Advanced Resource Discovery Protocol for Semantic-enabled M-Commerce

Michele Ruta*
Politecnico di Bari
Via Re David 200, I-70125 Bari
Italy
phone: +39 080 5963515
fax: +39 080 5963410
e-mail: **m.ruta@poliba.it**

Tommaso Di Noia,
Politecnico di Bari
Via Re David 200, I-70125 Bari
Italy
phone: +39 080 5963903
fax: +39 080 5963410
e-mail: **t.dinoia@poliba.it**

Eugenio Di Sciascio,
Politecnico di Bari
Via Re David 200, I-70125 Bari
Italy
phone: +39 080 5963641
fax: +39 080 5963410
e-mail: **disciascio@poliba.it**

Francesco Maria Donini
Università della Tuscia
Via San Carlo 32, I-01100 Viterbo
Italy
phone: +39 0761 357 861
fax: +39 0761 357 887
e-mail: **donini@unitus.it**

Giacomo Piscitelli
Politecnico di Bari
Via Re David 200, I-70125 Bari
Italy
phone: +39 080 5963308
fax: +39 080 5963410
e-mail: **piscitel@poliba.it**

(* Corresponding author)

Advanced Resource Discovery Protocol for Semantic-enabled M-Commerce

Michele Ruta, Politecnico di Bari, Italia

Tommaso Di Noia, Politecnico di Bari, Italia

Eugenio Di Sciascio, Politecnico di Bari, Italia

Francesco Maria Donini, Università della Tuscia, Italia

Giacomo Piscitelli, Politecnico di Bari, Italia

INTRODUCTION

New mobile architectures allow for stable networked links from almost everywhere, and more and more people make use of information resources for work and business purposes on mobile systems. Although technological improvements in the standardization processes proceed rapidly, many challenges, mostly aimed at the deployment of value added services on mobile platforms, are still unsolved. In particular the evolution of wireless-enabled handheld devices and their capillary diffusion have increased the need for more sophisticated Service Discovery Protocols (SDPs).

Here we present an approach which improves Bluetooth SDP, to provide m-commerce resources to the users within a piconet, extending the basic service discovery with semantic capabilities. In particular we exploit and enhance the SDP in order to identify generic resources rather than only services.

We have integrated a "semantic layer" within the application level of the standard Bluetooth stack in order to enable a simple interchange of semantically annotated information between a mobile client performing a query and a server exposing available resources.

We adopt a simple piconet configuration where a stable networked zone server, equipped with a Bluetooth interface, collects requests from mobile clients and hosts a semantic facilitator to match requests with available resources. Both requests and resources are expressed as semantically annotated descriptions, so that a *semantic distance* can be computed as part of the ranking function, to choose most promising resources for a given request.

STATE OF THE ART

Usually, resource discovery protocols involve a requester, a lookup or directory server and finally a resource provider. Most common SDPs as Service Location Protocol (SLP), Jini, UPnP (Universal Plug aNd Play), Salutation or UDDI (Universal Description Discovery and Integration), include registration and lookup of resources as well as matching mechanisms (Barbeau M., 2000).

All these systems generally work in a similar manner. Basically a client issues a query to a directory server or to a specific resource provider. The request may explicitly contain a resource name with one or more attributes. The lookup server -or directly the resource provider- attempts to match the query pattern with resource descriptions stored in its database, then it replies to the client with discovered resources identification and location Liu J., Zhang Q., Li B., Zhu W. & Zhang J. (2002).

These discovery architectures are based on some common assumptions about network infrastructure under the application layer in the protocol stack. In particular, current SDPs usually require a continuous and robust network connectivity, which may not be the case in wireless contexts, and especially in the ad-hoc ones. In fact in such environments, network consistence varies continuously and temporary disconnections occur frequently, so bringing to a substantial decrease to traditional SDP performances (Chakraborty D., Perich F., Avancha S. & Joshi A., 2001).

Actually there are several issues that restrain the expansion of advanced wireless applications. Among them, the variability of scenarios. An ad-hoc environment is based on short range, low power technologies like Bluetooth (Bluetooth, 1999), which grant the peer to peer interaction among hosts. In such a mobile infrastructure there could be one or more devices providing and using resources but, as a MANET is a very unpredictable environment, a flexible resource search system is needed to overcome difficulties due to the host mobility. Furthermore, existing mobile resource discovery methods use simple string-matching, which is largely inefficient in advanced scenarios as the ones related to electronic commerce. In fact in these cases there is the need to submit articulate requests to the system, to obtain adequate responses (Chakraborty D. & Chen H., 2000).

