Semantic User Profiles and their Applications in a Mobile Environment

Andreas von Hessling
Institut für Informatik
Universität Koblenz-Landau
D 56070 Koblenz
afri@uni-koblenz.de

Thomas Kleemann
Institut für Informatik
Universität Koblenz-Landau
D 56070 Koblenz
tomkl@uni-koblenz.de

Alex Sinner
Institut für Informatik
Universität Koblenz-Landau
D 56070 Koblenz
sinner@uni-koblenz.de

ABSTRACT
In this paper, we propose a peer-to-peer based mobile environment consisting of stations providing semantic services and users with mobile devices which manage their owner's semantic profile. In this environment the distributed computing power replaces centralized profile management. The service stations broadcast their services in a description logics based ontology language. When users equipped with their mobile device enter the range of a station, the semantic service is matched against the semantic user profile. The matching framework supports matching of services against user profiles, but is easily extended to matching services against other services or user profiles against other user profiles.

Author Keywords
Mobile environment, peer-to-peer, user profiles, semantic modeling, matching, description logics, ontologies

INTRODUCTION
The presented work is motivated by two aspects : the development of powerful mobile devices and the semantic web.

Current advances in the domain of mobile devices ranging from personal digital assistants (PDAs) and smart phones enable a new realm of applications. They have the computing power and storage capacity of last decade's high performance computers, and yet are small enough to carry around with you all the time. Today, mobile phones are widespread consumer devices which can be operated painlessly by the majority of the population. It is a safe bet that in the near future, regular mobile phones will have even more computing power than todays smart phones.

The development of the semantic web provided us with necessary tools to handle computer-understandable semantics. Semantic web pages are enriched with semantic annotations, usually encoded in some XML-flavored ontology language like DAML+OIL [13] or OWL [3]. These languages are machine understandable and are also strongly related to Description Logics (DL) [4,5]. Therefore they can be processed by DL reasoners, like FaCT [12] or RACER [11].

These two developments allow us to devise new applications which take into account the new opportunities offered by semantic information representation and the combination of computing power and ubiquitous availability.

In this paper, we will turn our attention towards the concept of semantic user profiles and their possible applications in a mobile environment. In such an environment, places that offer services are equipped with bluetooth-enabled computers to broadcast their services to passing people with bluetooth-enabled mobile devices which store their owner's profile. These computers and mobile devices need not be connected to the Internet, all necessary communication between them is done via short-range wireless bluetooth connections. On page 2, we will discuss the mobile environment in detail.

In this environment, there is no central mediator managing all the services and users, so interoperability aspects have to be considered. If we want a multitude of services and users to interoperate and communicate in a coherent way, we need a common language. In our case, this will be a description logics based ontology language similar to the languages that are used in the semantic web. Service descriptions and user profiles are encoded in this semantic language, which enables us to use the same user profile with a multitude of different services. This makes sense from two different viewpoints : first, it relieves us from having to create a new user profile from scratch (a non-trivial task) for every service provider, and second, the same interests in the profile can be relevant in a wide variety of services, possibly originating from many different providers

For illustration purposes, consider the following example : Bob is a person and owns a bluetooth-enabled PDA which manages his user profile. (See page 2) Bob is a devoted cinephile. Most of his free time, he goes to the movies or rents DVDs (no VCRs) at the local video store. Additionally, he is always eager to meet new people who share his hobby.

In all of the above scenarios (movie theater, video store and community), Bob uses his PDA with one single semantic user profile to get recommendations for services he is likely

to use. The user profile manages facts about which kinds of movies he likes, but also allows for statements like 'I want to see epic movies only in the cinema' or 'I am only interested in renting or buying DVDs, not VCRs'. Services, on the other hand, might state a list of movies to be shown (or rent), together with information about their classification, and their medium. If a service is matched positively against the interests stated in the user profile, the user will be notified about it and can decide whether to use it. The salient feature of semantic user profiles is that they can be used in a wide variety of completely different applications. The user carries his profile along with him and is independent of any specific service provider. Figure 1 visualizes the above scenario.

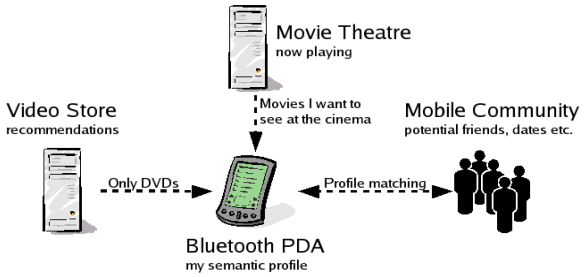

Figure 1. A scenario of related semantic services

In the following, we describe the requirements and features for a *semantic mobile environment* before we elaborate the concepts of *semantic services* and *semantic user profiles*.

