Semantic Matchmaking of Advanced Personalized Mobile Services using Intelligent Agents

Krunoslav Tržec[1], Alisa Devlić[2], Gordan Ježić[2], Mario Kušek[2], Saša Dešić[1]

[1] Ericsson Nikola Tesla
R&D Centre
Krapinska 45, HR-10000 Zagreb, CROATIA
E-mail: krunoslav.trzec@ericsson.com

[2] University of Zagreb
Faculty of Electrical Engineering and Computing,
Department of Telecommunications
Unska 3, HR-10000 Zagreb, CROATIA
E-mail: alisa.devlic@fer.hr

Abstract: **This article presents semantic matchmaking of advanced personalized mobile services using intelligent middle agents in an electronic market (e-market). The traditional telecommunication services are implemented as Web services using emerging Parlay X Web Service framework and semantically described with DARPA Agent Markup Language for Services (DAML-S) ontology. This enables us to compose advanced personalized mobile services, such as traffic management Web service, that exploit benefits of both Internet information services and traditional telecom services. Moreover, using DAML-S for semantic description of information and telecom Web services opens up exciting opportunities in the automation of matchmaking process in the e-market. We have adopted a component-based approach in building our intelligent agents responsible for semantic matchmaking of personalized mobile services. A scenario with personalized mobile Web service for traffic management in the target location shows that proposed intelligent agent-based semantic matchmaking is more efficient than traditional matchmaking of Web services based on Web Service Description Language (WSDL) or Universal Description, Discovery and Integration (UDDI) infrastructure.**

INTRODUCTION

Service matchmaking is the process whereby potential trading partners become aware of each other's existence in an e-market. A buyer wishing to purchase access to a service must find potential service providers able to meet its needs. The matchmaking is required since the buyer's requirements may not be initially fully specified, and the service providers may offer similar, but not the same services. The process of matchmaking should not result in the service level agreement (SLA) becoming fully specified between the user and the selected service provider: this is the purpose of the negotiation process which follows after service matchmaking in the electronic market. This paper will focus on enhanced matchmaking process based on the Semantic Web reasoning techniques applied by the intelligent agents that play broker role in the multi-agent system for QoS management [1].

The broker agents, actually, represent middle agents that assist buyer agents in matchmaking of converged information (datacom) and telecom services from one or more relevant service providers (represented by seller agents) in the e-marketplace. In particular, matchmaking of advanced personalized mobile services will be considered since it can be represented as a composite Web service using Parlay X Web Service framework [2]. The term personalization in our case refers to the delivery of the content related to the current user status and target geographical location selected by the user. Because of its properties and wide applicability, broker components are a natural candidate for the Web services infrastructure. Unfortunately, the current Web services architecture does not include brokers with the functionality of semantic matchmaking.

Instead, matchmaking of Web services is usually done by the use of either WSDL or UDDI standard [3]. However, WSDL does not support semantic description of services. For example, it does not provide the definition of logical constraints between its input and output parameters. Furthermore, UDDI, another emerging Extensible Markup Language (XML)-based standard for Web service description, does not represent service capabilities so the matchmaking can only be done by string matching on the defined attributes such as name or address of service provider.