With specific reference to the SDP in the Bluetooth stack, it is based on a 128 bit Universally Unique Identifier (UUID); each numeric ID is associated to a single service class. In other words Bluetooth SDP is code-based and consequently it can handle only exact matches. Yet, if we want to search and retrieve resources whose description cannot be classified within a rigid schema (*e.g.* the description of goods in a shopping mall) a more powerful discovery architecture is needed (Avancha S., Joshi A. & Finin T., 2002). SDP should be able to cope with non-exact matches (Chakraborty D. & Chen H., 2000), and to provide a ranked list of discovered resources, computing a distance between each retrieved resource and the request after a matchmaking process.

To achieve these goals, we exploit both theoretical approach and technologies of Semantic Web vision and adapt them to small ad-hoc networks based on the Bluetooth technology (Ruta M., Di Noia T., Di Sciascio E., Donini F.M. & Piscitelli G., 2005).

In a semantic-enabled Web –what is known as the Semantic Web vision– each available resource should be annotated using RDF (RDF Primer, 2004) with respect to an OWL ontology (Antoniou G. & van Harmelen F., 2003). There is a close relation between the OWL-DL subset of OWL and Description Logics (DLs) (Baader F., Calvanese D., Mc Guinness D., Nardi D. & Patel-Schneider P., 2002) semantics, which allows the use of DLs based reasoners in order to infer new information from the one available in the annotation itself.

In the rest of the paper we will refer to DIG (Bechhofer S., 2003) instead of OWL-DL because it is less verbose and more compact: a good characteristic in an ad-hoc scenario. DIG can be seen as a syntactic variant of OWL-DL.

THE PROPOSED APPROACH

In what follows we outline our framework and we sketch the rationale behind it. We adopt a mobile commerce context as reference scenario.

In our mobile environment, a user contacts via Bluetooth a zone resource provider (from now on *hotspot*) and submits her semantically annotated request in DIG formalism. We assume the zone server –which classifies resource contents by means of an OWL ontology– has previously identified shopping malls willing to promote their goods and it has already collected semantically annotated descriptions of goods. Each resource in the m-marketplace owns an URI and exposes its OWL description.

The *hotspot* is endowed with a *MatchMaker* (in our system we adapt the MAMAS-tng reasoner (Di Noia T., Di Sciascio E., Donini F.M. & Mongiello M., 2004)), which carries out the matchmaking process between each compatible offered resource and the requested one measuring a ”semantic distance”. The provided result is a list of discovered resources matching the user demand, ranked according to their degree of correspondence to the demand itself.

By integrating a semantic layer within the OSI Bluetooth stack at service discovery level, the management of both syntactic and semantic discovery of resources becomes possible. Hence, the Bluetooth standard is enriched by new functionalities which allow to maintain a backward compatibility (handheld device connectivity), but also to add the support to matchmaking of semantically annotated resources. To implement matchmaking and ontology support features, we have introduced a *Semantic Service Discovery* functionality into the stack, slightly modifying the existing Bluetooth discovery protocol.

Recall that SDP uses a simple request/response method for data exchange between SDP client and SDP server (Gryazin E., 2002). We associated unused classes of 128 bit UUIDs in the original Bluetooth standard to mark each specific ontology and we call this identifier *OUUID* (*Ontology Universally Unique Identifier*). In this way, we can perform a preliminary exclusion of supply descriptions that do not refer to the same ontology of the request (Chakraborty D., Perich F., Avancha S. & Joshi A., 2001). With OUUID matching we do not identify a single service, but directly the context of resources we are looking for, which can be seen as a class of similar services. Each resource semantically annotated is stored within the *hotspot* as resource record. A 32-bit identifier is uniquely associated to a semantic resource record within the *hotspot*, we call

SemanticResourceRecordHandle. Each resource record contains general information about a single semantic enabled resource and it entirely consists of a list of resource attributes. In addition to the *OUUID* attribute, there are *ResourceName*, *ResourceDescription*, and a variable number of *ResourceUtilityAttr_i* attributes (in our current implementation 2 of them). *ResourceName* is a text string containing a human-readable name for the resource, the second one is a text string including the resource description expressed in DIG formalism and the last ones are numeric values used according to specific applications; in general, they can be associated to context-aware attributes of a resource (Lee C. & Helal S., 2003) as for example its price or the physical distance it has from the *hotspot* (expressed in metres or in terms of needed time to get to the resource). We use them as parameters of the overall *utility function* that computes matchmaking results.

