A Deductive Semantic Brokering System

Grigoris Antoniou[1], Thomas Skylogiannis[1], Antonis Bikakis[1], Nick Bassiliades[2]

[1] Computer Science Department, University of Crete, Greece
Institute of Computer Science, FORTH, Greece
{antoniou, dogjohn, bikakis}@ics.forth.gr
[2] Department of Informatics, Aristotle University of Thessaloniki, Greece
nbassili@csd.auth.gr

Abstract. In this paper we study the brokering and matchmaking problem in the tourism domain, that is, how a requester's requirements and preferences can be matched against a set of offerings collected by a broker. The proposed solution uses the Semantic Web standard of RDF to represent the offerings, and a deductive logical language for expressing the requirements and preferences. We motivate and explain the approach we propose, and report on a prototypical implementation exhibiting the described functionality in a multi-agent environment.

1 Introduction

E-Commerce describes the revolution that is currently transforming the way business is conducted through the use of information technology, and in particular the World Wide Web. The 2nd generation of e-Commerce will be realized through the use of automated methods of information technology. Web users will be represented by *software agents*. According to [8], there is an increasing use of software agents for all the aspects of e-Commerce.

As software agents start to engage in e-commerce, new issues arise. Information must be organized in a way that is accessible by both humans and machines. Additionally, machines must be able to access, process and interpret the information in the same way. This vision is consistent with the Semantic Web initiative 4], which enriches the current Web through the use of machine-processable information about the meaning (semantics) of information content. This way, the meaning of displayed information is accessible not only to humans, but also to software agents.

The focus of the present work is on semantic-based electronic markets. Such markets help both service providers and requesters to match their interests. The key operations in such markets are to: (a) Identify appropriate services that satisfy user requirements; and (b) Select the best service based on the user's preferences.

How to address these questions using Semantic Web technology is the main focus of the present work. Previous works related to our approach include [10, 13, 7, 11 and 5]. Our work distinguishes itself from previous efforts in a novel combination of standard Semantic Web technology with a nonmonotonic rules language that allows one to express requirements and preferences.

2 Proposed Solution

2.1 General Approach

The three basic roles that we identify in our brokering system are the *Service Requester*, the *Service Provider*, and the *Broker*. Directory Facilitator is a service, which agents use to find each other and register the protocols and the ontologies that they use. The technical solution we provide is based on the following key ideas:

- Service requesters, service providers and brokers are represented by software agents that run on the JADE multi-agent platform.
- The requirements of the service requester are represented in defeasible logic, using rules and priorities. These requirements include both indispensable requirements that must be met for a service to be acceptable, and soft requirements (preferences) that can be used to select among the potentially acceptable offerings.
- The offerings or advertisements are represented in a certain semi-structured format using the Semantic Web standard language RDF for describing Web resources.
- The terminology shared by providers, requesters and brokers is organized in an ontology using the Semantic Web standard of RDF Schema.
- The broker is also a software agent and has special knowledge both for the declarative language and the advertisement format. It also has the ability to perform semantic checks to the information it receives.
- When the broker receives a request it matches the request to the advertisements by running the request specification against the available offerings, making use of information provided by the shared ontology, as required. Then the requester's preferences are applied to select the most suitable offering(s).
- For the persistent storage of advertisements, an RDF repository, and particularly ICS-FORTH RDF Suite [1], is used.

2.2 Description of Offerings

The offerings are described in RDF, the standard Semantic Web language for representing factual statements. This choice (a) supports interoperability among agents and applications, and (b) facilitates the easy publication, collection and combination in decentralized dynamic settings. The offerings are enriched through reference to shared ontologies, e.g. of the tourism domain or geographical terms. We assume that these ontologies are expressed in RDF Schema. We have chosen this language over the use of OWL because at present it is not clear how the deductive capabilities of OWL and rule systems can be combined. We could certainly use most features of OWL Lite, given that they can be expressed using rules 6].

2.3 Description of Requests and Preferences

We have chosen defeasible logic to represent requesters' requirements and preferences because it satisfies the above criteria. In particular,

- It is a formal language with well-understood meaning, thus it is also predictable and explainable.
- It is designed to be executable; implementations are described in [9]. It is also scalable, as demonstrated in the same paper, where it is shown that 100,000 rules can be processed efficiently, due to its low computational complexity.
- It is expressive, as demonstrated by the use of rules in various areas of information technology. In fact, among the logical systems, it is rule-based systems that have been best integrated in mainstream IT.
- Finally, it is suitable for expressing requirements and preferences in our setting. This is so, because it supports the natural representation of important features: (a) Rules with exceptions are a useful feature in our problem. For example, a general rule may specify acceptable offerings, while more specific rules may describe cases in which the general rule should not apply and the offering should not be accepted. (b) Priorities are an integral part of defeasible logic, and are useful for expressing user preferences for selecting the most appropriate offerings from the set of the acceptable offerings.

