A goal-oriented perspective on approaches to web service discovery

Saïd Assar  
Telecom Ecole de Management, Institut Mines-Telecom  
9, rue C. Fourier. 91011 Evry, France  
said.assar@it-sudparis.eu

Carine Souveyet  
CRI, University of Paris I La Sorbonne  
90, rue de Tolbiac, 75013 Paris, France  
carine.souveyet@univ-paris1.fr

Kadan Aljoumaa  
Higher Institute of Applied Science and Technology  
P.O. Box 31983, Damascus, Syria  
kadan.aljoumaa@hiast.edu.sy

Abstract—Goal modeling is a prominent design paradigm in various domains of information systems engineering. In the field of service oriented computing (SOC) and in the field of semantic web services, emergent research works are basing their engineering approach on the goal concept. Because of the complexity of processes underlying SOC, the usage of the goal concept can vary to a large extent. In this paper, we study approaches to service oriented engineering and propose a framework to analyze and better understand how the goal concept is used in web service discovery. The framework is inspired by the four world vision of information systems engineering (i.e. subject, system, development and usage). Using this framework, we review eight prominent research works in SOE. Through this analysis, we seek to better understand the link between semantic web technologies and the goal concept, and what are the challenging issues in terms of goal usage in service discovery.

Keywords: goal-oriented web service; business-oriented web service; semantic web service; ontology based search; semantic similarity

I. INTRODUCTION

Service orientation is a major trend in distributed computing. Relying on the web service core technology, IT systems in the future shall be composed of web services as the building block instead of proprietary solutions. Although there are reports on successful implementation of service-oriented architecture (SOA) in industry [1], there are still many problems and challenges facing full scale adoption [2] and enterprises maturity towards SOA is still in its early phases [3]. Indeed, the technologies underlying SOA are very complex, and service-oriented engineering (SOE) find its origin in diverse disciplines that are woven together in an intricate manner [4]. Two points of view have to be taken into account: service providers and service consumers. They have different concerns and are confronted with different set of problems [5]. In order to reach economies of scale, service providers need to offer a service portfolio that is highly standardized and stable over time. On the other hand, service consumers have to take the current offering of services as a starting point and try to align their requirements with these services.

From the consumer point of view, a critical step in this process is service discovery, i.e. the identification of existing web services that can potentially be used in the context of a new web-based application. The underlying problem consists in matching user requirements expressed in queries with service description stored in service repositories. Initial technologies developed for this end (i.e. WSDL1 and UDDI2) are text and keyword based. They provide limited possibilities for service discovery and exploration, and large scale practical experimentations have proved this limit [6]. Semantic web services emerged in order to enhance these possibilities [7][8][9][10]. Semantic web services are described with meta-data on the basis of domain ontologies as a mean to enable their automatic location, execution, combination and use. Technological approaches for semantic web services follow two slightly different paths. The first uses specific ontologies, either adapted from the semantic web (i.e. OWL-S3) or specifically developed for this purpose (i.e. WSMO4 and WSML5). The second path extends the existing WSDL standard by adding semantic annotations which defines links between the web service and ontological domain knowledge (i.e. SAWSDL6, standardized by W3C).

From a requirements engineering point of view, the goal concept is known to be familiar to business users, and goal modeling is recognized as a cornerstone approach for requirement elicitation, negotiation and validation [11][12]. Further, in the ontological approach to semantic web service description, the goal concept is explicitly part of the WSMO model. From another point of view, service is also a rather familiar concept to business users [13][14], and bringing the goal and the service concepts together in a general vision to service-oriented engineering is an appealing endeavor [15][16].

1 WSDL: Web service Description Language  
2 UDDI: Universal Description, Discovery, and Integration  
3 OWL-S: a specific ontology for semantic web services built on top of Web Ontology Language (OWL)  
4 WSMO: Web Service Modeling Ontology  
5 WSML: Web Service Modeling Language  
6 SAWSDL: Semantic Annotations for WSDL
A. Problem addressed

Service discovery and semantic web technologies have received tremendous attention in the academic research community. However, to our knowledge, only a limited set of service oriented engineering approaches are clearly based on the goal concept. To get an order of magnitude of the existing corpus of research related to these topics, we run six simple queries using the Advanced Scholar Search functionality on Google Scholar. The search was restricted to scientific articles (i.e. excluding patents), and to only two subject areas (Business and Engineering). We surrounded search terms in brackets, we used the search field with all of the words, and we limited the search to words that occur in the title of the article. The results are shown in Table 1. Although the terms goal and service are well occurring in research paper titles, terms goal and web service or semantic web service show limited occurrence and confirms our claims.