In particular, to allow the representation and the identification of a semantic resource description we introduced in the data representation of the original Bluetooth standard two new *data element type descriptor*: OUUID and DIG text string. The first one is associated to the type descriptor value 9 whereas to the second one corresponds the type descriptor value 10 (both reserved in the original standard). We will associate 1, 2, 4 byte as valid size for the first one and 5, 6, 7 for the DIG text string.

Since the communication is referred to the peer layers of the protocol stack, each transaction is represented by one request Protocol Data Unit (PDU) and another PDU as response. If the SDP request needs more than a single PDU (this case is frequent enough if we use semantic service discovery) the SDP server generates a partial response and the SDP client waits for next part of the complete answer.

By adding two SDP features *SDP_OntologySearch* (request and response) and *SDP_SemanticServiceSearch* (request and response) to the original standard (exploiting not used PDU ID) we inserted together with the original SDP capabilities further semantic enabled resource search functions, see table 1.

PDU ID	**Description**
0x00	Reserved
0x01	SDP_ErrorResponse
0x02	SDP_ServiceSearchRequest
0x03	SDP_ServiceSearchResponse
0x04	SDP_ServiceAttributeRequest
0x05	SDP_ServiceAttributeResponse
0x06	SDP_ServiceSearchAttributeRequest
0x07	SDP_ServiceSearchAttributeResponse
0x08	SDP_OntologySearchRequest
0x09	SDP_OntologySearchResponse
0x0A	SDP_SemanticServiceSearchRequest
0x0B	SDP_SemanticServiceSearchResponse
0x0C-0xFF	Reserved

Table 1: list of PDU IDs with corresponding descriptions

The transaction between service requester and *hotspot* starts after ad-hoc network creation. When a user becomes a member of a MANET, she is able to ask for a specific service/resource (by submitting a semantic-based description). The generic steps, up to response providing, for a service request are detailed in the following:

1. The user searches for a specific ontology identifier by submitting one or more $OUUID_R$ she manages by means of her client application
2. The *hotspot* selects OUUIDs matching each $OUUID_R$ and replies to the client

3. The user sends a service request (*R*) to the *hotspot*
4. The *hotspot* extracts descriptions of each resource cached within the *hotspot* itself, which is classified with the previously selected $OUUID_R$
5. The *hotspot* performs the matchmaking process between *R* and selected resources it shares. Taking into account the matchmaking results, all the resources are ranked with respect to *R*
6. The *hotspot* replies to the user.

It is important to remark that basically all the previous steps are based on the original SDP in Bluetooth. No modifications are made to the original structure of transactions, but simply we differently use the SDP framework. In what follows we outline the structure of the SDP PDUs we added within the original framework to allow semantic resource discovery.

The first one is the *SDP_OntologySearchRequest* PDU. Their parameters are shown in table 2.

PDU ID	**parameters**
0x08	- ***OntologySearchPattern*** - *ContinuationState*

Table 2: SDP_OntologySearchRequest PDU parameters

The *OntologySearchPattern* is a data element sequence where each element in the sequence is a OUUID. The sequence must contain at least 1 and at most 12 OUUIDs, as in the original standard. The list of OUUIDs is an ontology search pattern. The *ContinuationState* parameter maintains the same purpose of the original Bluetooth (Bluetooth, 1999).

The *SDP_OntologySearchResponse* PDU is generated by the previous PDU. Their parameters are reported in table 3.