The expressing capabilities of defeasible logic are exhibited in the following example: "Antonis has the following preferences about his holiday travel package: He wants to depart from Athens and considers that the hotel at the place of vacation must offer breakfast. In addition, he would like either the existence of a swimming pool at the hotel to relax all the day, or a car equipped with A/C, to make daily excursions at the island. However, Antonis believes that if there is no parking area at the hotel, the car is useless, because he adds to him extra effort and fatigue. Lastly, if the tickets for his transportation to the island are not included in the travel package, he is not willing to accept it…". This verbal description of Antonis' hard requirements about acceptable offers can be modeled through the following defeasible logic rules:

$\boldsymbol{r_1}$: *from(X,athens),includesResort(X,Y),breakfast(Y,true)* $\Rightarrow$ *accept(X)*
$\boldsymbol{r_2}$: *from(X,athens),includesResort(X,Y),swimmingPool(Y,true)* $\Rightarrow$ *accept(X)*
$\boldsymbol{r_3}$: *from(X,athens), includesService(X,Z), hasVehicle(Z,W), vehicleAC(W,true)* $\Rightarrow$ *accept(X)*
$\boldsymbol{r_4}$: *includesResort(X,Y),parking(Y,false)* $\Rightarrow$ *~accept(X)*
$\boldsymbol{r_5}$: *~includesTransportation(X,Z)* $\Rightarrow$ *~accept(X)*

$r_4 > r_3,\ r_1 > r_4,\ r_2 > r_4,\ r_5 > r_1,\ r_5 > r_2,\ r_5 > r_3$

3 Brokering System Implementation

3.1 Multi-Agent Framework

The agent framework we used for the development of our system is JADE (jade.cselt.it). JADE is an open-source middleware for the development of distributed multi-agent applications. It is Java-based and compliant with the FIPA specifications (www.fipa.org). It provides libraries for agent discovery, communication and interaction, based on FIPA standards.

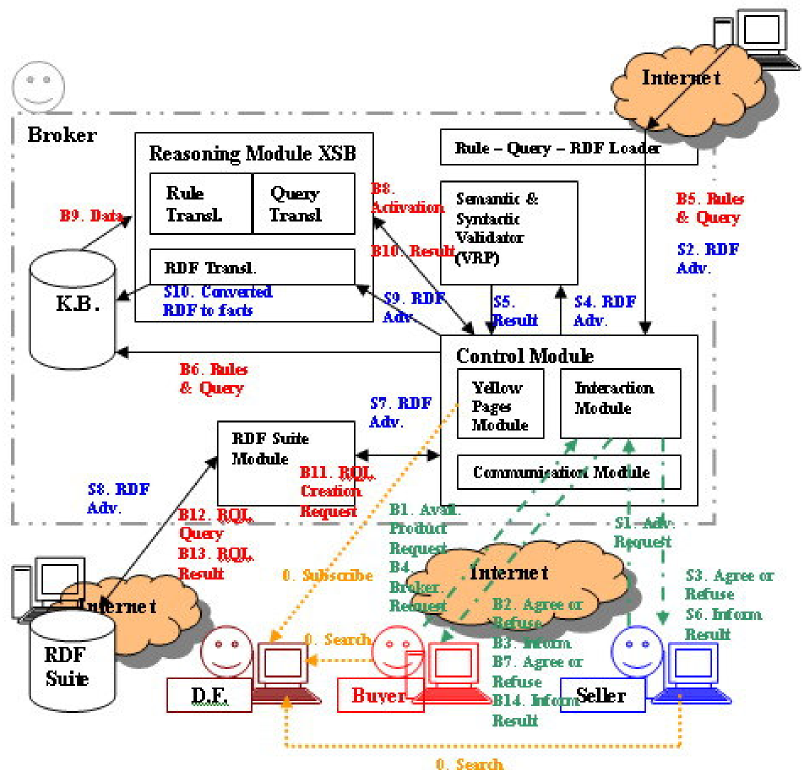

Fig. 1. The Brokering System Architecture.

