A Visual Semantic Search Framework for Finding Craft Services

Maximilien Kintz and Andrea Horch
Institute for Human Factors and Technology Management (IAT)
University of Stuttgart
Stuttgart, Germany
{maximilien.kintz, andrea.horch}@iat.uni-stuttgart.de

Abstract— **When a house is damaged, the house owner needs to quickly seek help from specialized craftsmen. Unfortunately, the complex structure of craft services and the lack of specialized knowledge make this task more difficult than it should be. To solve this problem, a standardized categorization of craftsmen and craft services has been developed in form of the German Crafts and Craftsmen Ontology (GeCCO). In this paper, an easy-to-use web-based tool for browsing and searching the ontology while taking advantage of all its semantic features is presented. Furthermore, a method for combining the ontology with a standard geographical web search tool is presented.**

Keywords-ontologies; graph visualization; Web search; craft services.

I. INTRODUCTION

Finding the nearest craftsman best suited to help you with damage in your house is, even in the time of quick and easy map-based Web searches, harder than it should be: domain specific knowledge such as "who can repair which damage?" or "how are crafts organized?" is often required from house owners, who usually are not craft services specialists at all.

To help users accomplish this task and in order to provide a structured way to find the appropriate service, we propose a visual semantic Web search framework based on an extensive self-created ontology: GeCCO - German Crafts and Craftsmen Ontology. The ontology is presented in detail in a paper published in 2012 (see [1] for more information).

The remainder of this paper is organized as follows: In Section 2, we discuss related work. In Section 3, we present requirements for a semantic search framework and assess the existing tools described in Section 2. In Section 4, we introduce a dynamic graphical user interface for browsing the craft services ontology. In Section 5, we describe the semantic Web search tool to find craft services. Finally, in the concluding Section 6, we present our methodology for evaluating the search tool, discuss limitations of the current implementation and propose ways to further develop the tool and possible future outcomes.

II. RELATED WORK

Related work and state of the art solutions are presented for ontology and graph visualization, semantic search and ontology search solutions, and finally for the description of technical and non-technical services.

A. Graphical Interface for Ontology and Graph Visualization

Many approaches to ontology visualization have been investigated and are today in use. Ontology viewers often already exist as part of ontology editors (examples of well-known visualization plug-ins for ontology editors being OntoGraf [4] for Protégé [5] or NavigOWL [6] for OntoStudio [7]). Katifori et al. have extensively investigated tools for ontology visualization [3]. More generally, ontologies can be viewed as graphs in which nodes represent classes and edges represent relations. Visualization techniques for large graphs have been investigated, e.g. in [8].

However, these tools tend to be targeted at specialists who understand the concepts of ontologies. In contrast, Web users seeking to achieve a simple common task, such as finding the appropriate craft service after damage, are often overstrained by the complexity of generic ontology viewers.

In a previous article, we introduced a tool for the visualization of a specific typo of graph: company business relations [15]. The software technologies used for the implementation of the graphical interface in the present work base on and largely extend this previous work.

B. Semantic Search

Semantic or ontology-based search engines have also already been widely investigated, e.g. by Lei et al. [17] who introduced SemSearch, a search engine allowing users without specific ontology knowledge to benefit from semantic Web technologies for their search queries. Natural language interfaces for semantic search engines have generally widely been investigated, another example being the ORAKEL interface [18]. Our work differs from these approaches in that it introduces a graphical component for ontology browsing and focuses on simple questions for the specific domain of craft services, such as "who can repair a

 ISBN: 978-1-61208-280-6

specific damage". The tool presented in this paper does not require any specific knowledge from the user, neither in semantic Web or ontologies nor in the domain of craft services.

