CTL Model Checking of Web Services Composition based on Open Workflow Nets Modeling

Zohra Sbaï, ENIT, Tunis, Tunisia
Rawand Guerfel, ENIT, Tunis, Tunisia

ABSTRACT

Web services composition (WSC) has an enormous potential for the organizations in the B2B area. In fact, different services collaborate through the exchange of messages to implement complex business processes. BPEL is one of the most used languages to develop such cooperation. However, it has been proved that its use is complex and can require some expertise in XML syntax. Even its graphical representation is not evident to handle. This is why the authors propose to model Web services using oWF-nets, a subclass of Petri nets, and then, to translate them to BPEL. Whilst, a WSC is with added value only if the involved services are compatible. So in this context, across the translation proposed the researchers develop a verification layer of the WSC compatibility. Hence, they propose a framework named D&A4WSC which allows to model the WSC by oWF-nets, to check their compatibility with the model checker NuSMV and to translate them if they are compatible in BPEL processes using the oWFN2BPEL compiler. D&A4WSC permits, furthermore, to formally analyze a BPEL process.

KEYWORDS

BPEL, Compatibility, CTL, Formal Analysis, Model Checking, Open Workflow Nets, Petri Nets, Web Services Composition

1. INTRODUCTION

To ensure a better communication, circulation and coordination of data within an enterprise, there is a need to use Information Systems. These Systems are complex in the way of their structuring and generate high costs for the enterprise. To resolve this problem, there exists conception architecture, SOA (Service Oriented Architecture), which proves to be an adequate solution. Indeed, SOA refers to a new way to manipulate, manage and integrate various application components of a computer system. It is generally implemented using Web services. Web services are applications that are exposed through standard interfaces. Interfaces are described using WSDL (Chinnici et al.) and communicate with each other through SOAP protocol. Users request for services, which are enrolled by their providers in the UDDI registry, using these interfaces. Once done, a mechanism of service discovery is required. However, sometimes this mechanism fails, despite the existence of required services. Indeed, some requirements are complex and single services cannot meet them. Therefore, a service needs to contact one or more other services and to be combined with them to get a composed one. Here, we talk about Web services composition (WSC) (Rao & Su (2005), Casati et al. (2000), Casati et al. (2001)). The WSC allows multiple services to communicate and collaborate by exchanging messages and to implement a composite service that can perform consumer complex tasks.

DOI: 10.4018/IJSSMET.2016010102

Copyright © 2016, IGI Global. Copying or distributing in print or electronic forms without written permission of IGI Global is prohibited.
In the SOA framework, process automation is a major focus, with the concepts of orchestration and composition of services. This is to centralize the logic of a process in a dedicated component, which supports the Web and business rules associated. This approach tends to reduce the impacts related to fluctuations in the process.

Several languages have been proposed to ensure this approach namely BPEL (Ma et al. (2009), Andrews et al. (2003)) which is based on the XML syntax. Indeed, BPEL is used to define the abstract and executable business processes as a set of Web services interaction coordinated recursively. However, to be composed, services should be compatible. This property ensures that we have a composite service which does not suffer from problems of deadlock. It is therefore necessary to make sure that these services will be able to interact properly, which requires a study, a clear understanding and a verification of the concept of compatibility (Sbaï & Barkaoui (2014), Guerfel et al. (2013)). In this context, the authors propose to use formal methods to model a given composition and thus to be able to verify it. In particular, they propose to use open workflow nets (oWF-nets), a subclass of Petri nets, which are characterized by their interface places that ensure the communication between the different services.

To implement their approach, the researchers propose D&A4WSC (Guerfel & Sbaï (2014)) as a framework having as a main goal the verification of WSC using formal methods to generate a valid BPEL process. More precisely, D&A4WSC allows first the modeling of a WSC using a composition of oWF-nets. Secondly, it allows the compatibility verification. To do this, the authors first check the compatibility at the level of exchanged messages between the different services, that is called syntactical verification. Then, they move to the semantic verification based on the principle of model checking, and more precisely, on the model checker NuSMV (Cimatti et al. (1999)). In fact, model checking is one of the most used formal verification methods because of its effectiveness, its ease of use and the generation of a counter example in case of error. If the compatibility is valid, then, the authors propose to generate an owfn file of the composition and to invoke the compiler oWFN2BPEL (Lohmann (2008)) which takes as input an owfn file and generates a BPEL file as output.

This paper is organized as follows: Section 2 presents preliminaries on oWFnets, compatibility and BPEL. Section 3 describes the approach of checking the compatibility of the composition of Web services and explains the translation of oWF-net to BPEL process. In Section 4, the framework D&A4WSC that implements the proposed approach is presented. To illustrate their approach, the authors expose a case study. In section 5, related works are discussed. Finally, section 6 concludes the paper and outlines future work.

2. PRELIMINARIES

This section is dedicated to present the main notions used in the rest of this work.

2.1. Open Workflow Nets

A Petri net is a formal language used to model the functionality behavior of concurrent systems. This modeling language is represented as an oriented graph which makes it readable and ensures an easy communication with different designers. Petri nets can be considered as a tool for verifying business processes since they have strong mathematical foundations. A Petri net (Barkaoui et al. (2007)) is represented by a graph containing: a set of places P, a set of transitions T and a set of arcs: F ⊆ (P × T) ∪ (T × P). Thus, a Petri net (PN) is defined by a quadruplet N= (P, T, F, W) with W is the set of weight of the arcs (x, y).

A marking of a PN at time t is M(t) = { m_1 (t); m_2 (t);..; m_n (t)} with m_i (t) is the number of tokens in place i at time t.
Dynamic Security Properties Monitoring Architecture for Cloud Computing
www.igi-global.com/chapter/dynamic-security-properties-monitoring-architecture/70037?camid=4v1a

The Compensation Benefit of ITIL® Skills and Certifications
Stuart Diaz Galup, Ronald Dattero and Jing Quan (2016). International Journal of Service Science, Management, Engineering, and Technology (pp. 1-15).
www.igi-global.com/article/the-compensation-benefit-of-itil-skills-and-certifications/149895?camid=4v1a