<table>
<thead>
<tr>
<th>Search strings for Google Scholar</th>
<th># of hits</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;semantic web service&quot;</td>
<td>872</td>
</tr>
<tr>
<td>&quot;discovery&quot; + &quot;web service&quot;</td>
<td>718</td>
</tr>
<tr>
<td>&quot;discovery&quot; + &quot;semantic web service&quot;</td>
<td>236</td>
</tr>
<tr>
<td>&quot;goal&quot; + &quot;service&quot;</td>
<td>129</td>
</tr>
<tr>
<td>&quot;goal&quot; + &quot;web service&quot;</td>
<td>21</td>
</tr>
<tr>
<td>&quot;goal&quot; + &quot;semantic web service&quot;</td>
<td>7</td>
</tr>
</tbody>
</table>

Thus, the objective of this paper is to explore the role and the usage of goal modeling in service-oriented engineering in general, and in service discovery in particular. More precisely, we seek answers to the following two related research questions:

**RQ1**: What are the links between semantic web services and goal-orientation in existing approaches to service oriented engineering in general, and to service discovery in particular?

**RQ2**: What are the important and challenging issues in terms of goal usage in service discovery?

B. Research method

The work presented here is based on a previous work initially thought out for a state of the art chapter in a PhD thesis [17]. We analyze here a set of carefully selected existing research works in service-oriented engineering in order to answer RQ1 and RQ2. This set was selected after a large review of existing publications directly or indirectly related to requirement engineering for services, to SOA and to semantic web services. Our investigation method does not however strictly fit into the category of systematic reviews [18], as the studied approaches have been selected on a one by one basis, and we did not base the selection process on the exclusive results of some precisely defined search process. Two selection criteria's were applied: (i) the selected work was prominent (i.e. a significant level of citations in Google Scholar), and at least two significant publications were available, and (ii) the approach was preferably implemented in a prototype and not limited to conceptual or theoretical considerations. Beyond that, we were particularly attentive to works in which the goal approach was claimed to be used.

We elaborate an analysis framework that is loosely based on the four worlds framework originally proposed for information systems engineering in [19][20], and which is regularly used for analysis purposes (see [21][11] for most prominent works). Using this framework, we look at approaches to service discovery in SOE in a comprehensive manner, and reflect on current investigations and open issues that provide a foundation for further research.

The structure of this paper is the following: in section 2, we present the analysis framework and describe it in detail. Section 3 applies our framework to eight service-oriented approaches and synthesis the results in table. In section 4, we comment on the content of the table and discuss the research questions. In section 5, we briefly review some related works, and then present in section 6 concluding remarks.

II. AN ANALYTICAL FRAMEWORK

Figure 1 presents the framework with all its variables. The subject view concerns the web service concept and its nature in the studied approach, and the usage view concerns its usage objectives. The representation view deals with the approach's conceptual foundations in terms of modeling choices to represent the products and the process on which the approach is based. Finally, the implementation view characterizes the technical facets of the approach in terms of design choices. Variables will be detailed in this section according to each view of the framework. For each variable are enumerated a set of values; all variables are mandatory and – unless specified – are multi-valued.

![Figure 1. The framework for analyzing SOE](image)

A. The subject view

This view contains two variables. The first is related to the nature of the service as it is defined and manipulated by the approach. Possible values are: Technological, Business, Goal. The second variable is related to the granularity of the service, it can be Simple and/or Aggregated (i.e. composite services).
B. The usage view

The objectives of an approach directly impact its proposed functionalities. A first objective is to define a new service by describing it and by publishing it in a repository; this definition could eventually be constructed by composing some existing services previously discovered. Other objectives are to discover existing services with the help of query and search functionalities, and to explore existing services in a browse like mode. A final objective is to invoke a service by executing it. Theses objectives and functionalities are related to different stages of the SOE: definition stage is on the provider side, design stage is at the user side (this includes requirements definition and system design), and runtime stage when the application is run and the services are invoked.

