geo-sensor web services that can be discovered, accessed and chained through web standards (SWE[3] [105]).

Wireless sensor networks (WSN), are changing the way we acquire geo-referenced information. New wireless sensors are small, typically connected to a wireless network, very low-power consuming and very low-cost devices. They will provide a huge amount of spatially referenced information along with high acquisition frequency [25]. Moreover, a set of developments within the category of geo-sensors is that of citizens as sensors, volunteering geographic information. The work in [50], illustrates the potential of up to 6 billion human sensors to monitor the state of the environment, validate global models with local knowledge, and provide information that only humans can capture. In general SDI model is not designed to manage geo-sensor-based data, which tend to arrive in real-time, are more stream-like and are organized in highly dynamic distributed system. Recently, some efforts are dedicated to explore whether the core ideas and technologies of the Semantic Web can also be applied to sensor networks to allow the development of an open information space which is called the Semantic Sensor Web[4]. The work in [121] discusses OGC specification about these issues and focuses on the integration of sensor technologies

[3] http://www.opengeospatial.org/projects/groups/sensorweb

[4] http://semsensweb.di.uoa.gr/

and Semantic Web technologies. Our approach can support the new flexibility this sensor-based scenario requires by adopting both P2P technologies and recent advances in semantic matching approaches.

11.3 Future work

In this section we delineate future work. In particular, we proceed in the following directions: (*i*) by applying and evaluating the proposed approach in different scenarios, and (*ii*) by extending the SPSM solution with different techniques.

11.3.1 Applications and evaluation

The following directions are to be pursued:

- Application of semantic discovery and composition on WPS geo-services and geo-data published in the SDI catalog of the Autonomous Province of Trento.
- Application of semantic coordination on distributed geo-sensors web services in a real world emergency scenario.
- Extensive and comparative evaluation of the matching approach on different kinds of GIS web services like the ones available from OGC specifications and from the GRASS package.
- Evaluation of the SPSM solution on GIS data ontologies, like the ones provided by the INSPIRE directive.

11.3.2 Extends of the SPSM solution

The following directions are to be pursued:

- Incorporation of domain specific preferences in order to drive approximation, thus allowing/prohibiting certain kinds of approximation (e.g., not approximating vector maps with raster maps, although these are both maps).

- Use of different kinds of thesauri like the multilingual GEMET or AGROVOC thesauri to support GIS specific terminology and multilingual matching.

- Extension of SPSM to perform spatial matching. Besides handling the meaning of names of the entities, spatial matching has to be performed, that is looking for the same geometry, similar spatial relationships, etc.

Bibliography

[1] Rohit Aggarwal, Kunal Verma, John Miller, and William Milnor. Constraint driven web service composition in METEOR-S. In *Proceedings of the 1st IEEE International Conference of Services Computing (SCC)*, pages 23–30, 2004.

[2] Rama Akkiraju, Joel Farrell, John Miller, Meenakshi Nagarajan, Marc-Thomas Schmidt, Amit Sheth, and Kunal Verma. Web Service Semantics - WSDL-S. Technical report, W3C, 2005.

[3] George Anadiotis, Paolo Besana, David de la Cruz, Dave Dupplaw, Frank van Harmelen, Spyros Kotoulas, Juan Pane, Adrian Perreau de Pinninck, Marco Schorlemmer, Ronny Siebes, and Lorenzino Vaccari. The openknowledge system: an interaction-centered approach to knowledge sharing. *In preparation*, 2009.

[4] Tony Andrews, Francisco Curbera, Hitesh Dholakia, Yaron Goland, Johannes Klein, Frank Leymann, Kevin Liu, Dieter Roller, Doug Smith, Satish Thatte, Ivana Trickovic, and Sanjiva Weerawarana. Business process execution language for web services, version 1.1. Technical report, BEA Systems, International Business Machines Corporation,Microsoft Corporation, SAP AG, Siebel Systems, 2003.

[5] Stephanos Androutsellis-Theotokis and Diomitis Spinellis. A survey of peer-to-peer content distribution technologies. *ACM Computing Surveys*, 36(4):335–371, 2004.

[6] Grigoris Antoniou and Frank van Harmelen. *Web Ontology Language: OWL*. Springer-Verlag, 2003.

[7] Franz Baader, Diego Calvanese, Deborah McGuinness, Daniele Nardi, and Peter Patel-Schneider, editors. *The description logic handbook: theory, implementations and applications*. Cambridge University Press, 2003.

[8] Carlo Batini, Maurizio Lenzerini, and Shamkant Navathe. A comparative analysis of methodologies for database schema integration. *ACM Computing Surveys*, 18(4):323–364, 1986.

