

Hybrid Ontology-based Matchmaking for Service Discovery*

Devis Bianchini**
DEA - Univ. of Brescia
bianchin@ing.unibs.it

Valeria De Antonellis**
DEA - Univ. of Brescia
deantone@ing.unibs.it

Michele Melchiori**
DEA - Univ. of Brescia
melchior@ing.unibs.it

ABSTRACT

The paper presents an ontology-based approach to service discovery, apt to support flexible and efficient matchmaking between service descriptions. We propose a hybrid approach that combines two matching strategies: a deductive strategy based on Description Logics with a reasoning procedure exploiting ontology knowledge to assess the type of match among services; a similarity-based strategy exploiting information retrieval metrics to measure the degree of match among services.

Keywords

Semantic Web Services, Ontology-based Service discovery, Service Ontologies, Hybrid Matchmaking

1. INTRODUCTION

Automated service discovery under highly dynamic and context-dependent requirements is recognized as a key step towards effective integration and cooperation in multi-knowledge distributed environments. Specifically, the emergence of semantics is considered a key issue to enforce timely discovery and dynamic composition of services. The use of semantic descriptions of services has been widely motivated and recommended to improve the process of locating services that fulfil a given request, enabling its (total or partial) automation. Our aim in this work is to propose an ontology-based hybrid approach where different kinds of comparison strategies are combined together: (i) a *deductive strategy* based on Description Logics with a reasoning procedure exploiting ontology knowledge to assess the type of match among services and (ii) a *similarity-based strategy* exploiting retrieval metrics to measure the degree of match among services. Specifically, the similarity approach is applied after

*This work has been partially supported by the European EU NoE INTEROP (<http://www.interop-noe.org/>) and by the MAIS (Multichannel Adaptive Information Systems) FIRB Project funded by the Italian Ministry of Education, University and Research (<http://www.mais-project.it>).

** Department of Electronics for Automation, via Branze 38, 25123, Brescia (Italy).

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

SAC'06 April 23-27, 2006, Dijon, France

Copyright 2006 ACM 1-59593-108-2/06/0004 ...\$5.00.

the deductive one in order to rank selected services according to the measured matching degree. The hybrid approach has been described in more details in [1] and is based on an ontological framework formed by three components: domain ontology, thesaurus and service ontology.

2. ONTOLOGICAL FRAMEWORK FOR SERVICE MODELING

A Domain Ontology *DomONT* is used to conceptualize domain knowledge and to provide semantics to service descriptions and contains the OWL-DL definitions of the elements used to describe services (input/output parameters, operations, service categories) in terms of *concepts* and *semantic relationships* between them. A thesaurus *TH* contains *terminological relationships* referring to names of ontology elements (e.g., *synonymy*, *hypernymy*, etc.) to better capture their meaning; it is automatically derived from the lexical system WordNet. A Service Ontology *ServONT* organizes services on three layers of abstraction. In the middle layer we have the Abstract Services, that are introduced to summarize the functionalities of sets of similar Concrete Services; these ones are positioned in the lower level of the ontology and are directly invocable services that implement the functionalities represented by Abstract Services. Finally, at the top layer of abstraction of the ontology, we have Subject Categories, that organize Abstract Services into standard available taxonomies to provide a topic-driven access to them. Two kinds of semantic relationships between Abstract Services are identified to express when an Abstract Service adds some functionalities to the capabilities provided by another one (*specialization*) and when an Abstract Service provides the overall set of functionalities of a set of other ones (*composition*). Ontologies are described using *SHOIN(D+)* Description Logic which has been implemented using the XML-based syntax of OWL-DL.

3. DEDUCTIVE AND SIMILARITY-BASED MATCHMAKING

3.1 Deductive matching

In a deductive approach based on Description Logics the description \mathcal{R} of a request is compared with the description \mathcal{S} of an offer. Following general guidelines in [3], we define five kinds of match, that can be intuitively described as follows:

- *exact match*, when the request and the offer present the same functionalities (this is a strong condition);
- *plug-in match*, when the offer provides at least the required functionalities and possibly adds new ones;
- *subsume match*, when the functionalities provided by the offer are less than the required ones (it is like *plug-in match*,

but with the roles of \mathcal{R} and \mathcal{S} exchanged);

- *intersection match*, when the request and the offer present some common functionalities;
- *mismatch*, when no common functionalities exist between the request and the offer.

In our approach the kind of match is determined exploiting the domain knowledge provided by *DomONT* and verifying the existence of semantic relationships (*equivalence*, *subsumption*) between ontology concepts corresponding to elements of the compared service descriptions. In addition, the thesaurus \mathcal{TH} is used to take into consideration terminological relationships between names of service description elements and names of ontology concepts.

3.2 Similarity-based matching

The similarity between \mathcal{R} and \mathcal{S} is based on terminological relationships in the thesaurus and is computed through two properly defined coefficients based on the Dice's formula and widely experimented [2]: (i) an *Entity-based similarity coefficient*, obtained by comparing input/output parameter names of the service descriptions, to measure how much they are based on the same information; (ii) a *Functionality-based similarity coefficient*, obtained by comparing together operation names and their input/output parameter names, to measure how much the two services perform the same functionalities. Finally, the *Global similarity coefficient* $GSim$ between \mathcal{R} and \mathcal{S} is the measure of their level of overall similarity computed as the weighted sum of the Entity-based and Functionality-based similarity coefficients. The similarity analysis is based on the notion of *Name Affinity* between terms, which depends on the kind and on the number of terminological relationships between the considered terms in the thesaurus.