To resume, objectives = \{Define, Discover, Invoke\} and functionalities = \{Describe, Publish, Formulate a query, Search, Browse, Compose, Execute\}.

C. The representation view

This view is the most complex one in the framework. It is concerned with characterizing the models and the languages that are used by the approach. It is further divided in two sub-views: we first analyze product models, and then look at process models.

From the product view, the first element is the user input at design time. How does the user express its requirements when searching for a service? According to the approaches we have studied in this work, in its simplest form, this could be natural language list of words or structured sentences. More sophisticated forms are possible such as templates, business language and are further processed with the help of generic knowledge in SPARQL. 

The second variable characterizes the query model; it can be a list of keywords or a formal query for searching XML documents (XQuery) or for searching ontological knowledge (SPARQL 8). The third variable concerns the repository model used to describe the service. This could be either basic web service technology (WSDL) or semantic oriented technologies (c.f. section I). The third variable concerns the provider input at the definition stage, it is characterized in a similar way to user input. Finally, the last variable is the category of ontological knowledge. This can be generic (e.g. WordNet public ontology), or oriented towards some specific business application domain (e.g. tourism), or oriented towards technological knowledge about web services (e.g. YASA approach [22]).

From the process view, we consider three important processes in a SOE approach. The first is the publication process associated with description and publication functionalities (c.f. usage view). This process can be raw without any support from the system, or guided if the system provides some methodological support, or intelligent if the system has some reasoning capacities on what the service provider is doing during this process. In an identical manner, the process when an end-user is formulating a query can be characterized as raw, guided or intelligent. Finally, the matching process when it comes to search for services in the repository can be characterized as lexical if the matching is done at the string (or word) level, or semantic if it is done at the concept level.

D. The implementation view

The first variable in this view concerns the techniques that are used to implement the matching process. For lexical matching, a large set of techniques have been developed. They are known as Information Retrieval (IR) techniques, and two essential characteristics for these techniques are the recall (i.e. returned number of correct answers in relation to all correct answers) and the precision (i.e. returned number of correct answers in relation to all returned answers). For semantic matching, what is measured is similarity and semantic distance between two concepts in relation to some ontological representation. Trip and Travel would be similar concepts in tourism ontology, and the semantic distance between Hotel and Car location would be narrower than between Hospital and Beach. These two families of measurement techniques can be combined in order to enhance matching process [23]. The last variable in the implementation view is related to the software status of the proposed approach, it indicates if a software prototype has been realized or not.

III. APPLYING THE FRAMEWORK

Table II presents the application of the analysis framework to eight different SOE approaches. As mentioned at the beginning of this paper, these approaches have been selected on a one by one basis. We will in the following subsections briefly highlight the main characteristics of each approach.

A. Approach by Gomez & al. ([24] [25])

The GODO system is one of the oldest prototypes for semantic service discovery, to which have been lately added guiding facilities (GGODO). It exploits the goal concept as defined in WSMO semantic approach. Goals are extracted from queries in natural language through lexical analysis, and then matched against semantic web services' goals descriptors. Guidance is provided with the help of goal templates.

B. Approach by Klusch & al. ([26] [27])

The originality and strength of this approach lie in its hybrid (i.e. lexical and semantic) matching process. User input is expressed using OWL-S templates that are directly matched against OWL-S service descriptors. No guidance is provided by the system. The goal concept is not used in this approach.

C. Approach by Zachos & al. ([28][29])

This approach targets requirements elicitation using query expansion techniques. User queries are expressed in natural language and are further processed with the help of generic knowledge in WordNet ontology. Query extension and service exploration are expected to help user in better understanding requirements. The goal concept is not used in this approach.

D. Approach by Stollberg & al.([30][31])

Based on WSMO framework, the goal concept is directly exploited for service discovery. A graphical user interface visualizes hierarchies of goals structured as graphs with respect to their semantic similarity. A limited form of guidance is provided using goal templates.