[9] Steve Battle, Abraham Bernstein, Harold Boley, Benjamin Grosof, Michael Gruninger, Richard Hull, Michael Kifer, David Martin, Sheila McIlraith, Deborah McGuinness, Jianwen Su, and Said Tabet. Semantic Web Services Framework (SWSF) Overview. Technical report, W3C, 2005.

[10] Lars Bernard, Max Craglia, Michael Gould, and Werner Kuhn. Towards an SDI research agenda. In *Proceedings of the 11th European Commission-Geographic Information (EC-GI) and GIS Workshop*, pages 147–151, 2005.

[11] Tim Berners-Lee, J. Hendler, and O. Lassila. The Semantic Web. *Scientific American Magazine*, 2001.

[12] Paolo Besana, Fiona McNeill, Fausto Giunchiglia, Lorenzino Vaccari, Gaia Trecarichi, and Juan Pane. Web service integration via matching of interaction specifications. Technical report, University of Trento, Dipartimento di Ingegneria e Scienza dell'Informazione, 2008.

[13] Yaser Bishr. *Semantic aspects of interoperable GIS*. PhD Dissertation, International Institute for Aerospace Survey and Earth Sciences, Enschede, The Netherlands, 1997. ITC Publication No. 56, 154 pp.

[14] Yuri Breitbart. Multidatabase interoperability. *SIGMOD Record*, 19(3):53–60, 1990.

[15] Jos De Bruijn, Holger Lausen, Axel Polleres, and Dieter Fensel. The web service modeling language wsml: An overview. Technical report, DERI - Digital Enterprise Research Institute, 2005.

[16] Lorenzo Bruzzone, Paul C. Smits, and James C. Tilton. Foreword special issue on analysis of multitemporal remote sensing images. *IEEE Transactions on GeoScience and Remote Sensing*, 41(11):2419–2422, 2003.

[17] Caterina Caracciolo, Jérôme Euzenat, Laura Hollink, Ryutaro Ichise, Antoine Isaac, Véronique Malaisé, Christian Meilicke, Juan Pane, Pavel Shvaiko, Heiner Stuckenschmidt, Ondřej Šváb Zamazal, and Vojtěch Svátek. Results of the ontology alignment evaluation initiative 2008. In *Proceedings of the International Workshop on Ontology Matching (OM) at the 7th International Semantic Web Conference (ISWC)*, pages 73–119, 2008.

[18] Tiziana Catarci, Fabio De Rosa, Massimiliano de Leoni, Massimo Mecella, Michele Angelaccio, Schahram Dustdarz, Begona Gonzalvez, Giuseppe Iiritano, Alenka Krek, Guido Vetere, and Zdenek M. Zalis. Workpad: 2-layered peer-to-peer for emergency management through adaptive processes. In *Proceedings of the International Conference on Collaborative Computing: Networking, Applications and Worksharing*, pages 1–9, 2006.

[19] Liu Chen, Ma Xiujun, Chen Guanhua, Sun Yanfeng, and Feng Xuebing. A peer-to-peer architecture for dynamic executing gis web service composition. In *Proceedings of the International Geoscience and Remote Sensing Symposium, 2005 (IGARSS'05)*, pages 979–982, 2003.

[20] Ching chien Chen, Cyrus Shahabi, and Craig A. Knoblock. Utilizing road network data for automatic identification of road intersections from high resolution color orthoimagery. In *In Proceedings of the 2nd Workshop on Spatio-Temporal Database Management(STDBM'04)*, pages 1477–1480, 2004.

[21] Ching chien Chen, Snehal Thakkar, Craig Knoblock, and Cyrus Shahabi. Automatically annotating and integrating spatial datasets. In *Proceedings of the 8th International Symposium on Spatial and Temporal Databases (SSTD)*, pages 469–488, 2003.

[22] Erik Christensen, Francisco Curbera, Greg Meredith, and Sanjiva Weerawarana. Web Services Description Language (WSDL) 1.1. Technical report, W3C, 2001.

[23] European Commission. Towards a Shared Environmental Information System (SEIS). Technical report, European Commission, 2008.

[24] Lewis M. Cowardin, Virginia Carter, Francis C. Golet, and Edward T. LaRoe. *Classification of Wetlands and Deep water Habitats of the United States.* U.S. Department of the Interior, Fish and Wildlife Service, Washington, D.C. Jamestown, ND: Northern Prairie Wildlife Research Center, 1979.

[25] Max Craglia, Michael F. Goochild, Alessandro Annoni, Gilberto Câmara, Michael Gould, Werner Kuhn, David Mark, Ian Masser,

David Maguire, Steve Liang, and Ed Parsons. Next-generation digital earth. *International Journal of Spatial Data Infrastructures Research*, 3:146–167, 2008.

[26] Isabel F. Cruz and William Sunna. Structural alignment methods with applications to geospatial ontologies. *Transactions in GIS, special issue on Semantic Similarity Measurement and Geospatial Applications*, 12(6):683–711, 2008.