PDU ID	parameters
0x09	- ***TotalOntologyCount*** - ***OntologyRetrievedPattern*** - *ContinuationState*

Table 3: SDP_OntologySearchResponse PDU parameters

The *TotalOntologyCount* is an integer containing the number of ontology identifiers matching the requested ontology pattern. Whereas the *OntologyRetrievedPattern* is a data element sequence where each element in the sequence is a OUUID matching at least one sent with the *OntologySearchPattern*. If no OUUID matches the pattern, the *TotalOntologyCount* is set to 0 and the *OntologyRetrievedPattern* contains only a specific OUUID able to allow the browsing by the client of all the OUUIDs managed by the *hotspot* (see the following *ontology browsing* mechanism for further details). Hence the pattern sequence contains at least 1 and at most 12 OUUIDs.

The *SDP_SemanticServiceSearchRequest* PDU follows previous PDU. Their parameters are shown in table 4.

PDU ID	parameters
0x0A	- ***SemanticResourceDescription*** - ***ContextAwareParam1*** - ***ContextAwareParam2*** - *MaximumResourceRecordCount* - *ContinuationState*

Table 4: SDP_SemanticServiceSearchRequest PDU parameters

The *SemanticResourceDescription* is a data element text string in DIG formalism representing the resource we are searching for, *ContextAwareParam1* and *ContextAwareParam2* are data element unsigned integers. In our case study, which models an m-marketplace in an airport terminal, we use them respectively to indicate a reference price for the resource and the hour of the scheduled departure of the flight. Since a generic client interacting with a *hotspot* is in its range, using the above PDU parameter she can impose –among others– a proximity criterion in the resource discovery policy.

The *SDP_SemanticServiceSearchResponse* PDU is generated by the previous PDU. Their parameters are reported in table 5.

PDU ID	**parameters**
0x0B	- *TotalResourceRecordCount* - *CurrentResourceRecordCount* - ***SemanticResourceRecordHandleList*** - *ContinuationState*

Table 5: SDP_SemanticServiceSearchResponse PDU parameters

The *SemanticResourceRecordHandleList* includes a list of resource record handles. Each of the handles in the list refers to a resource record potentially matching the request. Note that this list of service record handles does not contain header fields, but only the 32-bit record handles. Hence, it does not have the data element format. The list of handles is arranged according to the relevance order of resources, excluding resources not compatible with the request . The other parameters maintain the same purpose of the original Bluetooth (Bluetooth, 1999).

In all the previous cases, the error handling is managed with the same mechanisms and techniques of Bluetooth standard (Bluetooth, 1999).

Notice that each resource retrieval session starts after settling between client and server the same ontology identifier (OUUID).

Nevertheless if a client does not support any ontology or if the supported ontology is not managed by the *hotspot*, it is desirable to discover what kind of merchandise class (and then what OUUIDs) are handled by the zone server without any a priori information about resources. For this purpose we use the *service browsing* feature (Bluetooth, 1999) in a slightly different fashion w.r.t. the original Bluetooth standard, so calling this mechanism *ontology browsing*. It is based on an attribute shared by all semantic enabled resource classes, the *BrowseSemanticGroupList* attribute which contains a list of OUUIDs. Each of them represents the browse group a resource may be associated with for browsing.

Browse groups are organized in a hierarchical fashion, hence when a client desires to browse a *hotspot* merchandise class, she can create an *ontology search pattern* containing the OUUID that represents the *root browse semantic group*. All resources that may be browsed at the top level are made members of the *root browse semantic group* by having the root browse group OUUID as a value within the *BrowseSemanticGroupList* attribute.

Generally a *hotspot* supports relatively few merchandise classes, hence all of their resources will be placed in the root browse group. However, the resources exposed by a provider may be organised in a browse group hierarchy, by defining additional browse groups below the root browse group.

Having determined the goods category and the corresponding reference ontology, the client can also download a DIG version of it from the *hotspot* as *.jar* file (such a file extension – among other things– also allows a total compatibility with the Connected Limited Device Configuration (CLDC) technology).

Also notice that since the proposed approach is fully compliant with Semantic Web technologies, the user exploits the same semantic enabled descriptions she may use in other semantic web compliant systems (for example in the web site of a shopping mall). That is, there is no need for different customized resource descriptions and modelling, if the user employs different applications either on the web or in mobile systems. The syntax and formal semantics of the descriptions is unique with respect to the reference ontology and can be shared among different environments.