From the functional point of view, JADE provides the basic services necessary to distributed peer-to-peer applications in the fixed and mobile environment. JADE allows each agent to dynamically discover other agents and communicate with them according to the peer-to-peer paradigm. From the application point of view, each agent is identified by a unique name and provides a set of services. It can register and modify its services and/or search for agents providing given services, it can control its life cycle and, in particular, communicate with all other peers.

3.2 System Architecture and Modules

The architecture (Fig. 1) of the broker consists of five main modules: (a) reasoning module, (b) control module, (c) semantic and syntactic validator, (d) RDF Suite module, and (e) rule-query-RDF loader module. Reasoning and control modules consist of other sub-modules as one can see in Fig.1 which depicts the overall system architecture. The other three modules are stand-alone. Finally, the control module is responsible for the coordination of all the other modules.

RDF Translator: The role of the RDF translator is to transform the RDF statements into logical facts, and the RDFS statements into logical facts and rules. This transformation allows the RDF/S information to be processed by the rules provided by the Service Requester. The RDF data are transformed into logical facts of the form: *Predicate(Subject,Object)*. To capture the semantics of RDF Schema constructs, we create the following rules.

a*: C(X):- rdf:type(X,C).*

b*: C(X):- rdfs:subClassOf(Sc,C),Sc(X).*

c*: P(X,Y):- rdfs:subPropertyOf(Sp,P),Sp(X,Y).*

d*: D(X):- rdfs:domain(P,D),P(X,Z).*

e*: R(Z):- rdfs:range(P,R),P(X,Z).*

All the above rules are created at compile-time, i.e. before the actual querying takes place. Therefore, although the above rules at first sight seem second-order because they contain variables in place of predicate names, they are actually first-order rules, i.e. predicate names are constant at run-time.

Semantic-Syntactic Validator: This module is an embedded version of VRP[14], a parser for validating RDF descriptions. Upon receipt of an advertisement, its RDF description is checked by this module. Among others, the tests performed are: class hierarchy loops, property hierarchy loops, domain/range of subproperties, source/target resources of properties and types of resources.

Interaction and Communication Modules: The communication module is responsible for sensing the network and notifying the control module when an external event occurs. In order to decide the course of action based on the incoming message's type, the broker agent extracts the message from the queue and examines its type, i.e. whether it is a "Broker Request", "Advertise Request" etc. Accordingly it activates the interaction module. The interaction module consists of different interaction protocols that extend the standard FIPA Request interaction protocol.

RDF Suite Module: The RDF Suite module is responsible for all the actions related with the handling of the advertisements and the domain ontology. The most important functions of this module are:

- Initial upload of RDFS ontology and RDF instances into the RDF repository.
- Update of the RDF repository.
- Preparation of RQL queries and forwarding to the RDF Suite.
- Receipt of RQL queries' results.

Rule-Query-RDF Loader. The role of this module is to download the files, which contain the rules and the query of the user, in defeasible logic format. It also downloads the service providers' advertisements (RDF descriptions). Finally, it implements methods for file handling. For the implementation of this module we used the API of Java for File Management and the API for Networking.

Reasoning Module: The role of the Reasoning Module is to apply the queries to files, which contain the facts and the rules, and to evaluate the answer. When the Service Requester makes a query, the Reasoning Module compiles the files containing the facts and the rules, and applies the query to the compiled files. The answer is sent to the Control Module. XSB Prolog is used as the compiler and reasoning module for our system. We made this choice, as XSB supports the well-founded semantics of logic programs through the use of tabled predicates, and its sk_not negation operator.

Rule Parser & Translator: The Rule Parser is responsible for checking the validity of the defeasible rules, which are submitted by the Service Requester. The rules are considered to be valid, if they follow the standard syntax of defeasible logic, as described in [2]. If they are valid, the theories are passed to the Logic Translator.

The Rule Translator is responsible for transforming the rules submitted by the Service Requester using into Prolog rules that emulate the semantics of defeasible logic. The method for this translation is described in detail in [3].

Query Translator: Its role is to translate the users' queries into valid Prolog queries, that can be applied to the files representing the rules and the facts.

4 Conclusions and Future Work

In this paper we studied the brokering and matchmaking problem in the tourism domain, that is, how a requester's requirements and preferences can be matched against a set of offerings collected by a broker. The proposed solution uses the Semantic Web standard of RDF to represent the offerings, and a deductive logical language for expressing the requirements and preferences. We motivated and explained the approach we propose, and reported on a prototypical implementation exhibiting the described functionality in a multi-agent environment.