A SEMANTIC MOBILE ENVIRONMENT

The Big Picture

The vision of the semantic web, as described in [7], was dominated by the idea of semantic web agents. It was to create an environment where *software agents roaming from page to page can readily carry out sophisticated tasks for users.*

Unfortunately, despite enormous efforts being done in the field, the semantic web has yet to happen. Up to now, the sheer amount of data on the world wide web and the lack of standard ontologies prevent the world wide web from evolving into the semantic web. Before these obstacles are overcome, the only viable solution is to develop small isolated semantic subnets.

Fortunately, in this work we will not deal with a semantic web approach as described above, even if there is a close relationship. Instead of the semantic web, we will consider the 'real' world, where people equipped with high-end mobile devices roam freely in an ubiquitously computerized environment. There is no need for central servers or agent technologies, since people with their mobile devices take on the role of semantic web agents. They go to places like video stores, movie theaters, railway stations, etc., where they can use the semantic services offered.

System Topology

Our mobile environment mainly consists of service providers and consumers with mobile clients who use them. Every client can however also play the role of a service provider, for example when offering 'dating services'. So the environment is not hierarchically organized. Rather, it can be seen as a mobile wireless peer-to-peer network, even if the purpose of some 'peers' is only to offer services (see also Figure 1).

This topology has several advantages over other approaches requiring a central server:

- *Independence*
  It is possible to set up access points for semantic services in the remotest parts of the world. No Internet connection is required. All we need is a mobile device with wireless connectivity and a semantic user profile and another device which offers semantic services over the same kind of wireless connectivity.

- *Cost Effectiveness*
  Internet connections from mobile devices are still expensive. Since we are independent from the Internet and any service providers to use semantic services, these costs do not apply.

- *Privacy Management*
  A major drawback of most centralized approaches is, that very personal data about many people is collected on a server. The concerned people have no control whatsoever about what exactly is stored and who has access to this data. In our approach all data about the user are stored on the owners mobile device. For most applications, all the computations using this data is done locally, so it is not necessary to send it to other peers. Should some applications require this, the user has the power not to allow it, or to disclose only the parts approved by her.

- *Scalability*
  Since the management of all user profiles is done on their respective mobile devices, no central profile database is required. New mobile clients can be added to the system at will, without encountering any scalability issues.

On the other hand some challenges arise from the intended setting. The increase in hardware requirements on the mobile side restricts the use to modern devices. Only recent handheld devices and smartphones have enough computing and storage resources for on-device user profile management and reasoning support. We are however confident that the mobile industry development will take care of this issue by developing more powerful devices. Another issue is that most bluetooth devices allow only seven open connections at once. Should a mobile client get service offers from more than seven parties at once, this might cause problems. Due to the limited duration of the matching process this does not evoke severe limitations. And finally, no description logic reasoner is available (yet) for mobile devices. Since computing resources are very limited on mobile devices, no of-the-shelf reasoner can be used. We are working on an implementation of a description logic reasoner for J2ME (Java 2 Mobile Edition) to overcome this situation.

In the following, we will describe more into detail the concepts of semantic services and semantic user profiles. Before we can do so, we have first to give a language to express semantics.

Description Logics as a Semantic Language

The idea of using Description Logics as a semantic language is not new [5]. In this paper we use the expressiveness of the DL $\mathcal{ALCHI}$, which is the standard DL $\mathcal{ALC}$ extended with role hierarchies and inverse roles. Syntax and semantics of $\mathcal{ALCHI}$ are amply described in [4].

In our mobile environment, we consider semantic services, semantic user profiles and concepts to which the services and profiles refer. The ontology about the concepts in the real world can be considered as a general vocabulary and is independent from the system itself, while the service and user profile ontologies describe specific capabilites of their respective components. We expect all participants of the service to incorporate the shared parts of the ontology. Figure 2 shows a sample ontology for the movies example in DL notation.

$$\begin{aligned} Thing &\sqsubseteq \top \\ Movie &\sqsubseteq Thing \\ Genre &\sqsubseteq Thing \\ Medium &\sqsubseteq Thing \\ DVD &\sqsubseteq Medium \\ VCR &\sqsubseteq Medium \\ CinemaScreen &\sqsubseteq Medium \\ Epic &\sqsubseteq Genre \\ Fantasy &\sqsubseteq Genre \\ SciFi &\sqsubseteq Genre \end{aligned}$$

Figure 2. A simple movie ontology

This simplistic movie ontology provides us with the necessary vocabulary to express more complex concepts like e.g.

$$RentalMovie \equiv Movie \sqcap \exists hasMedium.(DVD \sqcup VCR)$$

which describes a Movie that is either on a DVD or a VCR.