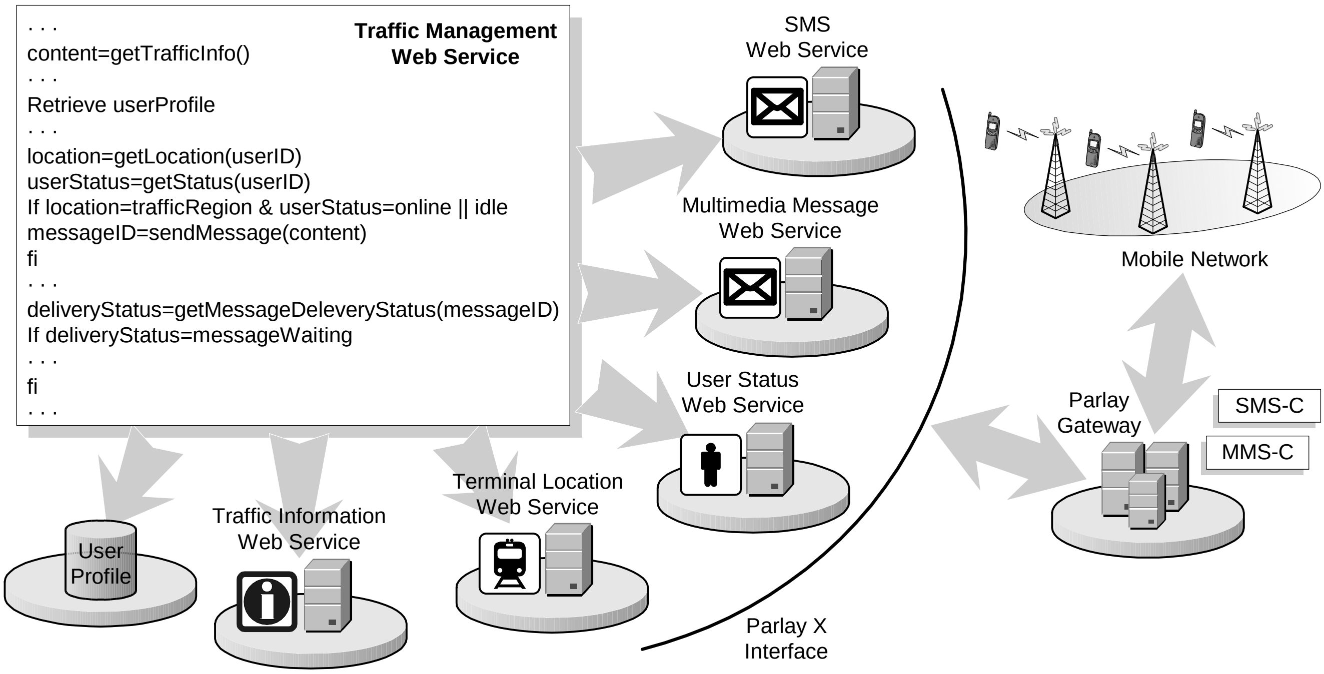

Figure 1 – Personalized mobile Web service for traffic management

In order to enable more efficient matchmaking of service providers' advertisements with users' requests, we have chosen DAML-S service description ontology as the basis to represent constructs like advertisements and service queries. DAML-S makes use of DAML+OIL which is an Artificial Intelligence (AI) inspired description logic (DL)-based language [3, 4]. Consequently, it supports our need for the semantic representation of services. DAML+OIL allows subsumption reasoning on concept taxonomies and the definition of relations between concepts. Moreover, it makes it possible to apply property restrictions on the parameters of service concepts. This means that we can use DAML-S to define the entities in SLA management lifecycle [4], such as advertisements and queries, and implement the matchmaking functionalities by using a DL reasoner to compute the subsumption relationships of those concepts.

PERSONALIZED MOBILE WEB SERVICES

Web services are designed to provide interoperability between diverse applications. The platform- and language-independent interfaces of Web services allow the easy integration of heterogeneous systems. Because of this flexibility, the idea of mobile Web services is to integrate datacom services with mobile applications, as well as mobile telecom services with PC-based applications. Taking into account user preferences specified in user profiles and using the Parlay X Web services, which provide a simple interface to telephony and other telecommunication services, it is possible to build advanced personalized mobile Web services.

We will show how personalization concept [5] (that refers to the mechanisms of discovering, selecting and combining services according to the user needs and preferences) is applied to the traffic management mobile Web service that shows traffic information in a location chosen by the user. This service, illustrated in Fig. 1, represents composition of Internet datacom Web service that offers traffic information and telecom Parlay X Web services that enable sending of SMS and MMS messages based on user status and selected location. The composite Web service first invokes a Web service that retrieves traffic information for the target location which user selected either by marking the location on the map (MMS) or by writing the name of the location (MMS/SMS). The user can also choose to see the traffic information in her current location (detected by terminal location Web service in which case our service becomes location-based service [6]). The traffic information can be sent to the user either as SMS or MMS message using the Parlay X Interface of the SMS or multimedia message Web service. These services invoke sending of message using either Short Message Service Center (SMS-C) or Multimedia Message Service Center (MMS-C) to reach the subscriber on the mobile network. Later, after the period defined in the user profile expired, the service checks if the traffic information has been successfully delivered to the user, and, if true, charges the subscriber.

