# Chapter 20
# A Test-Driven Approach to Behavioral Queries for Service Selection

**Laura Zavala**
*University of Maryland Baltimore County, USA*

**Benito Mendoza**
*New York City College of Technology, USA*

**Michael N. Huhns**
*University of South Carolina, USA*

## ABSTRACT

*Although the areas of Service-Oriented Computing (SOC) and Agile and Lean Software Development (LSD) have been evolving separately in the last few years, they share several commonalities. Both are intended to exploit reusability and exhibit adaptability. SOC in particular aims to facilitate the widespread and diverse use of small, loosely coupled units of functionality, called services. Such services have a decided agility advantage, because they allow for changing a service provider at runtime without affecting any of a group of diverse and possibly anonymous consumers. Moreover, they can be composed at both development-time and run-time to produce new functionalities. Automatic service discovery and selection are key aspects for composing services dynamically. Current approaches attempting to automate discovery and selection make use of only structural and functional aspects of the services, and in many situations, this does not suffice to discriminate between functionally similar but disparate services. Service behavior is difficult to specify prior to service execution and instead is better described based on experience with the execution of the service. In this chapter, the authors present a behavioral approach to service selection and runtime adaptation that, inspired by agile software development techniques, is based on behavioral queries specified as test cases. Behavior is evaluated through the analysis of execution values of functional and non-functional parameters. In addition to behavioral selection, the authors' approach allows for real-time evaluation of non-functional quality-of-service parameters, such as response time, availability, and latency.*

DOI: 10.4018/978-1-4666-4301-7.ch020

## INTRODUCTION

A methodology for software development based on services as fundamental building blocks, known as service-oriented computing, has become widely used in building enterprise systems, because it greatly enhances their flexibility and adaptability. As a further incentive for this methodology, the number of publicly available services is continuing to increase and the Internet is becoming an open repository of such atomic heterogeneous software components. Multiple services can be integrated to facilitate cooperation between various business parties, achieve agility of the business integration, and even provide value-added services for service consumers. Essential to these capabilities is the detection of entities, services, and other resources that can be used for satisfying a specification of desired functionality. The precision of the selection process improves the possibility of having services that find, connect, and communicate with one another automatically, sharing information and performing tasks without human intervention.

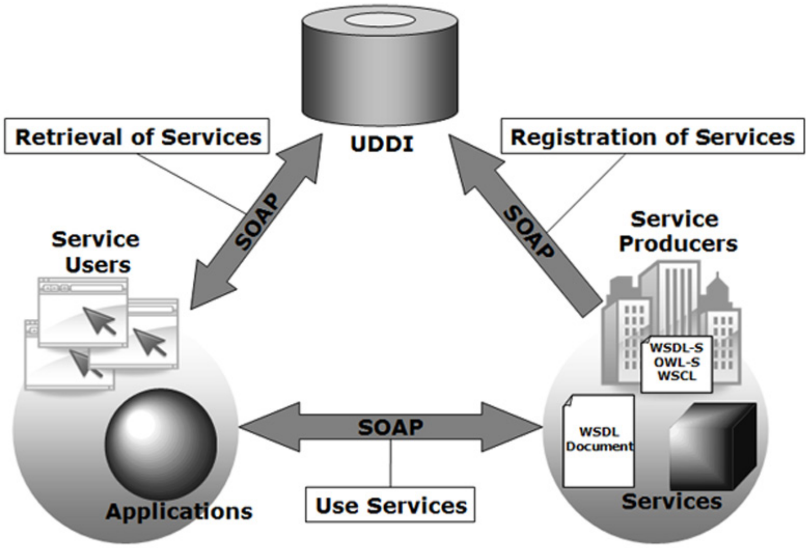
Web services are the current most promising instantiation of the service-oriented methodology. Web services comprise infrastructure for describing service structure, via WSDL (Christensen, Curbera, Meredith, & Weerawarana, 2001); specifying semantics and functionality via WSDL-S (Akkiraju et al., 2005), OWL-S (Martin et al., 2004), and WSMO (Roman et al., 2006)); supporting a service repository, via UDDI (Clement et al., 2004) or some less structured registry; interacting with services, via SOAP (Gudgin et al., 2007); and scheduling and orchestrating, via WSCL (Banerji et al., 2002) (Barry, 2003; Wombacher, Fankhauser, & Mahleko, 2004). The relationships among these components are depicted in Figure 1.

There are three orthogonal dimensions for describing a service at the knowledge level (Table 1):

- Structure,
- Function, and
- Behavior.

Current approaches for automating discovery and selection of services make use of only the first two. Syntactic and semantic search based on keywords on the structural definition of the service, usually the WSDL content, are used for

*Figure 1. The general architectural model for Web services*


UDDI
Retrieval of Services
Registration of Services
SOAP
SOAP
Service Users
Service Producers
WSDL-S
OWL-S
WSCL
SOAP
WSDL Document
Applications
Use Services
Services