C. Service Description and Matchmaking

A data interchange format is needed for the communication between the search engine in the background and the visualization interface. For this purpose, we use the Unified Service Description Language (USDL) introduced in [9]. USDL offers all required characteristics of service description (such as name and contact detail, capabilities or pricing models) and is better suited to describe generic real-world services, such as in our example craft services, than the possibly better known Web Service Description Language (WSDL) [2], tailored for the description of technical Web services. The limitations of WSDL for finding appropriate services and a solution for semantic service matchmaking were already investigated and presented by Paolucci et al. in [27].

III. Requirements for the Semantic Search Framework for Craft Services

Our goal is to provide users without any specific knowledge in semantic Web or craft services with a powerful user interface to easily find the appropriate craft service company that could help them with their damage. Furthermore, the results should be easily integrated in other software solutions, such as online service marketplaces (R5 in TABLE I). From this given general requirements, we derived five more specific requirements listed in TABLE I. The table also indicates how existing tools and frameworks discussed in the previous section fulfill these requirements.

TABLE I. Requirements for the search framework and existing solutions

Requirement	Existing solution			
	Grapjhical interfaces	***Sem-Search***	***ORAKEL***	***Paolucci et al.***
R1. No need for semantic web knowledge		X	X	
R2. No need for domain-specific knowledge		(X)	X	
R3. Navigation across domain classes and relations	X			
R4. High precision of search results	X	X	X	X
R5. Possibility to integrate in larger framework				X

Although existing solutions cover part of our requirements, a mash-up approach had to be followed to be able to cover all of them at once. In particular, it can be noted that many already existing frameworks provide a high precision of search results, which is a basic requirements for any specialized search tool (R4). However, in our use case and considering our target user-group (people searching for craftsmen and craft services), other requirements are more important. For example, navigation across classes and relations (to discover craft services related to other services or to specific objects, R3) or intuitive entry of query terms and reading of search results (implying that no need of specific knowledge in the specific domain of craft services (R2), or more generally of semantic Web (R1) is needed) were major design criteria. The following sections present the hybrid approach we used to try and cover all requirements.

IV. GeCCO Ontology Browser

In this section, we briefly describe the German Crafts and Craftsmen Ontology (GeCCO) and introduce a tool for visually browsing and searching the ontology to find the appropriate craft services.

A. The GeCCO Craft Services Ontology

Current standardized crafts categorization systems (such as the German Crafts Code or *Handwerksordnung* [10]), do neither include all needed logical connections between crafts and craftsmen nor the synonyms for a sufficient use in a search system. Current craftsmen search systems (such as MyHammer [11]) have their own categorizations of crafts and craftsmen, with the same limitations as the standardized systems.

GeCCO (German Crafts & Craftsmen Ontology) is a new ontology developed as part of the research project openXchange (information about the project can be found online [12] and in print [13]) and introduced in [1]. GeCCO creates an extensible class hierarchy and defines synonyms and all needed logical connections between the classes.

B. Browsing the Ontology with a Dynamic User Interface

Potential users of a craft services search tool do not necessarily have specific knowledge in the classification of craft services. As was already explained in [1], the goal of the GeCCO ontology was to help users navigate through craft services and related objects. Thus, users of a craft services search tools need to be offered a dedicated simple browsing and searching interface.

The Prefuse Flare [14] Flex framework was chosen, as its extended configurability and high performance had given good results in previous own work concentrating on the visualization of company relations [15]. However, to meet the specific needs of the craft services ontology visualization, a tool supporting more advanced functionalities needed to be implemented.

The first important improvement is the integration of a custom built OWL [26] parser that reads the ontology structure as stored in an OWL XML file and creates the graph structure needed by the Prefuse framework. We couldn't identify any already existing OWL parser for use with the Flex technology, which led to the implementation of this custom-built parser capable of correctly extracting classes and relations from an OWL file describing an ontology.