Although a user can browse standard service categories to search for services or use a keyword query to obtain services via UDDI, the absence of a semantic service description results in low recall and yields imprecise results. Recently, researchers have combined the Semantic Web technology with the matchmaking and dynamic composition of Web services. DAML-S, a semantic markup language based on the DAML language family, has been introduced to represent Web services. It encodes services' properties, capabilities, interfaces, and effects in concept classes and subclasses, based on ontology. The DAML-S provides three basic kinds of knowledge associated with a Web service: Service Profile (what the service does), Service Model (how the service works), and Service Grounding (how to use the service). The service profile is the primary construct by which a service is advertised, matched, and selected. However, in some cases an agent involved in service matchmaking (broker) might also find it useful to inspect the service process model to answer more detailed questions about the service. Using service process model, broker agent can increase efficiency in fulfilling user service requests. In order to show the advantages of agent-based semantic matchmaking, we will consider the matchmaking process in an electronic market for mobile Web services.

MATCHMAKING PROCESS BASED ON DAML-S SERVICE DESCRIPTION

In order to place an advert, service provider must be able to specify the set of offered services. In many cases, these services will not be fully specified immediately. Because of this, it needs a language which is rich enough to allow an abstraction of a service to be advertised, together with constraints over that abstraction. Similarly, for a service user to make a query, she must be able to specify, as an abstraction together with constraints, the set of services she is interested in. When a query is made, this is treated as constraint on the acceptable set of services to the user. The appropriate intelligent agent must identify which advertisements are compatible with the query and respond with a list of all such advertisers and their advertisements. Ideally, it would also return an abstraction of a service, together with constraints, specifying the most general solution acceptable to both the user and the service provider.

Consequently, we have chosen that matchmaking process in the e-market for personalized mobile services is based on DAML-S ontology for describing capabilities of Web services. DAML-S integrates rich class representations with a process model designed to capture not only the control flow and data flow of Web services but also their real-world side effects (preconditions and effects). Its well-defined semantics allows automated service matchmaking in the e-market with a known outcome using powerful reasoning techniques.

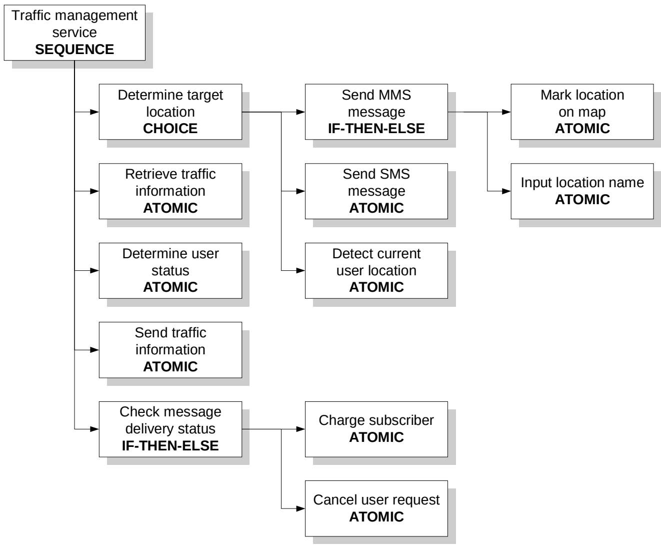

Figure 2 – DAML-S model of the traffic management service

Furthermore, service matchmaking can be enhanced by the use of DAML-S service models that are based on the key concept of a process, which describes a service not only in terms of inputs, outputs, preconditions, effects, but also, where appropriate, its composition of component subprocesses. DAML-S distinguishes between three types of processes: atomic, simple and composite. Atomic processes correspond to operations that the provider can perform directly; simple processes are elements of abstraction (they can be thought of as having single step executions but are not invocable); composite processes are collections of processes (either atomic or composite) organized on the basis of some control flow structure that is specified by control constructs. For example, a `sequence` construct defines composite process whose subprocesses are executed one after the other; `choice` control construct is used for non-deterministic choices between alternative control flows; `if-then-else` construct is used for conditional expressions.

Using the DAML-S service model in matchmaking process allows the user to make queries that take into account a way the inputs are transformed into outputs. For example, imagine a DAML-S advertisement for traffic management Web service. As it is shown in Fig. 2, such a composite service is considered to be a sequence of processes corresponding to the determination of target location, retrieval of traffic information in the selected target location, determination of user status, sending of retrieved traffic information to the user terminal, and getting of message delivery status. In the process of determination of target location where the traffic information should be checked, the user has the option to specify whether she wants to select location either by using the map, or by writing the name of the location, or by letting the service to determine her current location. Under the traditional matchmaking based on the service profile, the match would be successful only if the user provided the inputs for all three possible options. However, by using the process model, we can differentiate among the possible options and obtain a successful match by accepting an input corresponding to only one of these options.

