Web Services Discovery and Composition: a Schema Matching Approach

Sana Sellami
LSIS, Aix-Marseille University
Marseilles, France
sana.sellami@lsis.org

Omar Boucelma
LSIS, Aix-Marseille University
Marseilles, France
omar.boucelma@lsis.org

Abstract—Automated matching of service descriptions is the key to service discovery and composition. In this paper, we propose an approach for web services discovery and composition. The approach relies on (1) SAWSDL, a simple and generic annotation language, (2) an XML representation of a web service that carries both syntactic (e.g., WSDL) and semantic (e.g., SAWSDL) information, and (3) the reuse of available schema matchers. The approach departs from exiting ones because it does not advocate a specific matchmaking algorithm, and it promotes the combination of different schema matchers, allowing multiple discovery and composition strategies.

Keywords—Schema matching, service discovery, service composition.

I. INTRODUCTION

A number of solutions for web services discovery (WSD) and composition (WSC) has been proposed in the literature, but very few took leverage of existing schema matching (SM) tools and techniques that have been developed so far. This paper promotes such approach where SM techniques and tools are enhanced for WSD and WSC. Enhancement consists in adding semantics by means of annotated service descriptions, in using SAWSDL1, a simple and efficient mechanism for adding semantics to existing service descriptions, generally available as WSDL documents. Annotated descriptions are translated into XML documents then passed as inputs to well known matchers such as COMA++ [1] in order to find correspondences between a services request and a services' offer.

II. SCHEMA MATCHING FOR DISCOVERY AND COMPOSITION

Our approach consists of three phases: (1) both web services request (user demand) and offer (WSDL descriptions) are represented as XML schemas (XSD), amenable to serve as inputs for most existing schema matchers. Hence, given a set of WSDL descriptions, we first extract their salient features that we annotate with SAWSDL to provide semantic information, (2) we apply an existing matcher to find the correspondences between the input and the output of a service request and services' offer, and (3) finally, based on the service discovery result, we apply the same algorithm to find composite services corresponding to the user request.

A. From SAWSDL to XSD

A service is represented, first by means of its abstract part (Figure 1), then by its annotated version (Figure 2). We are using SAWSDL to annotate WSDL 2.02 salient features. As operations encapsulate those features, the matching process will focus on operations.

Figure 1. XML Schema representation (WSDL 2.0 interface)

Figure 2. Annotated Input/Output XSD

B. Web Service Discovery and Composition

Given a repository of services, and a user request, the goal of the discovery process is to automatically match request and services, i.e. to find the best possible match between elements of a request (a desired service) and those of a service being offered. Figure 3 illustrates this process where two services (request and offer) are matched by means of an operation that returns a car price.

Figure 3. Matching a request with an offer

1 http://www.w3.org/TR/sawsdl/

2 http://www.w3.org/TR/​wsdl20/
The matching process returns a set of tuples \((e_1, e_2, s)\) where \(e_1\) is a request element, \(e_2\) a service element and \(s\) a similarity value between 0 (strong dissimilarity) and 1 (strong similarity). The overall similarity of two services indicates the degree to which their respective operations match. For each pair of operations, we compare the input/output with their elements (defined in the XSD sequence types) and SAWSDL annotations. If there is no correspondence, then SAWSDL semantic annotations are considered as separate concepts. Because SAWSDL modelReference attribute refers to an OWL description, we are using an OWL matcher to find the similarities between ontological concepts. Finally, the individual similarity values are aggregated in order to compute the overall similarity between operations.

For experimentation purposes, we first used COMA++ [1] because it is a composite matcher that takes as input schemas or ontologies and combines different matching algorithms. Then we compared our approach with the one used with SAWSDL-MX [3], a hybrid SAWSDL web service matchmaker which performs both logic-based and IR-based (test retrieval) matching. Moreover, SAWSDL-MX is well-accepted in the semantic service matchmaking research community. For compatibility reasons, each testbed service contains one interface per service, one operation per interface, and one input/output per operation. Figure 4 below illustrates our experimentation.

![Figure 4. Average precision numbers](image1)

The composition process also relies on the core matching method described above. A composition task consists in finding a chain \(S_1, S_2, \ldots, S_N\) of Web services. We reformulate the composition problem as a list of matching problems where the output of a service \(S_i\) should match the input of a service \(S_{i+1}\). Figure 5 below illustrates this process.

![Figure 5. Composite service](image2)

### III. RELATED WORK

Web services discovery and composition is still an important research area [5],[4],[2]. Most of the approaches developed so far deal either with WSDL or OWL services. Discovery of SAWSDL annotated services approaches have been recently proposed [3]: this is partly due to the fact that SAWSDL specification has been recommended by the W3C in 2007.

To the best of our knowledge, most of SAWSDL based approaches deal with discovery rather than with the composition problem. Moreover, those approaches rely on ad-hoc matchmaking algorithms rather than the reuse of existing schema matchers. We are advocating here is to tackle both discovery and composition in reusing a whole set of schema matching techniques and tools that are being made accessible.

### IV. CONCLUSION

In this paper, we have proposed an approach for service discovery and composition in reusing existing matchers. Our goal is to transform web service descriptions (WSDL with SAWSDL annotations) into a generic XML representation that can be processed by existing schema matchers.

We are currently conducting experimentation and finalizing our framework. Future work will investigate the enrichment of WSDL descriptions, taking into account other properties (e.g., non functional properties) that may play an important role in the filtering process.

### REFERENCES


