International Journal of Web Services Research
Volume 17 • Issue 2 • April-June 2020

Applying Formal Methods to SOA and SaaS Service Compositions for ERP Systems

Maya Souilah Benabdelhafid, LIRE Laboratory, Constantine 2 Abdelhamid Mehri University, School of Accounting and Finances, Algeria

Mahmoud Boufaida, LIRE Laboratory, Constantine 2 Abdelhamid Mehri University, Algeria

ABSTRACT

Service oriented architectures (SOA) and software as a service (SaaS) models are accelerating the use of the service concept. They have been rapidly adopted by enterprise resource planning (ERP) providers and offers to organizations modern environments favorable for making strategic decisions. However, as the SaaS concept is often confused with the SOA one, a solution is needed to clarify this problem before moving to the introduction of a framework that aggregates several services. Moreover, most of the services-based solutions rely on a particular composition and only a few are equipped with appropriate formal tools for automating the service verification. They focus on specific formal models and are based on particular properties. This article proposes a formal framework that reveals a set of general steps starting from defining the service composition before shifting to the use of the chosen formal methods so as to verify the ERP system properties.

KEYWORDS

Enterprise Resource Planning, Formal Methods, Orchestration, Service Oriented Architecture, Software as a Service

INTRODUCTION

Enterprise Resource Planning (ERP) (Soobrayen et al., 2019) is provided as a program with a set of functionalities that are run on a single database. This centralized system gives access to improved accurate information within shorter periods of time, which helps the managers make strategic decisions. It collects, records, integrates, manages and delivers data across all the organization departments by including different functional modules that reflect the departmental structure of an organization (accounting, sales, production, and so on). ERP systems have been implemented 'on-premise' as products bought by customers. The last couple of years, they have tremendously evolved and upgraded processes to enhance their provided modules and increase their integration capabilities. ERP manufacturers are developing different modules to cover and support the functional units of organizations. Nevertheless, successfully implementing such 'on-premise' ERP is complex, costly and not flexible enough.

Service based ERP systems are increasingly interesting researchers in the academic and industry worlds. Different solutions are being suggested in the cloud computing that provides a Software as a Service (SaaS) (Johansson & Ruivo, 2013). A SaaS based ERP solution is one which provides enhanced data storage capacity, security and control. It gives a real-time access with more mobility. Consequently, the organizations do not have to make expensive upfront investments in IT hardware and servers, nor significant personnel resources to managing it. They are usually fast and easy to install

DOI: 10.4018/IJWSR.2020040101

than on-premises deployment. Another kind of solutions make use of Service Oriented Architecture (SOA) given in the service oriented computing (SOC) by supporting service composition solutions to construct a flexible system. SOA is a conceptual tool that works as a facilitator to any ERP module linking multiple functions together using re-usability to meet customer demand quickly.

The best-known enabler supporting both SaaS and SOA are Web services which are related to technical capabilities that allow systems to connect with one another through standardized interfaces. A survey on the issues that need to be considered when launching ERPs can be found in (Sørheller et al., 2018).

Considering ERP modules as services is interesting, but complicated. The composition still has different challenges related to service heterogeneity for which many efforts have been made using formal verifications. Up to now, there is not a comprehensive methodology on this topic. This paper proposes a framework, which discusses the steps that designers are supposed to follow in order to create modern ERP systems. This verification can be efficient by using formal methods that promote the development of well-founded solutions and their automation. The theories are reused and the ERP approach gained tool-supports. The remainder of this paper is structured as follows. First, the literature is reviewed. Then, SaaS and SOA concepts are explained and compared for preparing the reader to the raised problem. Next, a comprehensive description related to the framework is presented. After that, initial results conducted in the healthcare sector and quantifying some of the gains delivered by the framework are revealed. Finally, some concluding remarks and perspectives are presented.

LITERATURE REVIEW

According to some recent review papers, SOA is adopted by different research works in order to develop ERP systems (Habadi et al., 2017, Holmberg & Johansson, 2017) while others have chosen SaaS models (Seethamraju, 2015, Elmonem et al., 2016, Desai et al., 2018). (Chen et al., 2015) defined a cloud ERP on which enterprise customers can select services and customize an ERP to meet their specific needs. The authors proposed a Web-based solution that can be used to automate the ERP customization process.

Furthermore, most of the researchers worked on SaaS or SOA concepts separately. A few of them proposed their fusion. For instance, (Nassif & Capretz, 2010) proposed a model for illustrating the way in which SaaS applications can be offered as SOA services. A SaaS application that is required for migration into SOA is located. A thorough investigation is performed to study its maturity level. SaaS features are identified and converted to Web services that are modified to conform to SOA principles. The resulting SaaS-SOA services are prepared for publication. Based on this fusion vision, (Jin et al., 2013) proposed a model with an architecture including several steps. A study was carried out aiming to develop a mold industry by using a SaaS provider to purchase software needs.

Nevertheless, the above research works concentrated on the SOA and SaaS fusion, but did not explain how the SaaS platforms, once turned into SOA, could be composed. According to (Jula et al., 2014), the dynamic composition is one of the best approaches that has been proposed by researchers and applied by cloud providers. For the efficient handling of failures, the service composition verification is of a great importance since it can guarantee the robustness and high quality of Web based software. From the literature, there are some surveys that review formal service composition approaches. For instance, (Souri et al., 2018a) ((Souri et al., 2018b), respectively) present a rich survey in the SOC domain (in the cloud computing, respectively). They provide systematic literature reviews to examine the current technical studies and categorize them in three fields: specification and process algebra, model checking, and theorem proving. For instance, (Cheng et al., 2015) present an automatic Web service composition that deals with both input/output compatibility and behavioral constraint compatibility of fuzzy semantic services. The user input and output requirements are modeled as a set of facts and a goal statement in the horn clauses. The composition problem is transformed into a horn clause logic reasoning problem. A fuzzy predicate Petri Nets (PN) is applied to model the horn clause set, and t-invariant technique is used to determine the existence of composite services fulfilling