In e-commerce scenarios, the match between demand and supply involves not only the description of the good but also data-oriented properties. It would be quite strange to have a commercial transaction without taking into account price, quantity, availability among others. The demander usually specifies how much she is willing to pay, how many items she wants to buy, the delivery date. Hence, the overall match value depends not only on the distance between the (semantic-enabled) description of the demand and of the supply. It has to take into account the description distance with the difference of (the one asked by the demander and the other proposed by the seller), quantity, delivery date. The overall utility function combines all these values to give a global value representing the match degree.

Also notice that, in m-commerce applications, in addition to “commercial” parameters also context-aware variables should influence matching results. For example, in our airport case study, we consider the price difference but also the physical distance between requester and seller to weigh the match degree. The distance becomes an interesting value since a user has a temporal deadline for shopping: the scheduled hour of her flight. Hence, a resource might be chosen also according to its proximity to the user.

We will express this distance in terms of time to elapse for reaching the shop where a resource is, leaving from the *hotspot* area. In such a manner the *hotspot* will exclude resources not reachable by the user while she is waiting for boarding and it will assign to resources unlikely reachable (farther) a weight smaller than one assigned to easily reachable ones.

The above approach can be further extended to other data-type properties.

The utility function we used depends on:

p_D : price specified by the demander

p_O : price specified by the supplier

t_D : time interval available to the client

t_O : time to reach the supplier and come back, leaving from the *hotspot* area

s_match : score computed during the semantic matchmaking process, computed through *rankPotential* (Di Noia T., Di Sciascio E., Donini F.M. & Mongiello M., 2004) algorithm.

$$u(s\_match, p_D, p_O, t_D, t_O) = \frac{s\_match}{2} + \frac{\tanh \frac{t_D - t_O}{\beta}}{3} + \frac{(1+\alpha)p_D - p_O}{6(1+\alpha)p_D} \qquad (1)$$

Notice that p_D is weighted by a (1+α) factor. The idea behind this weight is that, usually, the demander is willing to pay up to some more than what she originally specified on condition that she finds the requested item, or something very similar. In the tests we carried out, we find α=0.1 and β=10 are values in accordance with user preferences. These value seem to be in some accordance with experience, but they could be changed according to different specific considerations.

RUNNING EXAMPLE

A simple example can clarify the rationale of our setting. Here we will present a case study analogous to the one presented in (Avancha S., Joshi A. & Finin T., 2002) by Avancha et al. and we face it by means of our approach.

Let us suppose a user is in a duty free area of an airport, she is waiting for her flight to come back home and she is equipped with a wireless-enabled PDA. She forgot to buy a present for her beloved little nephew and now she wants to purchase it from one of the airport gift stores.

In particular she is searching for a learning toy strictly suitable for a kid (she dislikes a child toy or a baby toy) and possibly the toy should not have any electric power supply.

Clearly this request is too complex to be expressed by means of standard UUID Bluetooth SDP mechanism. In addition, non-exact matches between resource request and offered ones is highly probable and the on/off matching system provided by the original standard in this case could be largely inefficient.

Hence both the semantic resource request and offered ones can be expressed in an DIG statement exploiting DL semantics and encapsulated in an SDP PDU.

The *hotspot* equipped with MAMAS reasoner collects the request and initially selects supplies expressed by means of the same ontology shared with the requester. Hence a primary selection of suitable resources is performed. In addition, the matchmaker carries out the matchmaking process between each offered resource in the m-marketplace and the requested one measuring a ”semantic distance” (Colucci S., Di Noia T., Di Sciascio E., Donini F.M. & Mongiello M., 2005). Finally the matchmaking results are ranked and returned to the user.

A subset of the ontology used as a reference in the examples is reported in Figure 1. For the sake of simplicity, only the class hierarchy and disjoint relations are represented.