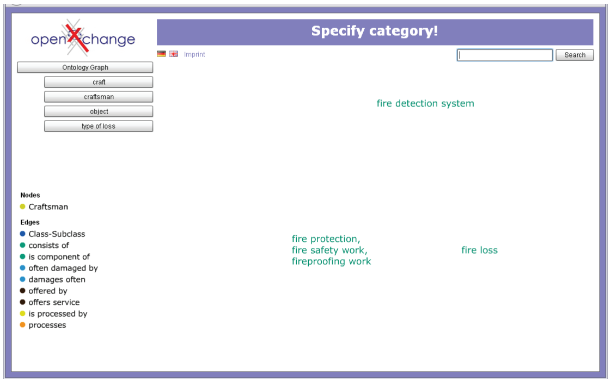

Figure 1. Example search results for a search for "fire"

The tool supports all languages defined in the ontology (in the present case, German and English) by taking advantage of the differentiation introduced in the ontology between unique class names and possibly multiple and language-dependent labels. Synonyms are visually presented, since it is typical for craft services to be known under different names in different regions of Germany (a typical example being Flaschner and Klempner, both referring to a plumber). Furthermore, craft services which used to be distinct services but have merged over the years are also indicated as such.

The browsing starts with an overview of the ontology. By zooming, clicking and panning, the user can navigate to a specific part of the ontology.

Using the search textbox provided on the upper right side, the user can perform a simple text-based search. In this case, the search tool queries the ontology and returns appropriate results as follow:

- if the user entered a word that does not match to a unique class, a circular presentation of all matching classes is returned (Figure 1), letting the user chose the appropriate one or enter a new search query,
- if the user entered the name of a known ontology class, the detailed view of this class is shown (Figure 2).

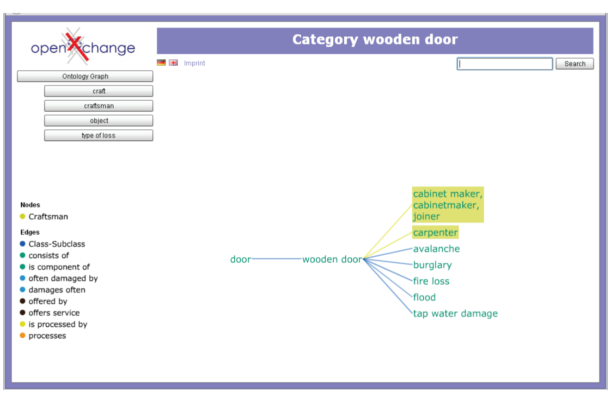

Figure 2. Detailed view of the category "wooden door"

The detailed view of a category of the ontology indicates all related upper-classes and sub-classes, linked by edges colored according to the relation they represent. The key for the color-coding is presented on the left side of the browser window. For example, on Figure 2, blue edges indicate the "often damaged by" relation as well as its inverse "damages often". Classes are represented with their labels. To allow a faster and simpler navigation and differentiation from damaged objects, classes representing craft services are highlighted.

By double-clicking on the name of a craft service, the user can choose between showing the graph for this service or being redirected to a Web-based craft service search tool (see Section 5), so as to find the contact data of the nearest office.

V. Craft Services Web Search Tool

The GeCCO ontology browser helps users identify the appropriate craft service needed to help them. This identification is the starting point for searching a specific craft service company, for example the one that is located nearest to the user. To that end, a dedicated Web-based search interface for a semantic Web service was built and linked to the ontology browser.

A. Graphical Interface for Ontology and Graph Visualization

The craft services Web search front end consists in a single text box, in which a user can enter his search query.

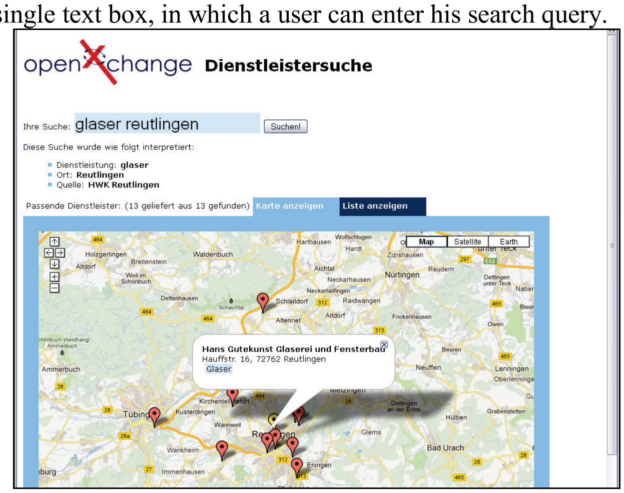

Figure 3. Map-based search result presentation for a search for Glaser (glazier) near the city of Retulingen