---

8 SPARQL (Simple Protocol And RDF Query Language) is for querying RDF data sources (RDF: standard for encoding metadata in the Semantic Web)
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Subject</td>
<td>Service nature</td>
<td>Technological</td>
<td>Technological</td>
<td>Technological</td>
<td>Technological</td>
<td>Technological</td>
<td>Technological</td>
<td>Business</td>
<td>Business</td>
</tr>
<tr>
<td></td>
<td>Granularity</td>
<td>Atomic</td>
<td>Atomic</td>
<td>Atomic</td>
<td>Atomic</td>
<td>Atomic</td>
<td>Atomic</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Usage</td>
<td>Objectives</td>
<td>Discover</td>
<td>Define, Discover</td>
<td>Discover, Invoke</td>
<td>Discover, Invoke</td>
<td>Discover</td>
<td>Discover</td>
<td>Discover</td>
<td>Discover</td>
</tr>
<tr>
<td></td>
<td>Functionalities</td>
<td>Formulate a query, Search</td>
<td>Describe, Publish, Formulate a query, Search</td>
<td>Formulate a query, Search, Browse, Execute</td>
<td>Formulate a query, Search, Compose, Execute</td>
<td>Formulate a query, Search</td>
<td>Formulate a query, Search</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Representation</td>
<td>User input</td>
<td>NL, Goal</td>
<td>OWL-S template</td>
<td>NL</td>
<td>Goal template</td>
<td>Task, Goal</td>
<td>SAWSDL Template</td>
<td>Goal, Map</td>
<td>Goal, Map</td>
</tr>
<tr>
<td></td>
<td>Query</td>
<td>SPARQL</td>
<td>?</td>
<td>XQuery</td>
<td>?</td>
<td>SPARQL (?)</td>
<td>SPARQL</td>
<td>Keywords</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Service descriptor</td>
<td>WSMO+WSML</td>
<td>OWL-S</td>
<td>WSDL</td>
<td>WSMO+WSML</td>
<td>WSMO+WSML</td>
<td>SAWSDL extended, OWL-S</td>
<td>OWL-S</td>
<td>WSDL</td>
</tr>
<tr>
<td></td>
<td>Provider input</td>
<td>-</td>
<td>OWL-S template</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Ontology's category</td>
<td>Domain oriented</td>
<td>Domain oriented</td>
<td>Domain oriented</td>
<td>Domain oriented (business tasks and goals)</td>
<td>Domain oriented (technical)</td>
<td>Domain oriented</td>
<td>None</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Publication</td>
<td>-</td>
<td>Raw</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Query formulation</td>
<td>Guided</td>
<td>Guided, Intelligent</td>
<td>Raw (tasks), Intelligent (goals)</td>
<td>Raw</td>
<td>Guided</td>
<td>Raw</td>
<td>Guided</td>
<td>Raw</td>
</tr>
<tr>
<td></td>
<td>Implementation</td>
<td>IR based, Semantic distance</td>
<td>IR based, Semantic distance</td>
<td>IR based, Semantic distance</td>
<td>Semantic distance (?)</td>
<td>Semantic distance (?)</td>
<td>Semantic distance (?)</td>
<td>IR based</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ontology's usage</td>
<td>External</td>
<td>Internal</td>
<td>Internal</td>
<td>Internal (technical), External (domain)</td>
<td>Internal (map fragments), External (domain)</td>
<td>None</td>
<td>None</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Software status</td>
<td>Prototype</td>
<td>Prototype</td>
<td>Prototype</td>
<td>Prototype</td>
<td>Prototype</td>
<td>Prototype</td>
<td>None</td>
<td></td>
</tr>
</tbody>
</table>

(*) : We use a question mark if the information was unavailable or if we could not know for sure
E. Approach by da Silva & al. ([32][33])

Its originality lies in the combination of two user oriented concepts: goals and tasks. Relying on the WSMO framework, experts define domain and task ontologies. Tasks provide knowledge about behavior descriptions, and are mapped to services. Queries can be expressed directly as goals; they are then semantically matched with goals in WSMO ontology.

F. Approach by Chabeb & al. ([22][34])

This approach combines SAWSDL and OWL-S. In order to give type (i.e. category) information to semantic annotations, a technical ontology is constructed using OWL-S. Annotations are enriched with links to the technical ontology for type information, and user queries are matched against this enriched description. The goal concept is not used in this approach.