3.3 Hybrid matching

The deductive and similarity-based approaches are combined to enhance precision and flexibility of the matching process. The resulting hybrid matching algorithm is shown in Figure 1. Firstly, the deductive strategy is applied to classify the match between request \mathcal{R} and each available service \mathcal{S} (CLASSIFY-MATCH function), then similarity evaluation is performed according to the following rules:

- if *exact* or *plug-in match* occurs, from the request viewpoint the offer provides completely the required functionalities, so $GSim(\mathcal{R}, \mathcal{S})$ is set to 1 (full similarity) without computing the similarity coefficients;
- if *mismatch* occurs, $GSim(\mathcal{R}, \mathcal{S})$ is directly set to zero;
- if *subsume* or *intersection match* occurs, the offer fulfils the request only partially and similarity coefficients are computed (EVALUATE-SIMILARITY function) to quantify how much the offer satisfies the request; in this case, $GSim(\mathcal{R}, \mathcal{S}) \in (0, 1)$.

Only available services for which the $GSim(\mathcal{R}, \mathcal{S})$ is equal or greater than a given threshold ϕ are proposed among the searching results, ranked with respect to the $GSim$ values. The application of this hybrid approach ensures flexibility, since not only *exact* or *plug-in match* are considered (this would seem too restrictive), but also partial matches are taken into consideration by evaluating the similarity degree. Moreover, this evaluation decreases false negatives, since it does not exclude from the searching results those services that do not have *exact* or *plug-in match*, but are quite similar from the functional viewpoint with the request \mathcal{R} .

Furthermore, in our approach semantic relationships between concrete services, abstract services and subject categories in the Service Ontology are exploited to make more efficient the service discovery procedure (rows 16-25)). In fact, the discovery algorithm is applied at the category and

```

(1) FC-MATCH
(2) Inputs: a request  $\mathcal{R}$ ;
(3)            $ServONT, DomONT, \mathcal{TH}$ 
(4)           a threshold  $\phi$ ;
(5) Outputs: a list Results of candidate concrete services
           whose degree of similarity with  $\mathcal{R}$  is equal or greater than  $\phi$ 
           and properly ranked w.r.t. their degree of match with  $\mathcal{R}$ 

(6) begin
(7)   load to  $\mathcal{AS}$  the list of abstract services in  $ServONT$ ;
(8)   Candidates :=  $\emptyset$ ; Results :=  $\emptyset$ ;
(9)   foreach  $S \in \mathcal{AS}$ 
(10)    if( $S \notin$  Candidates) {
(11)     remove  $S$  from  $\mathcal{AS}$ ;
(12)      $TypeOfMatch_S :=$  CLASSIFY-MATCH( $\mathcal{R}, S, DomONT, \mathcal{TH}$ )
(13)     if( $TypeOfMatch_S <>$  'mismatch') {
(14)       $GSim(\mathcal{R}, S) :=$  EVALUATE-SIMILARITY( $\mathcal{R}, S, DomONT, \mathcal{TH}$ );
(15)      add tuple  $\langle S, TypeOfMatch_S, GSim(\mathcal{R}, S) \rangle$  to Candidates
(16)      foreach  $S_i \in \mathcal{AS}$ 
(17)       if  $S_i$  is a specialization of  $S$  in  $ServONT$  AND
(18)        ( $TypeOfMatch_S =$  'exact' OR
(19)          $TypeOfMatch_S =$  " 'plug-in' ") {
(20)           $GSim(\mathcal{R}, S_i) := GSim(\mathcal{R}, S)$ ;
(21)          add tuple  $\langle S_i, TypeOfMatch_{S_i}, GSim(\mathcal{R}, S_i) \rangle$ 
(22)           to Candidates;
(23)          remove  $S_i$  from  $\mathcal{AS}$ ;
(24)        }
(25)      }
(26)    }
(27)    else
(28)     foreach  $S_i \in \mathcal{AS}$ 
(29)      if  $S_i$  is a generalization of  $S$  in  $ServONT$ 
(30)       remove  $S_i$  from  $\mathcal{AS}$ ;
(31)    }
(32)  }
(33) end

```

Figure 1: The hybrid algorithm for service discovery

abstract layers and can be speed up by considering the semantic relationships between abstract services, according to the following intuition: if an abstract service \mathcal{S}_a matches with a given service request \mathcal{R} , then also abstract services that provide the same capabilities of \mathcal{S}_a (as expressed by means of semantic relationships) match with \mathcal{R} .

4. CONCLUSIONS AND FUTURE WORK

We have presented a hybrid service discovery approach that combines a deductive matching based on Description Logics and a similarity-based matching, also exploiting semantic organization of services in a Service Ontology to speed up semantic discovery. The combination of the two types of matching enhances flexibility and effectiveness of the matching process, while use of Service Ontology improves efficiency. The proposed approach is being experimented in the touristic domain to discover suitable available services for flight booking, hotel reservation, travel planning. Future efforts will investigate further matching modalities also according to levels of granularity in a wide usage scenario.

5. ADDITIONAL AUTHORS

Denise Salvi (email: denise.salvi@ing.unibs.it).

6. REFERENCES

- [1] D. Bianchini, V. De Antonellis, and M. Melchiori. Capability Matching and Similarity Reasoning in Service Discovery. In *Int. Conf. CAiSE Workshops INTEROP-EMOI*, Porto, 2005.
- [2] S. Castano and V. De Antonellis. A Framework for expressing Semantic Relationships between Multiple Information Systems for Cooperation. *Information Systems*, 123(3-4):253–277, 1998.
- [3] I. Horrocks and L. Li. A Software Framework for Matchmaking Based on Semantic Web Technology. *Special Issue on Semantic Web Services and Their Role in Enterprise Application Integration and E-Commerce, Int. Journal of Electronic Commerce (IJEC)*, 2004.