Chapter 6

A Goal–Driven Approach for Service–Oriented Systems Design

Chiung-Hon Leon Lee
Nanhua University, Taiwan

Alan Liu
National Chung Cheng University, Taiwan

Arthur Shr
Louisiana State University, USA

ABSTRACT

This chapter introduces a goal-driven approach for modeling service requests in service-oriented systems. There are three challenges to this approach: how to represent a service request, how to map the request to predefined models, and how to generate a series of actions to satisfy these models. A goal model is used for user requirements elicitation, analysis, and representation. The service request from users is extracted and mapped to these goal models by a case-based method. A planning mechanism is designed to produce a sequence of activities to satisfy the extracted goals. By these methods, a more convenient and efficient service for users can be provided. A general architecture for an intention-aware service-oriented system is proposed for demonstrating how to apply the proposed approach.

INTRODUCTION

In our daily lives, services may be provided by anybody, but the quality of services provided may differ from person to person. For those who have worked together for a long time, the desire of one person does not need to be expressed clearly or precisely, because they have mutual understanding. If a boss tells a capable assistant about attending a meeting on certain day in a certain city, the assistant may use the past experience about attending a meeting to arrange the business trip that the boss desires. The business trip may need services for arranging the transportation, booking a hotel, making appointments, etc. If a service-
A goal-driven approach is widely used for software requirements elicitation and representation (Lee, Xue, & Kuo, 2001). Requirements specifications are generated by a goal-based requirements analysis process in which the functional and nonfunctional requirements will be extended with goal models. A set of computable goal models that represents the user requirements can be selected and refined as the basis of defining services. The software designers can also design related system services based on these models. The user’s vague and imprecise intention can be extracted and mapped to computer understandable and computable goal models.

Figure 2 shows a model to represent the problem discussed above. In the figure, software requirements are the source of what a user wants to use, the user needs to enter a set of clear and precise requests to get a service in return.

Several techniques and standards have been proposed to facilitate accessing web services, but it is still hard to let an ordinary user to enter a simple request and receive results directly returned from the service providers to satisfy the user’s requirements, making it a challenging problem for a web service system. The requested system may have to interact with the user to elicit more information for acquiring the suitable web services, discovering services, selecting services, composing services, processing information returned from the services, and demonstrating the results to the user. For these purposes, how to integrate service providers, system’s capabilities, and elicited user information becomes an important issue when developing an intelligent web services application. An interaction scenario among the system user, service requestor, service broker, and service provider is illustrated in Figure 1. The service requestor interacts with the user and tries to complete the user’s delegations. The service broker collects a list of web services advertisements from service providers, accepts the service request from the requestor, and tries to discover, select, and access a web service to satisfy the service request from the user.

Because the service requestor has no knowledge about the service providers, the initial query it sends to the service broker might not correspond exactly with the query input that a service provider might require in order to provide a suitable service. Such problem may be simple as a value missing for a parameter, or complicated as ambiguous keywords to process. The requestor may have to interact with the user to elicit more information for accessing the web services.

Figure 1. An interaction scenario of the user, requester, broker, and provider