[27] Adrian Perreau de Pinninck, David Dupplaw, Spyros Kotoulas, and Ronny Siebes. The openknowledge kernel. *International Journal of Applied Mathematics and Computer Sciences (IJAMCS)*, 4(3):162–167, 2007.

[28] Liping Di. The implementation of geospatial web services at geobrain. In *Proceedings of the 2005 NASA Earth Science Technology Conference (CDROM)*, 2005.

[29] Liping Di, Peisheng Zhao, Wenli Yang, and Peng Yue. Ontology-driven automatic geospatial-processing modeling based on web-service chaining. In *Proceedings of the 6th Earth Science Technology Conference (ESTC) - CDROM*, 2006.

[30] John Dini, G. Gowan, and Peter Goodman. South african national wetland inventory, proposed wetland classification system for south africa. Technical report, South African Wetlands Conservation Programme, 1998.

[31] Thomas Erl. *Service-Oriented Architecture: Principles of Service Design*. Prentice Hall, 2005.

[32] Jérôme Euzenat. Towards composing and benchmarking ontology alignments. In *Proceedings of the Workshop on Semantic Integration*

at the International Semantic Web Conference (ISWC), pages 165–166, 2003.

[33] Jérôme Euzenat, Antoine Isaac, Christian Meilicke, Pavel Shvaiko, Heiner Stuckenschmidt, Ondřej Šváb, Vojtech Svátek, Willem Robert van Hage, and Mikalai Yatskevich. Results of the ontology alignment evaluation initiative OAEI. In *Proceedings of the Workshop on Ontology Matching (OM) at the 6th International Semantic Web Conference (ISWC) + the 2nd Asian Semantic Web Conference (ASWC)*, pages 96–132, 2007.

[34] Jérôme Euzenat and Pavel Shvaiko. *Ontology matching*. Springer, 2007.

[35] Joel Farrell and Holger Lausen. Semantic Annotations for WSDL and XML Schema. Technical report, W3C, 2007.

[36] Dieter Fensel. *Ontologies: A Silver Bullet for Knowledge Management and Electronic Commerce*. Springer Verlag, 2004.

[37] Renato Fileto, Ling Liu, Calton Pu, Eduardo Delgado Assad, and Claudia Bauzer Medeiros. POESIA: an ontological workflow approach for composing web services in agriculture. *The VLDB Journal*, 12(4):352–367, 2003.

[38] Frederico Fonseca, Max EgenHofer, Peggy Agouris, and Gilberto Câmara. Using ontologies for integrated geographic information systems. *Transactions in GIS*, 6(3):231–257, 2002.

[39] Timothy W. Foresman. Evolution and implementation of the digital earth vision, technology and society. *International Journal of Digital Earth*, 1(1):4–16, 2008.

[40] Ian Foster and Carl Kesselman. *The GRID: blueprint for a new computing infrastructure*. Morgan Kaufmann Publishers Inc. San Francisco, CA, USA, 1998.

[41] Anders Friis-Christensen. Approaches to solve schema heterogeneity at european level. In *Proceeding of the 11th EC-GI & GIS Workshop, ESDI: Setting the Framework*, 2005.

[42] Fausto Giunchiglia, Maurizio Marchese, and Ilya Zaihrayeu. Encoding classifications into lightweight ontologies. *Journal of Data Semantics*, VIII:57–81, 2007.

[43] Fausto Giunchiglia, Fiona McNeill, Mikalai Yatskevich, Juan Pane, Paolo Besana, and Pavel Shvaiko. Approximate structure-preserving semantic matching. In *Proceedings of the 7th Conference on Ontologies, DataBases, and Applications of Semantics (ODBASE)*, pages 1234–1237, 2008.

[44] Fausto Giunchiglia and Pavel Shvaiko. Semantic matching. *The Knowledge Engineering Review*, 18(3):265–280, 2003.

[45] Fausto Giunchiglia and Toby Walsh. A theory of abstraction. *Artificial Intelligence*, 57(2-3):323–389, 1992.

[46] Fausto Giunchiglia, Mikalai Yatskevich, Paolo Avesani, and Pavel Shvaiko. A large scale dataset for the evaluation of ontology matching systems. *To appear in The Knowledge Engineering Review Journal*, 24(2), 2009.

[47] Fausto Giunchiglia, Mikalai Yatskevich, and Enrico Giunchiglia. Efficient semantic matching. In *Proceedings of the 2nd European Semantic Web Conference (ESWC)*, pages 272–289, 2005.

[48] Fausto Giunchiglia, Mikalai Yatskevich, and Pavel Shvaiko. Semantic matching: Algorithms and implementation. *Journal on Data Semantics*, IX:1–38, 2007.