The user does not have to use a complex search form to perform advanced search queries: The tool recognizes which parts of the search queries need to be matched with which possible search criteria. Classes defined in the ontology (crafts and goods), using GeCCO as a reference, as well as German cities, using a geocoding service as reference (in the current implementation case, the Google Maps geocoding API was used) can automatically be recognized. Thus, a search for a glazier near the city of Reutlingen (cf. Figure 3) can be entered as "glazier reutlingen" and, using semantic

 ISBN: 978-1-61208-280-6

search capabilities, will be correctly interpreted by the tool as the combination {glazier: craft service, Reutlingen: city}.

The results are retrieved from the Web service and presented to the user in two possible manners.

The first presentation is a Map-based one (also shown in Figure 3). The reference point for the search is indicated by a yellow bubble, the matching craft services by red bubbles. By clicking on a bubble exposes detailed information on the respective company, such as name, available services and contact information. This view is best suited for geographic searching and selection of the nearest service provider.

Figure 4. List-based search result presentation for a search for a Schreiner (carpenter) named Beck

The second presentation is a list-based view of the results (see Figure 4). The same information is presented as a conventional text list allowing for rapid scanning and exporting matching data lists.

B. *Search Tool Architecture*

The architecture of the craft services visual search framework (see Figure 5) can be categorized in three main parts:

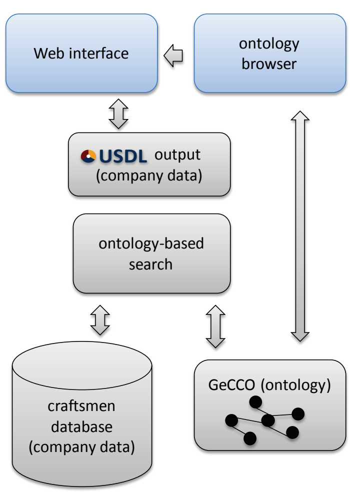

Figure 5. Architecture of the visual semantic craft services search framework

- a repository consisting in the ontology as an OWL XML file and a MySQL [24] database containing craft services company data (addresses, names, etc.),
- a search SOAP Web service, and
- a user interface (presented in sections 4.B and 5.A) consisting of the ontology browser and of the Web-based search tool.

The used technologies are established open-source software components.

```
<?xml version="1.0" encoding="ASCII"?>
<usdl3:ServiceDescription xmi:version="2.0"
xmlns:xmi="http://www.omg.org/XMI"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:foundation="http://internet-ofservices.
com/usdl/foundation/20100416-M4"
xmlns:participantmodule="http://internet-ofservices.
com/usdl/modules/participants/20100416-M4"
xmlns:pricingmodule="http://internet-ofservices.
com/usdl/modules/pricing/20100416-M4"
xmlns:servicelevelmodule="http://internet-ofservices.
com/usdl/modules/servicelevel/20100416-M4"
xmlns:usdl3="http://internet-ofservices.
com/usdl/20100416-M4">
<Guid>d11c40be-5df9-4a02-8e32-ea78323c0f55</Guid>
<Service xmi:id="ServiceWrapper_8732241">
<ServiceElements xmi:id="Service_29245520">
<Guid>bdcb0674-9d90-479f-8fb7-92b6e6142fe0</Guid>
<Version>1.0</Version>
<Name>Name Handwerksunternehmen</Name>
<PublicationTime xsi:type="foundation:AbsolutePointInTime"
xmi:id="AbsolutePointInTime_32657640">
<TimeZone>Europe/Berlin</TimeZone>
<Value>2010-09-17T10:57:25.616+0200</Value>
</PublicationTime>
<Nature>Human</Nature>
<Certifications>
…
```

