An automatic subdigraph renovation plan for failure recovery of composite semantic Web services

Hadi SABOOHI (✉), Sameem ABDUL KAREEM

Department of Artificial Intelligence, Faculty of Computer Science and Information Technology, University of Malaya, Kuala Lumpur 50603, Malaysia

© Higher Education Press and Springer-Verlag Berlin Heidelberg 2013

Abstract A Web service-based system never fulfills a user’s goal unless a failure recovery approach exists. It is inevitable that several Web services may either perish or fail before or during transactions. The completion of a composite process relies on the smooth execution of all constituent Web services. A mediator acts as an intermediary between providers and consumers to monitor the execution of these services. If a service fails, the mediator has to recover the whole composite process or else jeopardize achieving the intended goals. The atomic replacement of a perished Web service usually does not apply because the process of locating a matched Web service is unreliable. Even the system cannot depend on the replacement of the dead service with a composite service. In this paper, we propose an automatic renovation plan for failure recovery of composite semantic services based on an approach of subdigraph replacement. A replacement subdigraph is posed in lieu of an original subdigraph, which includes the failed service. The replacement is done in two separate phases, offline and online, to make the recovery faster. The offline phase foresees all possible subdigraphs, pre-calculates them, and ranks several possible replacements. The online phase compensates the unwanted effects and executes the replacement subdigraph in lieu of the original subdigraph. We have evaluated our approach during an experiment and have found that we could recover more than half of the simulated failures. These achievements show a significant improvement compared to current approaches.

Keywords semantic Web service, composite services, failure recovery, subdigraph replacement

1 Introduction

Service-oriented architecture (SOA) is a collection of services which coordinates their interactions. Web service is a software system that implements a service. Web services support interactions between machines over a network, i.e., the Internet. Henceforth, we use the terms “service” and “Web service” interchangeably even though they are essentially different [1].

Web services are described using a standard machine-processable format like Web service description language (WSDL)\(^1\). The description consists of technical aspects of the definitions of what the functional properties of the service are and how a user can invoke and exploit the service.

“Semantics” annotates the Web services by some additional descriptions in a specific format that enables machines to process the Web service descriptions. These add-on descriptions come together with an ontology of the concepts which disambiguates the meaning of functional properties of the Web services such as their inputs and outputs [2].

In a service-oriented architecture, a user requests a service. A service mediator (also known as an agent) stands between the user and the providers. It finds a provider which can fulfill the user’s goal. There is no restriction on the number

Received July 30, 2012; accepted August 13, 2013

E-mail: saboohi@siswa.um.edu.my

\(^1\) http://www.w3.org/TR/wsdll20