Semantic Services

A *semantic service* is of the following form:

$$Service \equiv \exists provides.Thing^{1}$$

Basically, a semantic service is a description of what is provided. Using different subroles of $provides$ we can also specify types of services. In the scope of this paper, we restrict ourselves to defining two subroles, $offers$ and $demands$. The semantics of these subroles is defined through the matching of services and profiles, but for now we can say that $demands$ allows mainly users to define short-term interests they want to advertise, while $offers$ is the standard way of providing services. It is possible to extend the definition with descriptions about the service provider and maybe service modalities, but in the scope of this paper we will stick to this simple definition.

Together with the vocabulary ontology (Figure 2), we are able to express complex services for movie theaters, video stores and persons (in the community scenario) such as:

$$\begin{aligned} &VideoStoreService1 \\ &\quad \equiv \exists offers.(\ Movie \\ &\qquad\qquad \sqcap \exists hasGenre.SciFi \\ &\qquad\qquad \sqcap \exists hasMedium.VCR) \\ &CinemaService1 \\ &\quad \equiv \exists offers.(\ Movie \\ &\qquad\qquad \sqcap \exists hasGenre.Epic \\ &\qquad\qquad \sqcap \exists hasMedium.CinemaScreen) \\ &BobService1 \\ &\quad \equiv \exists demands.(\ Movie \\ &\qquad\qquad \sqcap \exists hasGenre.Fantasy) \end{aligned}$$

The first service from the video store offers a SciFi Movie on a VCR tape. The second service from the cinema offers an epic movie shown on a cinema screen. Bob's service asks for fantasy movies.

The services are however of no use without their counterpart, a semantic user profile, which we will now describe.

Semantic User Profiles

A *semantic user profile* is a description of a user's interests and disinterests:

$$Profile \equiv \begin{array}{l} \prod_i \exists hasInterest.Interest_i \\ \prod_j \forall hasInterest.(\neg Disinterest_j) \end{array}$$

This definition states that the user is interested in all concepts covered by the $Interest_i$ concepts without the concepts covered by $Disinterest_i$. This profile description is quite similar to the one found in [8], but we do without encoding numeric levels of interest. Even though a user profile might contain additional information about his identity and possessions or capabilities, we will focus solely on an anonymous semantic description of her long term interests.

For the semantic services and profiles to work together, it is indispensable that the Things offered in the service definition use the same ontology (or vocabulary) as the interests from the profile definition. Figure 3 represents a sample profile of Bob stating that he likes epic movies shown in a cinema but does not like movies on VCR.

Semantic Matching

Having defined our notions of services and profiles we now introduce our approach to semantic matching. The goal of semantic matching is to determine whether a given profile is semantically compatible to a particular service and, if so, how well both do match. Ontologically speaking, the con-

[1] For a correct DL semantics, we have to additionally restrict the service provided to exactly what is provided with a value restriction: $Service \equiv \exists provides.Thing \sqcap \forall provides.Thing$. To enhance readability we omit these value restrictions. The same is valid later on for the formalization of semantic user profiles

$$
\begin{aligned}
Bob \equiv\ & \exists hasInterest.(\ Movie \\
& \sqcap \exists hasGenre.Epic \\
& \sqcap \exists hasMedium.CinemaScreen) \\
\sqcap & \forall hasInterest.(\ \neg Movie \\
& \sqcup \neg \exists hasMedium.VCR)
\end{aligned}
$$

Figure 3. Bob's profile

cepts describing a user's interests and dislikes have to be compared to the concepts offered by a service[2]:

$$
\begin{aligned}
UsersInterest &\equiv \exists hasInterest^{-1}.Profile \\
ServiceOffer &\equiv \exists offers^{-1}.Service
\end{aligned}
$$

Based on this, a concept *Match* is expressed as the intersection of these two concepts.

$$Match \equiv UsersInterest \sqcap ServiceOffer$$

If $Match$ is empty, the user is not interested in the service. Otherwise, we can check the subsumption relationships between $Match$, $UsersInterest$ and $ServiceOffer$ to determine a match degree. Horrocks et al. have proposed a matching approach that achieves this task [14].

RELATED WORK

In the following, we want to quickly describe some related approaches for semantic matchmaking of services.