Let us suppose that after the *hotspot* selects supplies, its Knowledge Base is populated with the following individuals whose description is represented using DL formalism:

- ***Alice_in_wonderland.*** Price 20$. 5 min from the *hotspot*:

  $\texttt{book} \sqcap \forall \texttt{has\_genre.fantasy}$

- ***Barbie_car.*** Price 80$. 10 min from the *hotspot*:

car ⊓ ∀suggested_for.girl ⊓ ∀has_power_supply.battery

- ***classic_guitar.*** Price 90$. 17 min from the *hotspot*:

  musical_instrument ⊓ ∀suitable_for.kid ⊓ (≤ 0 has_power_supply)

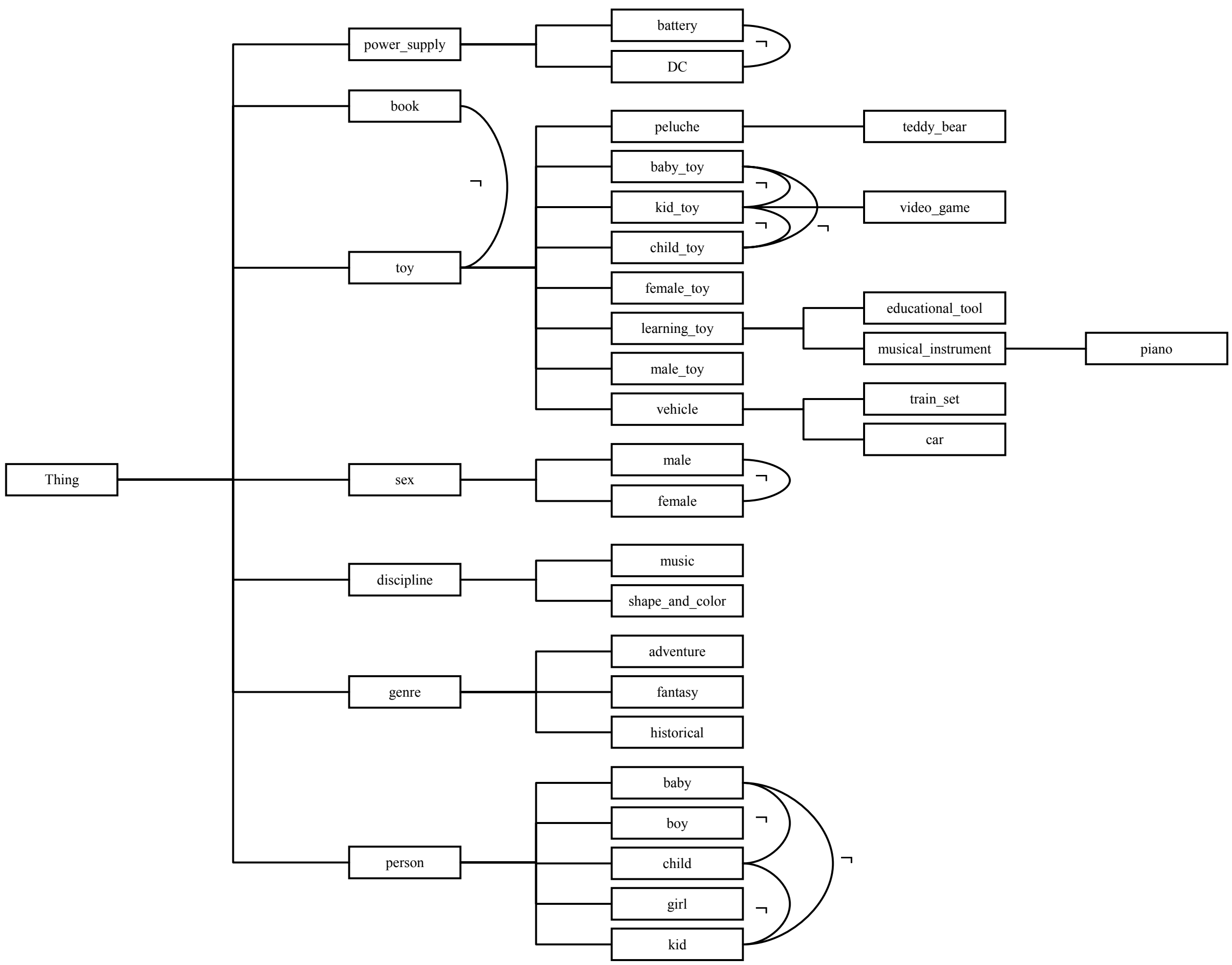

Figure 1: The simple toy store ontology used as reference in the example

- ***shape_order.*** Price 40$. 15 min from the *hotspot*:

  educational_tool ⊓ ∀suitable_for.child ⊓ ∀stimulates_to_learn.