Figure 6. Excerpt from the XML serialization of the USDL description of a generic craft service

The search Web-front-end consists of a simple Java servlet-based Web application running on an Apache Tomcat [25] server. The Web application is entirely focused on user interface generation and only passes queries to the Web service and presents results in HTML. No interpretation or search happens at that level: all search related activities are performed by the Web service.

The Web application and Web service exchange queries and results using a simple self-specified XML format that is encapsulated in SOAP messages. In the search results messages, the actual service descriptions (i.e. name, address, associated crafts and related information) are encoded using the XML serialization of USDL (see Figure 6. for an example).

The use of the self-defined XML language is restricted to metadata such as the total number of results found, or the way in which the search query was interpreted. USDL was used since it provides a complete description language for services, from basic information such as contact data, up to more complex descriptions of capabilities or pricing models for these capabilities. Thus, the nature of the information being transmitted by the service could be vastly enriched without changing anything to the search service architecture.

To facilitate the integration of the search service in an online service marketplace regrouping other similar or complementary services, the interface, search parameters, and result formats were also described using the USDL language.

The ontology-based search service uses a simple architecture. For the development of the SOAP interface, the Apache Axis2 [23] framework was used, as it allows a very simple integration with standard Java applications. The search engine interprets the search query (i.e. if the ingoing query was "glazier Reutlingen", in the interpreted query "Reutlingen" is mapped to "city" and "glazier" to "craftsman) to build an SQL query and retrieve the results.

The algorithm used to interpret the query works as follows:

1) The query is split in individual tokens
2) Each token is mapped to an ontology class, i.e. a possible craft service category, damaged object, etc. If a token could successfully be mapped to an ontology class, it is removed from the list. If necessary, stemming of fuzzy mapping of keywords and class could be implemented (this was considered not useful for our test scenarios).
3) Non-matched tokens are sent to the Google Maps API [22] geocoding service. If they could successfully be mapped to German cities, they are considered as geographical restriction and removed from the list.
4) Remaining tokens are considered as possible names of craft services providers.

This algorithm gave satisfying results with all basic test examples, with some obvious limitations such as synonyms or multi-word city names (see Section 5.2 for possible workarounds and improvements).

VI. Discussion

In this final section, fulfilled requirements, the evaluation phase with several partners, achieved results, known limitations and possible plans for future work are presented.

A. Evaluation

The GeCCO (German Crafts and Craftsmen Ontology) as well as the tool were evaluated in cooperation with the Chamber of Industry and Commerce of the Stuttgart Region [19] and several regional Chambers of crafts (Handwerkskammer, i.e. organizations that federate craftsmen in a local area), in particular the Handwerkskammer Reutlingen [20], which provided a large test database containing over 10.000 craft service names and contact data.

These prerequisites allowed evaluating the suitability of the tool's design principle, the combination with the ontology and the scalability of the architecture with a dataset of a reasonable size. The evaluation partners were very satisfied with the new search possibilities offered by the tool, and discussions for the realization of productive implementations of the tool are currently in progress.

By using well established graph visualization techniques, it was possible to build a tool allowing non-specialized users to take advantage of a powerful craft services ontology to perform efficient semantic search activities without knowledge in semantic Web technologies nor specific knowledge in craft services.

The semantic search framework presented integrates an ontology navigator with a more classical text-based search, and allows the users to take advantage of both traditional search techniques and of graph navigation, so as to easily find the nearest craftsmen as well as get an overview of the crafts men professional environment.

Since the implementation makes advantage of USDL, the development activities and test results supported the work of the USDL W3C incubator group [16], to help improving and standardizing USDL. The use of USDL also guarantees the possibility to integrate the results in another tool, such as a service marketplace.