The user can choose which parts of DAML-S service description will use. The root node in Fig. 2 represents the advertisement of entire traffic management Web service. The leaf nodes correspond to the atomic processes of the service provider advertisement. The inputs and outputs of these atomic nodes are listed in Table 1. Since the traffic management Web service is encoded by the DAML-S modeling primitives, we have defined, for example, the concept of user status with the `collection` DAML+OIL datatype. It can take one of the following values: `online`, `offline`, `busy` or `other`. The status attributes are reused from the Parlay X specification of the user status Web service as the idea of the Semantic Web lies in reusability of the existing schemes and their extensibility to the other ontologies. The concept of `UserStatus` is illustrated with the following DAML statements:

```
<daml:Class rdf:ID="UserStatus">
     <daml:oneOf
     parseType="daml:collection">
     <daml:Thing rdf:about="#Online"/>
     <daml:Thing rdf:about="#Offline"/>
     <daml:Thing rdf:about="#Busy"/>
     <daml:Thing rdf:about="#Other"/>
     </daml:oneOf>
</daml:Class>
```

Similarly, using DAML+OIL ontology language, we have defined all concepts specified in Table 1, which enables us to perform a series of test cases and evaluate our matchmaking algorithm. To understand the matchmaking algorithm we adopted in our prototype, the definition of the matching degree is introduced since it is not particularly useful merely to determine that an advertisement and query are not semantically incompatible. Based on the semantic equivalence of the query and the advertisement, the following categories of matches have been defined:

- *Exact*. The match is said to be exact when the requested service is the same as the advertised service;
- *PlugIn*. A plug-in match results when the requested service is subsumed by the advertised service. Hence, the advertised service can be substituted or plugged-in in place of the requested service. Such a match is less accurate but is also capable of satisfying the request;
- *Subsume*. When the requested service subsumes the advertised service we call such a match Subsume. Consequently, the advertisement will only partially satisfy the request;
- *Fail*. The matchmaking results in failure if neither of the above matching degrees is not satisfied.

Table 1 – Inputs and outputs for the traffic management service advertisement

Atomic Process	Inputs	Outputs
Mark location on map	location on map	location name
Input location name	location name	
Send SMS message	location name	
Detect current user location		location name
Retrieve traffic information		traffic information
Determine user status		user status
Send traffic information		traffic information
Charge subscriber		charging information
Cancel user request		cancel notification

Imagine a sample query posed to the service advertisement constructed from the specifications in Fig. 2 and Table 1. The user query, for example, requests a service that accepts the location name and sends a traffic information as the output. Taking into account service process model, the query results in a positive match even though we are not specifying all the inputs possible, since it is possible to execute the service with the inputs provided and produce the desired output.

BROKER COMPONETS FOR MATCHMAKING OF DAML-S SERVICE DESCRIPTIONS

The intelligent agents that play broker role in the e-market for personalized mobile services perform complex reasoning tasks, including interpreting service provider advertisements and service user queries. Since both advertisements and queries are based on DAML-S ontology, accomplishing these tasks requires agent components that allow reasoning on the information specified in those semantic service descriptions. Consequently, the broker agents are equipped with two components in order to perform matchmaking process: DAML Inference Engine and DAML-S Matchmaker. Figure 3 shows matchmaking process in the e-market as well as broker components used for matchmaking DAML-S service descriptions. In the matchmaking process buyer agent (which acts on behalf of an user) sends a query to the broker that compares the advertisements sent by seller agents (which act on behalf of service providers) and chooses the most appropriate service according to matchmaking algorithms.