G. Approach by Mirbel & al.([35][36])

In this approach, queries are input as maps in which user intentions (i.e. goals) are linked using strategies which express manners to attain goals [42]. Maps describe processes at a high level of abstraction. They are translated into SPARQL queries which run on OWL-S domain ontology. For guidance purpose, map fragments are stored and reused.

H. Approach by Driss & al.([37][38])

This approach relies also on maps to capture user functional and non-functional requirements. Keywords are extracted and are used to directly search WSDL description in UDDI repositories using the search engine Service-Finder. Results are classified according to non-functional attributes.

IV. DISCUSSION

A. Goals, semantic services and the matching problem

In a preliminary response to RQ1, all approaches that explicitly use the goal concept – except Driss & al. – rely on ontological representations of semantic services. Indeed, semantic service descriptions in WSMO explicitly define a sub-structure dedicated to describe the service goal. This is exploited in judicious manners in order to raise the level of abstraction for user queries. However, there are multiple categories of goals and different meanings can be associated to the goal concept [11][42]. Further, the focus is mainly on functional requirements. NFR are explicitly considered only in the approach by Driss & al. through a filtering mechanism based on Quality of Service (QoS) attribute. Matching goals expressed in (or derived from) user queries with goals defined in service ontological descriptions relies on semantic matching using semantic distance measurements techniques. The work by Klush & al. is a clear illustration of the necessity of combining semantic matching techniques with lexical techniques in order to significantly enhance discovery performance. An open question here is the extent to which semantic matching is precise and efficient for goal matching.

Another specific aspect is related to comparing and/or matching service behavior. Two services can satisfy the same goal, but can show different behavior. How to help user in selecting the right one? This requires a behavioral description of web services. Such a description could however be considered contradictory with “black box” vision of services.

B. Browsing

Although browsing is extensively used for searching and finding information on the web, it is proposed in a goal based manner by only one approach (Stollberg & al.). It relies on visualizing a hierarchical graph structure in which nodes are goal templates and edges are generalization links. Nodes are associated with available semantic web services. Navigation in the space of available services can significantly help in better understanding the problems that can be solved by them. An open question is the structure of complex goals, and how browsing can exploits different categories of links between goals. This is tightly related to goal and service composition.

C. Guidance

Two techniques to guidance during query formulation can be distinguished in this study: reuse–based and expansion–based. In the former, query elements are suggested to the user. In GGODO prototype (Gomez & al.), a goal template is retrieved from a local repository and is matched with a lightweight ontology built from user request. In Stollberg & al., concrete client requests are defined by instantiating an existing goal template with concrete inputs. And in Mirbel & al., map (i.e. query) fragments are incrementally stored in a repository and suggested later for reuse. The expansion–based technique adopts a different approach: ontological knowledge is exploited to analyze user query and to further expand it. In Zachos & al., query expansion is proposed in order to make user requirements iteratively converge towards available services.

V. RELATED WORKS

There are quite many publications that review service oriented computing research from a global perspective (e.g. [4]), or review semantic services technologies and engineering approaches (i.e. [9][10][40]). Specialized analytical reviews are however less frequent. Concerning service identification out of business process and/or system specification, [39] is a recent review of all known techniques. However, publication process is not specifically addressed, and the goal concept is absent. Concerning service description languages, a recent systematic review can be found in [41]. One of the results of this review shows that there is still a lot to do in enhancing the description of non-functional aspects (e.g. quality) of services.

VI. CONCLUSION

Matching user requirements with service capabilities is a fundamental issue in service-oriented engineering. We have in this paper analyzed a significant set of approaches to service discovery, and tried to understand if and how the goal concept is used. Our analysis shows clearly that goals are becoming an important element in approaches to service discovery, and that it can be the core building block for a fully recognized SOC engineering method. We have nonetheless pointed to three important issues which need further investigation: goal matching, browsing the goal and service spaces, and enhancing guidance techniques during query formulation. An important limitation to this work is that we have deliberately ignored service composition and how it relates to goal usage for service discovery. Future work will propose a revised version of the framework in order to handle this issue.