[49] Moses Gone and Sven Shade. Towards semantic composition of geospatial web services using WSMO in comparison to BPEL. In *Proceedings of the 5th Geographic Information Day - Young Researchers Forum*, pages 43–63, 2007.

[50] Michael F. Goodchild. Citizens as voluntary sensors: spatial data infrastructure in the world of Web 2.0. *International Journal of Spatial Data Infrastructures Research*, 2:24–32, 2007.

[51] Michael F. Goodchild. The use cases of digital earth. *International Journal of Digital Earth*, 1(1):31–42, 2008.

[52] Al Gore. The digital earth: Understanding our planet in the 21st century. *Photogrammetric Engineering and Remote Sensing*, 65(5):528, 1998.

[53] Richard Groot and John McLaughlin. *Geospatial Data Infrastructure: Concepts, Cases and Good Practice*. Oxford University Press, 2000.

[54] Karl E. Grossner. Is Google Earth "Digital Earth"?-Defining a Vision. In *Proceedings of the 5th International Symposium on Digital Earth (ISDE)*, 2006.

[55] Thomas Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5(2):199–220, 1993.

[56] Jihong Guan, Leichun Wang, and Shuigeng Zhou. Enabling gis services in a p2p environment. In *Proceedings of the 1st International Conference on Formal Ontologies in Information Systems (FOIS)*, pages 776–781, 2004.

[57] Nicola Guarino. Formal ontology and information systems. In *Proceedings of the 1st International Conference on Formal Ontologies in Information Systems (FOIS)*, pages 3–15, 1998.

[58] Armin Haller, Emilia Cimpian, Adrian Mocan, Eyal Oren, and Christoph Bussler. WSMX - a semantic service-oriented architecture. In *Proceedings of the International Conference on Web Service (ICWS)*, pages 321–328, 2005.

[59] Guillermo Nudelman Hess, Cirano Iochpe, and Silvana Castano. An algorithm and implementation for geoontologies integration. In *Proceedings of the 8th Symposium on GeoInformatics*, pages 129–140, 2006.

[60] IEEE. IEEE standard computer dictionary. a compilation of IEEE standardcomputer glossaries. Technical report, IEEE, 1991.

[61] Project INTERREG IIIC. A multilingual glossary of civil protection for EU citizens. http://www.mo-di.net, 2005.

[62] INSPIRE. Directive 2007/2/ec of the european parliament and of the council of 14 march 2007 establishing an infrastructure for spatial information in the european community (inspire). *Official Journal of the European Union*, L108:1–14, 2007.

[63] Krzysztof Janowicz, Carsten Keßler, Mirco Schwarz, Marc Wilkes, Ilija Panov, Martin Espeter, and Boris Bäumer. Algorithm, implementation and application of the SIM-DL similarity server. *GeoSpatial Semantics*, 4853/2007:128–145, 2007.

[64] Krzysztof Janowicz, Marc Wilkes, and Michael Lutz. Similarity-based information retrieval and its role within spatial data infrastructures. *Geographic Information Science*, 5266:151–167, 2008.

[65] Vipul Kashyap and Amit Sheth. Semantic heterogeneity in global information systems: The role of metadata, context and ontologies. In Michael Papazoglou and Gunter Schlageter, editors, *Cooperative Information Systems*, pages 139–178. Academic Press, 1998.

[66] Michel Klein. Combining and relating ontologies: an analysis of problems and solutions. In *Proceedings of the Workshop on Ontologies and Information Sharing at the International Joint Conference on Artificial Intelligence (IJCAI)*, 2001.

[67] Eva Klien. Automating the semantic annotation of geodata. *Transaction in GIS, Special Issue on the Geospatial Semantic Web*, 11(3):437–452, 2007.

[68] Spyros Kotoulas and Ronny Siebes. Deliverable 2.2: Adaptive routing in structured peer-to-peer overlays. Technical report, OpenKnowledge, 2007.

[69] Alenka Krek and Manfred Bortenschlager. P2p computing and geoinformation technologies: Research and application challenges. In *Proceedings of the 15th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE'06)*, pages 87–88, 2006.

[70] Alenka Krek, Massimo Mecella, and Francesco Manti. Dynamic peer-to-peer based geoinformation services supporting mobile emergency management teams. In Coors, Rumor, Fendel & Zlatanova, editor, *Urban and regional data management*, pages 369–379. Taylor and Francis group, 2007.

[71] Werner Kuhn. Geospatial semantics: Why, of what, and how? *Journal on Data Semantics, special issue on Semantic-based Geographical Information Systems*, 3534:1–24, 2005.

[72] Dominik Kuropka and Mathias Weske. Implementing a semantic service provision platform: Concepts and experiences. *Special Issue on Service Oriented Architectures and Web Services of Journal Wirtschaftsinformatik*, 1:16–24, 2008.