TABLE II. Fulfillment of the Requirements

Requirement	Fulfilled	Comment
R1. No need for semantic web knowledge	Yes	Simple keyword-based queries
R2. No need for domain-specific knowledge	Yes	Users do not need to know about craft services since damaged objects are also integrated in the ontology
R3. Navigation across domain classes and relations	Yes	Graphical ontology browser, integration of synonyms in the ontology
R4. High precision of search results	Yes	Exact matching with ontology class
R5. Possibility to integrate in larger framework	Yes	Use of USDL for craft services descriptions

TABLE II summarizes the requirements defined in Section 3 and how they were fulfilled with the search framework we introduced.

B. Limitations

The most obvious limitation of the current implementation lies in the limited test data available, limited to the area of the city of Reutlingen. Although this does not impact the validity of the overall approach, including larger data sets is an important improvement factor.

Another improvement could be moving away from a Flash based framework, so as to allow the usage of the tool on the more and more common platforms which do not support Flash, such as current tablets or smartphones. The Protovis Framework [21] offers similar functionality with full HTML-based technologies and would possibly provide a satisfying replacement for Prefuse Flare.

Finally, concerning the Web search tool and the one-box search input currently used, limitations come from the word-based way used by the software to distinguish places, objects, craft services or craft service company names. For example, if the user enters "Bad Cannstatt Glaser", the tool will tend to match "Bad" to the object "Bath (tub)" and "Glaser" to the service "Glazier", leaving Cannstatt as a potential place or name. The correct matching would have been "Bad Cannstatt" as a place (a suburb of Stuttgart, Germany). To solve this problem, a possible solution would be to iteratively try to match unique words, then two or three-word combinations, and then choose for each matched item the most probable one. A complete algorithm for matching text-based queries to ontology classes and relations has been introduced in [17]. These approaches would however require, especially for geocoding, a solution offering a very high performance as many slightly different queries would be sent for each user request.

C. Future Work

The interest and benefits of a search tool are highly linked to the quality and completeness of the data that can be found by using it. Thus, one of the main goals of the further development of the search framework is the inclusion of larger datasets, probably by cooperating with additional Chambers of Crafts, so as to achieve a much higher recall over larger regions of Germany. Performance was not an issue in tests with a small user group and a moderate dataset. The search framework presented in this paper can be adapted to scale correctly, using a distributed architecture or simply with more powerful hardware.

Furthermore, including more data in the background could help further develop GeCCO, e.g., with regards to the lists of damaged objects which currently is limited to the most common objects concerned by property claims management.

Reasoners could be used to better search the ontology. Several reasoners were investigated and used to assess the integrity of the GeCCO onotology (as mentioned in [1]), but they could also be used at search time for complex queries.

In order to complement the already-performed evaluation with professional users and a limited number of non-specialists, a larger evaluation and test phase would help guarantee the effectiveness of our approach.

Finally, it can be noted that an advantage of our solution is that it can easily be applied to other domains than craft services, provided that an ontology describing the domain and appropriate data can be used or created.

Acknowledgment

The work published in this article was partially funded by the openXchange project of the German Federal Ministry of Economy and Technology under the promotional reference 01MQ09011.

References

[1] A. Horch and M. Kintz, GeCCO: German Crafts and Craftsmen Ontology – A Common Crafts Ontology. In: Proceedings of the 8th International Conference on Web Information Systems and Technologies (WEBIST). Porto, Portugal, pp. 355-360 (2012)

[2] E. Christensen, F. Curbera, G. Meredith, and S. Weerawarana, Web Services Description Language (WSDL) 1.1. (2011) Available online under http://www.w3.org/TR/wsdl (retrieved March 2013)

[3] A. Katifori, C. Halatsis, G. Lepouras, C. Vassilakis, and E. Giannopoulou, Ontology visualization methods - A survey. ACM Comput. Surv. 39, 4, Article 10 (October 2007)

