Using Concept Lattices to Support Service Selection

Lerina Aversano, University of Sannio, Italy
Marcello Bruno, University of Sannio, Italy
Gerardo Canfora, University of Sannio, Italy
Massimiliano Di Penta, University of Sannio, Italy
Damiano Distante, University of Sannio, Italy

ABSTRACT

When building a service-oriented system, a service integrator retrieves a set of potentially useful services from registries and then inspects their documentation to eventually decide which services to use. This task needs to be supported by proper tools that help service interface/documentation understanding, highlighting the relationships existing between the retrieved services. This paper proposes an approach, based on Formal Concept Analysis, to understand relationships between services, as well as between operations of a complex service, by analyzing service interfaces and documentation. The approach allows an analyst to cluster similar services, highlights hierarchical relationships and, in general, commonalities and differences between services. To support the proposed approach, we developed a tool that provides several service browsing capabilities. Finally, the approach has been evaluated with different case studies built upon real sets of services.

Keywords: formal concept analysis; service discovery and selection; WSDL information extraction

INTRODUCTION

The diffusion of Web services is radically changing the landscape of software engineering. It permits to realize a piece of functionality by searching, composing and executing services, while not physically integrating them in a system.

In this scenario, once requirements have been elicited, the analyst can use them as a guide to search whether there are some services able (alone or in combination) to (fully or partially) realize a particular requirement or use case (Jones, Maiden, Zachos, & Zhu, 2005). Hopefully, the query over the registries will produce a list of related services. However, to understand service features, and to select the most appropriate services for a particular need, very often the only source of documentation available is limited to the WSDL (Web Service Description Language) interface (Christensen,
Curbera, Meredith, & Weerawarana, 2001). The choice of services which are the most appropriate to the user functional and non-functional requirements could become a hard task, especially when the number of retrieved services is not small and their service interfaces complex. This introduces the need for a tool able to support the browsing and selection of services retrieved from registries. In particular, it might be useful to highlight a number of relationships that exist between any two services $s_1$ and $s_2$. For example:

1. $s_1$ and $s_2$ are completely independent, in other words, they do not share features nor information required to perform their task/produce results;
2. $s_1$ and $s_2$ share a set of features, while having each one some peculiar characteristics. For example, a service for purchasing a flight ticket and a service for purchasing a train ticket require a starting location, a destination and a travel date. However, the artifacts the two services produce are different (flight ticket vs. train ticket);
3. $s_2$ provides features additional to those provided by $s_1$. As an example, we may have two Webcam services both returning a photo of Manhattan and one of them also returning some short movies;
4. $s_2$ requires, in order to work, more information than $s_1$. For example, while a search hotel service only requires the city, the start date and the number of days of the stay, a hotel booking service also requires the credit card information;
5. $s_1$ and $s_2$ are fundamentally equivalent, i.e., one can be used in place of the other.

Similar relationships can be identified between two operations $o_1$ and $o_2$ of a complex service:

1. Operations $o_1$ and $o_2$ are completely independent, in other words, they do not share features, required inputs or produced outputs;
2. $o_1$ and $o_2$ share a set of features, inputs or outputs. However, each one has some difference from the other;
3. $o_2$ provides additional features or produce further outputs than $o_1$;
4. $o_2$ requires, in order to work, more inputs than $o_1$;
5. $o_1$ and $o_2$ are fundamentally equivalent, in other words, one can be used in place of the other.

This paper proposes the use of Formal Concept Analysis (FCA) and concept lattices to highlight the above defined relationships holding between different services or service operations. Concept lattices are built upon a context obtained from keywords extracted from service descriptions or operation parameters and are used to support the understanding of Web services through their interfaces. While similar techniques can be used to classify other types of documents, some characteristics of WSDLs, such as the use of complex types defined as XML schemas, makes the use of FCA particular useful in this context.

Overall, the main research contributions presented in this paper are the following:

- A novel approach for understanding service interfaces. The approach is based on an existing mathematical theory, FCA, and extends the preliminary work proposed in the paper (Bruno, Canfora, Di Penta, & Scognamiglio, 2005), allowing different comprehension levels;
- A tool named Service Explorer, implemented to support the service browsing and selection over concept lattices, and also providing mechanisms for query refinement; and
- The evaluation of the proposed approach through four different case studies.

The paper is organized as follows. First, we motivate and describe the proposed approach, explaining how it can be used to improve the comprehension of services from their interfaces and to support service annotation. Then, we de-
DSCWeaver: Synchronization-Constraint Aspect Extension to Procedural Process Specification Languages
www.igi-global.com/chapter/dscweaver-synchronization-constraint-aspect-extension/41538?camid=4v1a

Security for Web Services: Standards and Research Issues
www.igi-global.com/article/security-web-services/37388?camid=4v1a