[73] Yvan Leclerc, Martin Reddy, Lee Iverson, and Aaron Heller. The geoweb - a new paradigm for finding data on the web. In *Proceedings of the 20th International Cartographic Conference (ICC)*, 2001.

[74] Rob Lemmens, Andreas Wytzisk, Rolf de By, Carlos Granell, Michael Gould, and Peter van Oosterom. Integrating semantic and syntactic descriptions to chain geographic services. *IEEE Internet Computing*, 10(5):42–52, 2006.

[75] Ron Lemmens. *Semantic interoperability of distributed geo-services*. PhD Dissertation, Delft University of Technology, 2006. ISBN: 90-6164-250-7.

[76] Michael Lutz. Ontology-based service discovery in spatial data infrastructures. In *Proceedings of the 2nd Workshop on Geographic Information Retrieval*, pages 45–54, 2005.

[77] Michael Lutz and Eva Klien. Ontology-based retrieval of geographic information. *International Journal of Geographic Information Science*, 20(3):233–260, 2006.

[78] Michael Lutz, Roberto Lucchi, Anders Friis-christensen, and Nicole Ostländer. A rule-based description framework for the composition of geographic information services. *Geospatial Semantics*, 4853:114–127, 2007.

[79] Maurizio Marchese, Ivanyuckovich Alexander, and Lorenzino Vaccari. A web service approach to geographical data distribution among pub-

lic administrations. In *Proceedings of the 5th IFIP Conference on e-Commerce, e-Business, and e-Government (I3E)*, volume 189, pages 329–343, 2005.

[80] Maurizio Marchese, Lorenzino Vaccari, Pavel Shvaiko, and Juan Pane. An application of approximate ontology matching in eResponse. In *Proceedings of the 5th International Conference on Information Systems for Crisis Response and Management (ISCRAM)*, pages 294–304, 2008.

[81] Maurizio Marchese, Lorenzino Vaccari, Gaia Trecarichi, Nardine Osman, and Fiona McNeill. Interaction models to support peer coordination in crisis management. In *Proceedings of the 5th International Conference on Information Systems for Crisis Response and Management (ISCRAM)*, pages 230–241, 2008.

[82] Maurizio Marchese, Lorenzino Vaccari, Gaia Trecarichi, Nardine Osman, Fiona McNeill, and Paolo Besana. An interaction-centric approach to support peer coordination in distributed emergency response management. *Special Issue on Intelligent Decision Making in Dynamic Environments: Methods, Architectures and Applications of the Intelligent Decision Technologies (IDT): An International Journal*, 3(1), 2009.

[83] Maurizio Marchese, Lorenzino Vaccari, Gaia Trecarichi, Pavel Shvaiko, Juan Pane, Nardine Osman, and Fiona McNeill. Openknowledge deliverable 6.7: Interaction models for eResponse. Technical report, OpenKnowledge, 2008.

[84] David Martin, Mark Burstein, Jerry Hobbs, Ora Lassila, Drew McDermott, Sheila McIlraith, Srini Narayanan, Massimo Paolucci, Bijan Parsia, Terry Payne, Evren Sirin, Naveen Srinivasan, and Katia

Sycara. OWL-S: Semantic Markup for Web Services. *W3C Member Submission*, 22, 2004.

[85] Ian Masser. *Creating Spatial Data Infrastructures.* ESRI Press - RedLands - California, 2005.

[86] Ian Masser. *Building European Statial Data Infrastructures.* ESRI Press - RedLands - California, 2007.

[87] Patrick Maué. An extensible semantic catalogue for geospatial web services. *International Journal of Spatial Data Infrastructures Research*, 3:168–191, 2008.

[88] Sergey Melnik, Hector Garcia-Molina, and Erhard Rahm. Similarity flooding: A versatile graph matching algorithm and its application to schema matching. In *Proceedings of the 18th IEEE International Conference on Data Engineering (ICDE)*, pages 117–128, 2002.

[89] Martin Michalowski and Craig A. Knoblock. A constraint satisfaction approach to geospatial reasoning. In *Proceedings of the 17th conference on Innovative Applications of Artificial Intelligence (IAAA)*, pages 423–429, 2005.

[90] Rimon Mikhaiel and Eleni Stroulia. Examining usage protocols for service discovery. In *Proceedings of the 4th International Conference on Service Oriented Computing (ICSOC)*, pages 496–502, 2006.

[91] George A. Miller. Wordnet: A lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995.

[92] Robin Milner, Joachim Parrow, and David Walker. A calculus of mobile processes (part 1/2). *Information and Computation*, 100(1):177, 1992.

[93] Dejan S. Milojicic, Vana Kalogeraki, Rajan Lukose, Kiran Nagaraja, Jim Pruyne, Bruno Richard, Sami Rollins, and Zhichen Xu. Peer-to-peer computing. Technical report, HP Laboratories, 2002.

