A Dynamic and Adaptable Service Composition Architecture in the Cloud Based on a Multi-Agent System

Abdelhak Merizig, LINFI Laboratory, University of Biskra, Biskra, Algeria

Okba Kazar, LINFI Laboratory, University of Biskra, Biskra, Algeria

Maite Lopez-Sanchez, Mathematics and Computer Science Department, University of Barcelona, Barcelona, Spain

ABSTRACT

Nowadays, service composition is one of the major problems in the Cloud due to the exceptional growth in the number of services deployed by providers. Recently, atomic services have been found to be unable to deal with all client requirements. Traditional service composition gives the clients a composite service without non-functional parameters. To respond to both functional and non-functional parameters, we need a service composition. Since web services cannot communicate with each other or participate dynamically to handle changes service parameters in service composition, this issue has led us to use a dynamic entity represented by an agent based on dynamic architecture. This work proposes an agent-based architecture with a new cooperation protocol that can offer an automatic and adaptable service composition by providing a composite service with the maximum quality of service. The implementation of this model has been provided in order to evaluate the authors' system. The obtained results demonstrate the effectiveness of their proposed system.

KEYWORDS

Agent, Cloud Computing, Confidential Interval, Distributed Architecture, Dominance Relationship, Multi-Agent System, Quality of Service (QoS), Service Composition, SVM for Regression, Web Service

1. INTRODUCTION

In the recent years, cloud computing has been adopted as the most used solution by enterprises. This solution is provided as a service pooling in a private or public cloud (Yang & Wang, 2014). Moreover, cloud computing has emerged with three main service delivery models: Software as a Service (SaaS); Platform as a Service (PaaS); and Infrastructure as a Service (IaaS) (Freet et al., 2015; Garrison et al., 2015). Some well-known corporations that propose the Cloud as a solution, such as Google and Amazon web services (AWS), are now the main providers of such solutions to end users (simple clients or enterprises) with some enormous data centers situated in different geographical locations around the world. The properties and policies of Cloud encourage providers to use this technology in order to deploy and publish their services (Aceto et al., 2013; Shin, 2015). Each provider presents his web service using WSDL (Ben Seghir et al., 2015; Rahman & Meziane, 2011) by introducing

the web service's description, which includes the whole data about the deployed service to make the discovery of and differentiation between services much easier. Nevertheless, current web services cannot respond to all client requirements and the static vision presented as web services can neither work together to cooperate with each other nor interact with new data updates. An approach of service composition based on functional and non-functional parameters should lead to a solution to this issue (Upadhyaya et al., 2014). In fact, the service composition problem is classified as an NP-hard problem due to the huge number of the services deployed in the Cloud (Geyik, 2013).

In order to propose a service composition architecture, this work introduces the agent paradigm as an approach to solving the composition issue. According to the literature, the agent concept has many properties that encourage us to follow its paradigm. These properties lend a dynamic vision for an agent to be presented as a service, which shows some advantages, for example, autonomy, flexibility, parallel execution and also agent interaction (Briot, 2014; Graja et al., 2014). Assuming one service cannot meet all client requirements, we should present an architecture that can guarantee a composite service that can satisfy the end users' demands by handling all the given requirements with functional and non-functional parameters (Surianarayanan et al., 2015).

This paper uses a multi-agent system as an architecture that can handle the whole service composition process; this kind of complex systems can offer the best options for resolving the issues presented here through protocols by introducing a new agent coalition policy (Klein et al., 2012; Sless et al., 2014). Using a confidence factor and a dominance relationship gives the agents a way to select their neighbours or services that have the best QoS; these QoS influences the client's opinion about the service (Elfirdoussi et al., 2014; Ge et al., 2015), and to reduce the number of messages exchanged in the proposed system. The different types of agents (situated and mobile) gives the architecture an easy way to collect the required data from the Cloud at any location around the world (geo-distribution) (Marie, P et al., 2016). Also, this architecture facilitates treatment operations through the strong points of the Cloud (storage and calculus) (Cavet et al., 2012).

The presented work is organized as follows: Section 2 presents some related concepts used in our approach. Section 3 outlines some related work. In Section 4, we present our approach and provide the description for the proposed architecture. Section 5 explains the proposed algorithm for service composition. Section 6 shows the implementation and the results obtained for the proposed architecture. Finally, Section 7 concludes this paper.

2. BACKGROUND

2.1. Quality of Service (QoS)

In this paper, we are going to take into consideration some quality of service measures in the proposed service composition process. These QoS measures, also known as non-functional parameters, can give an evaluation of services to make the composite services more convenient for the end user. The QoS measures used in this paper are defined and formulated according to some works as follows:

1. **Cost:** It refers to the fee of accessing and using a web service that a service requester has to pay. Moreover, it depends on the number of tasks that a service user needs to execute (Karim et al., 2013; Rajeswari et al., 2014);
2. **Response Time:** The most important goal in web service delivery is to provide a service to the consumer within reasonable time. Time response is measured in terms of some sub-factors such as average response time and maximum response time promised by the service provider (Garg et al., 2013). The time response is calculated using the following formulation:

$$Response\ time = \sum_i {T_i}/{n} \tag{1}$$

Related Content

Management of Privacy and Security in Cloud Computing: Contractual Controls in Service Agreements
Deniz Tuncalp (2016). *Web-Based Services: Concepts, Methodologies, Tools, and Applications (pp. 1585-1610).*
www.igi-global.com/chapter/management-of-privacy-and-security-in-cloud-computing/140867?camid=4v1a

A Web Metadata Based-Model for Information Quality Prediction
Ricardo Barros, Geraldo Xexéo, Wallace A. Pinheiro and Jano de Souza (2008). *Handbook of Research on Web Information Systems Quality (pp. 324-343).*
www.igi-global.com/chapter/web-metadata-based-model-information/21981?camid=4v1a

Real-Time Unspecified Major Sub-Events Detection in the Twitter Data
Stream That Cause the Change in the Sentiment Score of the Targeted
Event

Ritesh Srivastava and M.P.S. Bhatia (2017). *International Journal of Information
Technology and Web Engineering (pp. 1-21).*

www.igi-global.com/article/real-time-unspecified-major-sub-events-detection-
in-the-twitter-data-stream-that-cause-the-change-in-the-sentiment-score-of-
the-targeted-event/188379?camid=4v1a

Towards a Flexible and Adaptable Modeling of Business Processes

Khadhir Bekki and Hafida Belachir (2011). *International Journal of Information
Technology and Web Engineering (pp. 57-67).*

www.igi-global.com/article/towards-flexible-adaptable-modeling-
business/55384?camid=4v1a