DReggie

DReggie [1] is a dynamic service discovery infrastructure targeted at mobile commerce applications that exploits semantic matching using the XML-based DAML (DARPA Agent Markup Language) to describe services. A DReggie Lookup Server to which DReggie Clients submit their services performs the matching process and returns information about matches back to the clients [9, pp. 3-4].

The DReggie matching approach mainly differs from our system by taking into account services attribute priorities which provide a means for the prioritization of different aspects of a service. In addition to that, it contains an online-validation process for services received by the Lookup Server, which performs checks for semantically ill-formed expressions.

Agents2Go

The Agents2Go project [2] delivers an agent-based dynamic service discovery and information retrieval system using broker agents. In its application domain, the making of recommendations concerning restaurants, a broker agent mediates the ontology-based queries of personal agents and the services of restaurant agents on a mere semantic basis. In particular, the Agents2Go methodology takes an decentralized approach based on mobile devices which use Internet-connectivity to connect to broker agents [10].

The Agents2Go approach's differences to our system lie in the distinction between service providers (the restaurant agents) and the service requesters. Opposed to this, our application allows for the merging of the service providing and service exploiting functionality.

General Differences

The main differences to our approach have already be pointed out on page 2. By avoiding central servers and agent technologies, we have created a simple yet powerful system, which has many advantages over conventional approaches, namely better privacy management by avoiding to store all profiles on a single server and a cheaper infrastructure by avoiding Internet connections, without giving up the advantages of user profiles and semantic interoperability.

FUTURE WORK

A prototype of this work has been realized [15]. In this prototype, only emulated mobile devices were used and the semantic matching was performed by making a connection to a RACER [11] server using the DIG interface [6].

To realize the system on real mobile devices, we are currently working on an implementation of a DL reasoner for the J2ME platform. Another challenge is the automatic management of the user profile. Currently, facts about interests have to be added manually to the profile. This will be obsolete, once the profile management is able to evaluate user feedback on services offered. Other work areas involve the development and adaptation of industrial ontologies to be used for service and profile descriptions. Finally, we plan to evaluate the system by deploying some service stations in cooperation with industry partners and local authorities.

ACKNOWLEDGMENTS

This work is partially funded by the "Stiftung Rheinland-Pfalz für Innovation".

REFERENCES

1. http://ebiquity.umbc.edu/v2.1/project/html/id/47/, retrieved on 24th November 2003.
2. http://www.csee.umbc.edu/~oratsi2/myresearch/agentsToGo/, retrieved on 21th November 2003.
3. Grigoris Antoniou and Frank van Harmelen. Web ontology language: Owl. In S. Staab and R. Studer, editors, *Handbook on Ontologies in Information Systems*. Springer-Verlag, 2003.
4. Franz Baader, Diego Calvanese, Deborah L. McGuinness, Daniele Nardi, and Peter F. Patel-Schneider. *The description logic handbook: theory, implementation, and applications*. Cambridge University Press, 2003.

[2]We note that we also allow matching between offering and demanding services, but for simplicity's sake we only cover matching of offering services agains user profiles

5. Franz Baader, Ian Horrocks, and Ulrike Sattler. Description logics as ontology languages for the semantic web.

6. Sean Bechhofer. The DIG description logic interface: DIG/1.1. Technical report, University of Manchester, February 2003.

7. Tim Berners-Lee, James Hendler, and Ora Lassila. The semantic Web. *Scientific American*, 284(5):34–43, May 2001.

8. Andrea Calì, Diego Calvanese, Simona Colucci, Tommaso Di Noia, and Francesco M. Donini. A description logic based approach for matching user profiles. In *Proc. of the 2004 Description Logic Workshop (DL 2004)*, 2004. To appear.

9. D. Chakraborty, F. Perich, S. Avancha, and A. Joshi. Dreggie: Semantic service discovery for m-commerce applications. In *20th Symposiom on Reliable Distributed Systems*, 2001.

10. Harry Chen, Anupam Joshi, and Timothy W. Finin. Dynamic service discovery for mobile computing: Intelligent agents meet jini in the aether. *Cluster Computing*, 4(4):pp. 343–354, 2001.

11. Volker Haarslev and Ralf Möller. RACER system description. *Lecture Notes in Computer Science*, 2083:701–706, 2001.

12. Ian Horrocks. The FaCT system. *Lecture Notes in Computer Science*, 1397:307–312, 1998.

13. Ian Horrocks. DAML+OIL: A reason-able web ontology language. In *Extending Database Technology*, pages 2–13, 2002.

14. Lei Li and Ian Horrocks. A software framework for matchmaking based on semantic web technology.

15. Andreas von Hessling. Ontology based profile matching on mobile devices, February 2004. bachelor's thesis.