[94] Ullas Nambiar, Kai Lin, Bertram Ludaescher, and Chaitan Baru. The geon portal: accelerating knowledge discovery in the geosciences. In *Proceedings of the 8th ACM International Workshop on Web Information and Data Management (WIDM)*, pages 83–90, 2006.

[95] National Research Council. *Toward a coordinated spatial data infrastructure for the nation.* Mapping Science Committee. Washington, DC: National Academy Press, 1993.

[96] Douglas Nebert. *Developing Spatial Data Infrastructures. The SDI CookBook.* Global Spatial Data Infrastructure (GSDI), 2004.

[97] Natalia F. Noy. Semantic integration: A survey of ontology-based approaches. *SIGMOD Record*, 33(4):65–70, 2004.

[98] Timothy L. Nyerges. Schema integration analysis for the development of GIS databases. *International Journal of Geographical Information Systems*, 3(2):153–183, 1989.

[99] Autonomous Province of Trento. Provincial and municipal emergency protection guidelines. linee guida per le attivit di previsione, prevenzione, protezione e la pianificazione di protezione civile provinciale e comunale. Technical report, Autonomous Province of Trento, Italy, 2005.

[100] Municipality of Trento. Civilian emergency plan against the flooding hydrogeological risk of the river Adige - piano di protezione civile comunale contro il rischio idrogeologico di inondazione del fiume adige. Technical report, Municipality of Trento town, Italy, 2002.

[101] Open Geospatial Consortium. Web Map Service implementation specification. Technical report, Open Geospatial Consortium, 2002.

[102] Open Geospatial Consortium. OGC Reference Model. Technical report, Open Geospatial Consortium, 2003.

[103] Open Geospatial Consortium. Web Feature Service implementation specification. Technical report, Open Geospatial Consortium, 2005.

[104] Open Geospatial Consortium. OGC Best Practices Document: Gazetteer Service - Application Profile of the Web Feature Service Implementation Specification. Technical report, Open Geospatial Consortium, 2006.

[105] Open Geospatial Consortium. OpenGIS Sensor Web Enablement Architecture Document. Technical report, Open Geospatial Consortium, 2006.

[106] Open Geospatial Consortium. OpenGIS Catalogue Services Specification. Technical report, Open Geospatial Consortium, 2007.

[107] Open Geospatial Consortium. OpenGIS Geography Markup Language (GML) encoding standard. Technical report, Open Geospatial Consortium, 2007.

[108] Swapna Oundhakar, Kunal Verma, Kaarthik Sivashanmugam, Amit Sheth, and John Miller. Discovery of web services in a multi-ontology and federated registry environment. *International Journal of Web Services Research*, 2(3):1–32, 2005.

[109] Juan Pane, Carles Sierra, Gaia Trecarichi, Maurizio Marchese, Paolo Besana, and Fiona McNeill. Openknowledge deliverable 4.3: Summative report on gea, trust and reputation: integration and evaluation results. Technical report, OpenKnowledge, 2007.

[110] Mike P. Papazoglou. Service-oriented computing: Concepts, characteristics and directions. In *Proceedings of the 4th International Conference on Web Information Systems Engineering (WISE)*, pages 3–12, 2003.

[111] Christine Parent, Stefano Spaccapietra, and Esteban Zimanyi. *Conceptual modeling for traditional and spatio-temporal applications. The MADS approach*. Springer, 2006.

[112] Manoj Paul and S. K. Ghosh. An approach for service oriented discovery and retrieval of spatial data. In *Proceedings of the International Workshop on Service Oriented Software Engineering (IW-SOSE)*, pages 84–94, 2006.

[113] Charles Petrie, Tiziana Margaria, Ulrich Kuster, Holger Lausen, and Michal Zaremba. Sws challenge: Status, perspectives, and lessons learned so far. In *Proceedings of the 9th International Conference on Enterprise Information Systems (ICEIS)*, pages 447–452, 2007.

[114] Abbas Rajabifard, Ian P. Williamson, Peter Holl, and Glenn Johnstone. From local to global sdi initiatives: a pyramid building blocks. In *Proceedings of the 4th Global Spatial Data Infrastructure (GSDI) Conference*, pages 13–15, 2000.

[115] David Robertson. A lightweight coordination calculus for agent systems. *Declarative Agent Languages and Technologies*, pages 183–197, 2004.

[116] David Robertson, Fausto Giunchiglia, Frank van Harmelen, Maurizio Marchese, Marta Sabou, Marco Schorlemmer, Nigel Shadbolt, Ronnie Siebes, Carles Sierra, Chris Walton, Srinandan Dasmahapatra, Dave Dupplaw, Paul Lewis, Mikalai Yatskevich, Spyros Kotoulas,

Adrian Perreau de Pinninck, and Antonis Loizou. Open knowledge. coordinating knowledge sharing through peer to peer interaction. *Languages, Methodologies and Development Tools for Multi-Agent Systems*, 5118:1–18, 2008.