Our approach has obvious *advantages* compared to other approaches. Particularly, (a) we do not provide a fixed algorithm for brokering but it is the user who specifies the algorithm on the basis of its preferences. (b) The architecture we provide is highly reusable. The system can be applied in any domain only with the addition of a new ontology and new rules which capture the preferences. (c) Using JADE, we exploit the advantages of peer-to-peer systems (i.e. travel agencies and broker as peers) and also make use of FIPA standards for agent communication and discovery. (d) We use a highly expressive language for preferences specification with interesting features, such as conflicting rules and priorities of rules. (e) We use RDF for the expression of advertisements. This choice supports interoperability among agents and applications and facilitates the easy publication, collection and combination in decentralized dynamic settings. (f) We allow for permanent storing of advertisements with the use of the RDF Suite repository. The main *drawbacks* of the current implementation are: (a) The advertisements cannot be removed automatically when they expire. (b) The syntax of the defeasible logic may appear too complex for many users, and should be supported by, possibly graphical, authoring tools.

In the future we intend to extend the described work in various directions: (i) Add advertisement removal functionality; (ii) Implement graphical user interfaces for the integrated system. (iii) Integrate the current brokering system with the negotiation system proposed in [12]. In our current implementation, a service requester agent is able to find potential products or services and potential service providers. We intend to extend our system to support direct involvement of the service requester in negotiation with the service provider for the resulted product, as soon as the brokering stage has been completed. Finally, as a long-term goal we intend to extend the semantic brokering approach presented in this paper to brokering for general purpose semantic web services, providing matchmaking between Web Service advertisements and requests described in OWL-S.

References

1. S. Alexaki, V. Christophides, G. Karvounarakis, D. Plexousakis and K. Tolle (2001). The ICS-FORTH RDFSuite: Managing Voluminous RDF Description Bases. In *Proc. 2nd International Workshop on the Semantic Web*, Hongkong, May 1, 2001.
2. G. Antoniou, D. Billington, G. Governatori and M.J. Maher (2001). Representation results for defeasible logic. *ACM Transactions on Computational Logic* 2, 2 (2001): 255 – 287.
3. G. Antoniou, A. Bikakis (2004). “A System for Non-Monotonic Rules on the Web». Workshop on Rules and Rule Markup Languages for the Semantic Web (RuleML 2004), G. Antoniou, H. Boley (Ed.), Springer-Verlag, Hiroshima, Japan, 8 Nov. 2004.
4. T. Berners-Lee (1999). *Weaving the Web*. Harper 1999.
5. Y-C. Chen, W-T. Hsu, P-H. Hung (2003). Towards Web Automation by Integrating Semantic Web and Web Services. In *Proc. 12th International WWW Conference.*
6. B. N. Grosof, I. Horrocks, R. Volz and S. Decker (2003). Description Logic Programs: Combining Logic Programs with Description Logic". In: *Proc. 12th Intl. Conf. on the World Wide Web (WWW-2003)*, ACM Press.
7. L. Li and I. Horrocks (2003). A Software Framework For Matchmaking Based on Semantic Web Technology. In *Proc. 12th International Conference on WWW*, ACM 2003.
8. P. Maes, R.H. Guttman and A.G. Moukas (1999). Agents That Buy and Sell. *Communications of the ACM* Vol. 42, No. 3 March 1999.
9. M.J. Maher, A. Rock, G. Antoniou, D. Billington and T. Miller (2001). Efficient Defeasible Reasoning Systems. *International Journal of Tools with Artificial Intelligence* 10,4 (2001): 483—501.
10. M. Nodine, W. Bohrer, A. Hee Hiong Ngu (1998). Semantic Brokering over Dynamic Heterogeneous Data Sources in InfoSleuth. In *Proc. 15th International Conference on Data Engineering*, IEEE Computer Society.
11. M. Paolucci, T. Kawamura, T.R. Payne, K. Sycara (2002). Semantic Matching of Web Services Capabilities. In *Proc. 1st International Semantic Web Conference (ISWC-2002).*
12. T. Skylogiannis, G. Antoniou, N. Bassiliades, G. Governatori (2005) "DR-NEGOTIATE-A System for Automated Agent Negotiation with Defeasible Logic-Based Strategies ". In proceedings of the IEEE international conference on e-Technology,e-Commerce and e-Service (EEE05). Hong Kong, China April 2005.
13. D. Trastour, C. Bartolini, J. Gonzalez- Castillo (2001). A Semantic Web Approach to Service Description for Matchmaking of Services. HP Technical Report. 2001.
14. VRP. The ICS-FORTH Validating Rdf Parser - VRP (2004). Available at: http://139.91.183.30:9090/RDF/VRP/.