  shape_and_color

- ***Playstation.*** Price 160$. 28 min from the *hotspot*:

  video_game ⊓ ∀has_power_supply.DC

- ***Winnie_the_pooh.*** Price 30\$. 15 min from the *hotspot*:

`teddy_bear ⊓ ∀suitable_for.baby`

On the other hand, the request ***D*** submitted to the system by the user can be formalized in DL syntax as follows:

`learning_toy ⊓ ∀suggested_for.boy ⊓ ∀suitable_for.kid ⊓ (≤ 0 has_power_supply)`

In addition she imposes a reference price of 200\$ (p_D=200) as well as the scheduled departure time as within 30 minutes (t_D=30).

In Table 6 matchmaking results are presented. The second column shows whether each retrieved resource is compatible or not with request *D* and, in case, the *rankPotential* computed result. In the fourth column, matchmaking results are also expressed in a relative form between 0 and 1 to allow a more immediate semantic comparison among requests and different resources and to put in a direct correspondence various rank values.

Finally in the last column results of the overall utility function application are shown.

demand – supply	**compatibility (y/n)**	**score**	**s_match**	**u(·)**
D - Alice_in_wonderland	n	-	-	-
D - Barbie_car	y	7	0.364	0.609
D - classic_guitar	y	3	0.727	0.748
D - shape_order	n	-	-	-
D - Playstation	y	5	0.546	0.378
D - Winnie_the_pooh	n	-	-	-

Table 6: matchmaking results

Notice that the semantic distance of the individual *classic_guitar* from *D* is the smaller one, then the system will recommend to the user this resource first. Hence the ranked list returned by the *hotspot* is a strict indication for the user about best available resources in the airport duty free piconet in order of relevance w.r.t. the request. Nevertheless a user can choose or not a resource according to her personal preferences and her initial purposes.

After having selected the best resource, the server of the chosen virtual shop will receive a connection request from the user PDA with its connection parameters and in this manner the transaction may start. The user can provide her credit card credentials, so that when she reaches the store, her gift will be already packed. This final part of the application is not yet implemented, but it is trivially achievable exploiting the above SDP infrastructure.

CONCLUSION AND FUTURE WORK

In this paper we have presented an advanced semantic enabled Resource Discovery Protocol for m-commerce applications. The proposed approach aims to completely recycle the basic functionalities of the original Bluetooth Service Discovery Protocol by simply adding semantic capabilities to the classic SDP ones and without introducing any change in the regular communication work of the standard. A matchmaking algorithm is used to measure the semantic similarity among demand and resource descriptions.

Future trends of the proposed framework aim to create a more advanced DSS to help a user in a generic m-marketplace. Under investigation is the support to creation of P2P small communities of mobile hosts where goods and resources are advertised and opinions about shopping are exchanged (Avancha S., D'Souza P., Perich F., Joshi A. & Yesha Y., 2003). If a user decides to "open" her shopping trolley sharing information she owns (purchased goods, discounts, opinion about specific vendors or products) the system will insert her in a buyer mobile community where she can exchange information with other users.

Another future activity focuses on strict control of the good advertising. In an m-marketplace, the system will send to various potential buyers best proposals about their interests.

We intend to implement a mechanism to advertise goods or services in a more direct and personalized fashion. From this point of view, an additional feature of the system is oriented to the user profiling extraction and management (Prestes R., Carvalho G., Paes R., Lucena C. & Endler M., 2004)(Ruta M., Di Noia T., Di Sciascio E., Donini F.M. & Piscitelli G., 2005)(von Hessling A., Kleemann T. & Sinner A., 2004). Without imposing any explicit profile submission to the user, the system could collect her preferences by means of previously submitted requests (Ruta M., Di Noia T., Di Sciascio E., Donini F.M. & Piscitelli G., 2005), *i.e.,* by means of the "history" of the user in the m-marketplace.

REFERENCES

Antoniou G. & van Harmelen F. (2003). "Web ontology language: Owl", *Handbook on Ontologies in Information Systems.*

Avancha S., D'Souza P., Perich F., Joshi A. & Yesha Y. (2003). "P2P m-commerce in pervasive environments", *ACM SIGecom Exchanges*, 3(4), 1-9.