[117] Dumitru Roman, Uwe Keller Holger Lausen, Jos de Bruijn, Rubén Lara, Michael Stollberg, Alex Polleres, Dieter Fensel, and Christoph Bussler. Web service modeling ontology (WSMO). *Applied Ontology*, 1(1):77–106, 2005.

[118] Stefan Schulte, Julian Eckert, Nicolas Repp, and Ralf Steinmetz. An approach to evaluate and enhance the retrieval of semantic web services. In *Proceedings of the 5th International Conference on Service Systems and Service Management (ICSSSM)*, pages 237–243, 2008.

[119] Angela Schwering. *Semantic Similarity Measurement including Spatial Relations for Semantic Information Retrieval of Geo-Spatial Data*. PhD thesis, Vrije Universiteit, 2003.

[120] Amit P. Sheth. Changing focus on interoperability in information systems: from systems, syntax, structure to semantics. *Interoperating Geographic Information Systems*, 47:5–29, 1999.

[121] Amit P. Sheth, Cory Henson, and Satya Sahoo. Semantic sensor web. *Internet Computing, IEEE*, 12(4):78–83, 2008.

[122] Amit P. Sheth and James A. Larson. Federated database systems for managing distributed, heterogeneous, and autonomous databases. *ACM Computing Survey*, 22(3):183–236, 1990.

[123] Pavel Shvaiko. *Iterative Schema-based Semantic Matching*. PhD thesis, International Doctorate School in Information and Communication Technology (ICT), University of Trento, 2006.

[124] Pavel Shvaiko and Jérôme Euzenat. A survey of schema-based matching approaches. *Journal on Data Semantics*, IV:146–171, 2005.

[125] Ronny Siebes, Frank van Harmelen, Spyros Kotoulas, David Dupplaw, Dietlind Geldoff, Fausto Giunchiglia, Maurizio Marchese, Fiona McNeill, Andrian Perreau de Pinninck, David Robertson, Marta Sabou, Carles Sierra, Lucia Specia, Austin Tate, and Mikalai Yatskevich2. Deliverable 2.1b: The functional description of the openknowledge system. Technical report, OpenKnowledge, 2007.

[126] Barry Smith and David M. Mark. Geographical categories: An ontological investigation. *International Journal of Geographical Information Science*, 15:591–612, 2001.

[127] Paul C. Smits and Anders Friis-Christensen. Resource discovery in a european spatial data infrastructure. *IEEE Transactions on Knowledge and Data Engineering*, 19(1):85–95, 2007.

[128] Anastasiya Sotnykova, Christelle Vangenot, Nadine Cullot, Nacéra Bennacer, and Marie-Aude Aufaure. Semantic mappings in description logics for spatio-temporal database schema integration. *Journal on data semantics III*, 3534:143–167, 2005.

[129] Eleni Stroulia and Yiqiao Wang. Structural and semantic matching for assessing web-service similarity. *International Journal of Cooperative Information System*, 14(4):407–438, 2005.

[130] Heiner Stuckenschmidt. *Ontology-Based Information Sharing in Weakly Structured Environments.* PhD thesis, Vrije Universiteit, 2003.

[131] William Sunna and Isabel F. Cruz. Using the agreementmaker to align ontologies for the oaei campaign 2007. In *Proceedings of the*

Workshop on Ontology Matching (OM) at the 6th International Semantic Web Conference (ISWC), volume 304, 2007.

[132] Stefan Tai, Rania Khalaf, and Thomas A. Mikalsen. Composition of coordinated web services. In *Proceedings of the ACM/IFIP/USENIX International Middleware Conference*, pages 294–310, 2004.

[133] Vlad Tanasescu, Alessio Gugliotta, John Domingue, Rob Davies, Leticia Gutiérrez-Villarías, Mary Rowlatt, Marc Richardson, and Sandra Stinčić. A semantic web services gis based emergency management application. In *Proceedings of the Workshop at the 5th Semantic Web for eGovernment of International Semantic Web Conference (ISWC)*, pages 959–966, 2006.

[134] Gaia Trecarichi, Veronica Rizzi, Lorenzino Vaccari, Maurizio Marchese, and Paolo Besana. Openknowledge at work: exploring centralized and decentralized information gathering in emergency contexts. In *Submitted to the 6th International Conference on Information Systems for Crisis Response and Management (ISCRAM)*, 2009.

[135] Gaia Trecarichi, Veronica Rizzi, Lorenzino Vaccari, Juan Pane, and Maurizio Marchese. Openknowledge deliverable 6.8: Summative report on use of ok approach in eResponse: integration and evaluation results. Technical report, OpenKnowledge, 2008.