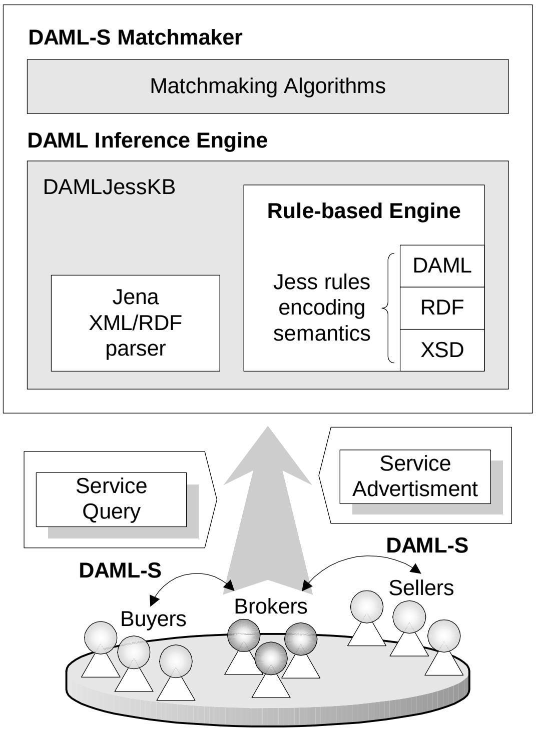

Figure 3 – Matchmaking process using broker agents

The matchmaking process requires two different reasoning tasks: to abstract from the query provider's required capabilities and to compare and match these capabilities with what available providers can really do. In order to accomplish these tasks the broker first uses DAML Inference Engine that, actually, represents DAMLJessKB off-the-shelf component for reasoning with DAML+OIL ontologies [7]. A broker using DAMLJessKB component can perform two classes of description logic reasoning: class instance reasoning and terminological reasoning about the relationships among the classes.

Rules supporting terminological reasoning enable inference on the relationships between classes and support DAML's underlying description logic semantics. This lets intelligent agents that play broker role to reason about which objects and classes can be automatically compared, contrasted, and manipulated on the basis of the input ontologies. Brokers invoke DAMLJessKB methods to load RDF (Resource Description Framework) as well as DAML documents. DAMLJessKB uses Another RDF Parser (ARP), which is part of the Jena toolkit [7], to parse RDF/DAML documents. Since the triples are collected by DAMLJessKB, some manipulations are necessary to correctly handle transition details that are related to syntax issues, anonymous nodes, and data types. Literal values are also manipulated to ensure consistent encapsulation without placing special demands on input ontologies or data.

Once DAML-S advertisement is parsed, DAMLJessKB asserts the triples into a rule-based engine along with the rules derived from the DAML semantics. For the rule-based engine, DAMLJessKB uses Jess [8] (Java Expert System Shell) since it provides easy interaction with Java programs, powerful scripting language, and rich expressiveness. The broker agent then queries Jess to obtain the information necessary for building the advertisement tree. Finally, the request query is parsed and the matchmaking process is executed on the basis of the matchmaking algorithms.

CONCLUSION

Research efforts in the area of the Semantic Web have contributed a lot to the vision of the personalized mobile services. Using DAML-S ontology, which has well-defined semantics, to describe personalized mobile Web services composed of Internet information and telecom Parlay X Web services enables semantic service matchmaking that results in enhanced utilization of personalized mobile services in the e-market. The semantic matchmaking of personalized mobile Web services can be efficiently performed by the use of broker intelligent agents that contain the off-the-shelf component for reasoning with DAML+OIL ontologies and matchmaking algorithms for DAML-S service descriptions.

REFERENCES

[1] K. Tržec and D. Huljenić: "Intelligent Agents for QoS Management", *Proceedings of the 1st International Joint Conference on Autonomous Agents & Multiagent Systems (AAMAS 2002)*, pp. 1405-1412, Bologna, Italy, 2002.

[2] H. Lofthouse, M. J. Yates, and R. Stretch: "Parlay X Web Services", *BT Technology Journal*, vol. 22, no. 1, 2004.

[3] M. Paolucci and K. Sycara: "Autonomous Semantic Web Services", *IEEE Internet Computing*, vol. 7, no. 5, pp. 34-41, 2003.

[4] K. Tržec: "Ontology Modeling for Agent-based Management of Service Level Agreements", *Proceedings of the 22nd International Conference on Computers in Telecommunications of the 26th International Convention (MIPRO 2003)*, pp. 51-56, Opatija, Croatia, 2003.

[5] W.-T. Balke and M. Wagner: "Towards Personalized Selection of Web Services", *Proceedings of the 12th International World Wide Web Conference (WWW 2003)*, Budapest, Hungary, 2003.

[6] A. Devlić and I. Podnar: "Location-aware Content Delivery Service Using Publish/Subscribe", *Proceedings of the tcmc2003, Workshop on Nomadic Data Services and Mobility*, Graz, Austria, 2003.

[7] J. Kopena and W. C. Regli: "DAMLJessKB: A Tool for Reasoning with the Semantic Web", *IEEE Intelligent Systems*, vol. 18, no. 3, pp. 74-77, 2003.

[8] E. J. Freiedman-Hill: *Jess in Action: Java Rule-based System*, Manning Publications, Greenwich, Connecticut, US, 2003.