Avancha S., Joshi A. & Finin T. (2002). "Enhanced service discovery in Bluetooth", *IEEE Computer*, 35(6), 96-99.

Baader F., Calvanese D., Mc Guinness D., Nardi D. & Patel-Schneider P. (2002). "The Description Logic Handbook". *Cambridge University Press*.

Barbeau M. (2000). "Service discovery protocols for ad hoc networking". *Workshop on Ad-hoc Communications (CASCON '00).*

Bechhofer S. (2003). "The DIG Description Logic Interface: DIG/1.1". http://dlweb.man.ac.uk/dig/2003/02/interface.pdf

Bluetooth specification document (1999). http://www.bluetooth.com.

Chakraborty D. & Chen H. (2000). "Service Discovery in the future for Mobile Commerce". *ACM Crossroads*, 7(2), 18-24.

Chakraborty D., Perich F., Avancha S. & Joshi A. (2001). "Dreggie: Semantic service discovery for m-commerce applications". *Workshop on Reliable and Secure Applications in Mobile Environment*.

Colucci S., Di Noia T., Di Sciascio E., Donini F.M. & Mongiello M. (2005). "Concept abduction and contraction for semantic-based discovery of matches and negotiation spaces in an e-marketplace". *Electronic Commerce Research and Applications*, 4(4), 345-361.

Di Noia T., Di Sciascio E., Donini F.M. & Mongiello M. (2004). “A system for principled matchmaking in an electronic marketplace”. *International Journal of Electronic Commerce*, 8(4), 9-37.

Gryazin E. (2002). “Service discovery in Bluetooth”. http://www.hpl.hp.com/techreports/2002/HPL-2002-233.pdf.

Lee C. & Helal S. (2003). “Context attributes: An approach to enable context awareness for service discovery”. *Symposium on Applications and the Internet (SAINT ’03)*, 22-30.

Liu J., Zhang Q., Li B., Zhu W. & Zhang J. (2002). “A unified framework for resource discovery and QoS-aware provider selection in ad hoc networks”. *ACM Mobile Computing and Communications Review*, 6(1), 13-21.

Prestes R., Carvalho G., Paes R., Lucena C. & Endler M. (2004). “Applying ontologies in open mobile systems”. *Workshop on Building Software for Pervasive Computing OOPSLA’04*.

RDF Primer-W3C Recommendation 10 February 2004. http://www.w3.org/TR/rdf-primer/

Ruta M., Di Noia T., Di Sciascio E., Donini F.M. & Piscitelli G. (2005). “Semantic based Collaborative P2P in Ubiquitous Computing”. *IEEE/WIC/ACM International Conference Web Intelligence 2005 (WI ‘05)*, 143-149.

von Hessling A., Kleemann T. & Sinner A. (2004). “Semantic user profiles and their applications in a mobile environment”. *Artificial Intelligence in Mobile Systems 2004*.

TERMS AND DEFINITIONS

Description Logics (DLs): A family of logic formalisms for Knowledge Representation. Basic syntax elements are concept names, role names, and individuals. Intuitively, concepts stand for sets of objects, and roles link objects in different concepts. Individuals are used for special named elements belonging to concepts. Basic elements can be combined using constructors to form concept and role expressions, and each DL has its own distinct set of constructors. DL-based systems are equipped with reasoning services: logical problems whose solution can make explicit knowledge that was implicit in the assertions.

M-marketplace: virtual environment where demands and supplies (submitted or offered by users equipped with mobile devices) encounter each other.

Ontology: an explicit and formal description referred to concepts of a specific domain (classes) and to relationships among them (roles or properties).

Piconet: Bluetooth based short range Wireless Personal Area Network. A Bluetooth piconet can host up to height mobile devices. More piconet form a *scatternet*.

SDP: Service Discovery Protocol. It identifies the application layer of an OSI protocol stack and manages the automatic detection of devices with joined services.

Semantically annotated resource: any kind of good, tangible or intangible (*e.g*. a document, an image, a product or a service) endowed of a description that refers to a shared ontology.

Semantic Matchmaking: the process of searching the space of possible matches between a request and several resources to find those best matching the request, according to given semantic criteria. It assumes that both the request and the resources are annotated according to a shared ontology.