[136] Shengru Tu and Mahdi Abdelguerfi. Web services for geographic information systems. *IEEE Internet Computing*, 10(5):13–15, 2006.

[137] Lorenzino Vaccari, Maurizio Marchese, Fausto Giunchiglia, Fiona McNeill, Stephen Potter, and Austin Tate. Openknowledge deliverable 6.5: Emergency monitoring scenarios. Technical report, OpenKnowledge, 2006.

[138] Lorenzino Vaccari, Maurizio Marchese, and Alexander Ivanyuckovich. A service oriented approach for geographical data sharing. In *Proceedings of the Workshop 11th EC GI & GIS Workshop: European Spatial Data Infrastructure: Setting the framework*, pages 134–136, 2005.

[139] Lorenzino Vaccari, Maurizio Marchese, and Pavel Shvaiko. Openknowledge deliverable 6.6: Emergency response gis service cluster. Technical report, OpenKnowledge, 2007.

[140] Lorenzino Vaccari and Juan Pane. Ontology matching evaluation using gis web services. In *Submitted to the 1st CFP: Workshop on Matching and Meaning: automated development, evolution and interpretation of ontologies*, 2009.

[141] Lorenzino Vaccari, Juan Pane, Pavel Shvaiko, Maurizio Marchese, Fausto Giunchiglia, Paolo Besana, and Fiona McNeill. Openknowledge deliverable 3.7: Summative report on matching implementation and benchmarking results. Technical report, OpenKnowledge, 2008.

[142] Lorenzino Vaccari, Pavel Shvaiko, Paolo Besana, Maurizio Marchese, and Juan Pane. An evaluation of ontology matching in geo-service applications. *Submitted to the GeoInformatica*, 2009.

[143] Lorenzino Vaccari, Pavel Shvaiko, and Maurizio Marchese. An emergent semantics approach to semantic integration of geo-services and geo-metadata in spatial data infrastructures. In *Proceedings of the 10th Global Spatial Data Infrastructure (GSDI) Conference*, 2008.

[144] Lorenzino Vaccari, Pavel Shvaiko, and Maurizio Marchese. A geo-service semantic integration in spatial data infrastructures. *International Journal of Spatial Data Infrastructures Research (IJSDIR)*, 4(2009):24–51, 2009.

[145] Gabriel Valiente. *Algorithms on Trees and Graphs*. Springer, 2002.

[146] Kunal Verma, Kaarthik Sivashanmugam, Amit Sheth, Abhijit Patil, Swapna Oundhakar, and John Miller. Meteor-s wsdi: A scalable p2p infrastructure of registries for semantic publication and discovery of web services. *Information Technology and Management*, 6(1):17–39, 2005.

[147] Thomas Vögele and Christoph Schlieder. The use of spatial metadata for information retrieval in peer-to-peer networks. In *Proceedings of the 5th AGILE Conference*, pages 279–289, 2002.

[148] Holger Wache, Thomas Voegele, Ubbo Visser, Heiner Stuckenschmidt, Gerhard Schuster, Holger Neumann, and Sebastian Hübner. Ontology-based integration of information - a survey of existing approaches. In *Proceedings of the Workshop on Ontologies and Information Sharing at the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 108–117, 2001.

[149] Christopher D. Walton. *Agency and the Semantic Web*. Oxford University Press, 2007.

[150] Jia Wenjue, Gong Jianya, and Li Bin. Gis integration and interoperability based on gis service chain. In *Proceedings of the IEEE International Geoscience And Remote Sensing Symposium (IGARSS)*, pages 4962–4965, 2005.

[151] Michael Wooldridge. *An Introduction to MultiAgent Systems*. John Wiley & Sons Ltd, 2002.

[152] Michael F. Worboys and Misbah S. Deen. Semantic heterogeneity in distributed geographic databases. *SIGMOD Record*, 20(4):30–34, 1991.

[153] Sun Yanfeng, Ma Xiujun, Xie Kunqing, Chen Guanhua, Liu Chen, Li Chenyu, and Chen Zhuo. A compensation mechanism in gis web service composition. In *Proceedings of the IEEE International Geoscience And Remote Sensing Symposium (IGARSS)*, 2005.

[154] Song Yu, Bai Xue, Ju Shuchun, and Han Xiujuan. Building dynamic gis services based on peer-to-peer. In *Proceedings of the 1st International conference on Semantics, Knowledge and Grid*, page 68, 2005.

[155] Ilya Zaihrayeu. *Towards Peer-to-Peer Information Management Systems*. PhD thesis, International Doctorate School in Information and Communication Technology (ICT), University of Trento, 2006.

[156] Peisheng Zhao and Liping Di. Semantic web service based geospatial knowledge discovery. In *Proceedings of the IEEE International Geoscience And Remote Sensing Symposium (IGARSS)*, pages 3490–3493, 2005.