[4] OntoGraf, http://protegewiki.stanford.edu/wiki/OntoGraf (retrieved March 2013)

[5] Protégé, http://protege.stanford.edu/ (retrieved March 2013)

[6] NavigOWL, http://protegewiki.stanford.edu/wiki/NavigOWL (retrieved March 2013)

[7] OntoStudio, http://www.ontoprise.de/en/products/ontostudio/ (retrieved March 2013)

[8] T. von Landesberger, A. Kuijper, T. Schreck, J. Kohlhammer, J.J. van Wijk, J.J., J.-D. Fekete, and D.W. Fellner, Visual Analysis of Large Graphs. 12th Joint Eurographics/IEEE-VGTC Symposium on Visualization, pp. 1719-1749 (2010)

[9] J. Cardoso, A. Barros, N. May, and U. Kylau, Towards a Unified Service Description Language for the Internet of Services: Requirements and First Developments. In: Proceedings of the 2010 IEEE International Conference on Services Computing, Miami, Florida, pp. 602-609 (2010)

[10] German Crafts Code (*Handwerksordnung*), http://www.gesetze-im-internet.de/hwo/ (retrieved March 2013)

[11] MyHammer, http://www.myhammer.com (retrieved March 2013)

[12] Project openXchange, http://www.openxchange-projekt.de (retrieved March 2013)

[13] A. Horch, M. Kintz, F. Koetter, T. Renner, M. Weidmann, and C. Ziegler, openXchange: Servicenetzwerk zur effizienten Abwicklung und Optimierung von Regulierungsprozessen bei Sachschaeden. Fraunhofer Verlag, Stuttgart (2012)

[14] Prefuse Flare Framework, http://flare.prefuse.org/ (retrieved March 2013)

[15] M. Kintz and J. Finzen, A Simple Method for Mining and Visualizing Company Relations Based on Web Sources. In: Proceedings of the 7th International Conference on Web Information Systems and Technologies (WEBIST), Noordwijkerhout, pp. 597-602 (2011)

[16] K. Kadner and D. Oberle (Editors), Unified Service Description Language XG Final Report. Available online under http://www.w3.org/2005/Incubator/usdl/XGR-usdl-20111027/ (retrieved March 2013). (2011)

[17] Y. Lei, V. Uren, and E. Motta, SemSearch: A Search Engine for the Semantic Web. Proc. 5th International Conference on Knowledge Engineering and Knowledge Management Managing Knowledge in a World of Networks, Lect. Notes in Comp. Sci., Springer, Podebrady, Czech Republic, pp. 238-245 (2006)

[18] P. Cimiano, ORAKEL: A Natural Language Interface to an F-Logic Knowledge Base. Proceedings of the 9th International Conference on Applications of Natural Language to Information Systems, pp. 401-406 (2004)

[19] IHK Region Stuttgart, http://www.stuttgart.ihk24.de/ (retrieved March 2013)

[20] Handwerkskammer Reutlingen, http://www.hwk-reutlingen.de/ (retrieved March 2013)

[21] Protovis, http://mbostock.github.com/protovis/ (retrieved March 2013)

[22] Google Maps API, https://developers.google.com/maps/ (retrieved March 2013)

[23] Apache Axis2, http://axis.apache.org/axis2/java/core/ (retrieved March 2013)

[24] MySQL, http://www.mysql.com/ (retrieved March 2013)

[25] Apache Tomcat, http://tomcat.apache.org/ (retrieved March 2013)

[26] OWL Web Ontology Language, http://www.w3.org/TR/owl-features/ (retrieved March 2013)

[27] M. Paolucci, T. Kawamura, T.R. Payne, and K.P. Sycara, Semantic Matching of Web Services Capabilities. In: Proceedings of the First International Semantic Web Conference on The Semantic Web (ISWC '02), Ian Horrocks and James A. Hendler (Eds.). Springer-Verlag, London, UK, UK, pp. 333-347 (2002)