The Journal of Systems and Software 125 (2017) 152–169

Contents lists available at ScienceDirect

The Journal of Systems and Software

journal homepage: www.elsevier.com/locate/jss

Software component and the semantic Web: An in-depth content analysis and integration history

Loveleen Kaur*, Ashutosh Mishra

Department of Computer Science and Engineering, Thapar University, Patiala, India

ARTICLE INFO

Article history:
Received 21 April 2016
Revised 15 November 2016
Accepted 18 November 2016
Available online 29 November 2016

Keywords:
Component-based software engineering
Semantic Web
Ontology
Reasoners
Web services
Linked Data

ABSTRACT

With the advent of Component-based software engineering (CBSE), large software systems are being built by integrating pre-built software components. The Semantic Web in association with CBSE has shown to offer powerful representation facilities and reasoning techniques to enhance and support querying, reasoning, discovery, etc. of software components. The goal of this paper is to research the applicability of Semantic Web technologies in performing the various tasks of CBSE and review the experimental results of the same in an easy and effective manner. To the best of our knowledge, this is the first study which provides an extensive review of the application of Semantic Web in CBSE from different perspectives. A systematic literature review of the Semantic Web approaches, employed for use in CBSE, reported from 2001 until 2015, is conducted in this research article. Empirical results have been drawn through the question-answer based analysis of the research, which clearly tells the year wise trend of the research articles, with the possible justification of the usage of Semantic Web technology and tools for a particular phase of CBSE. To conclude, gaps in the current research and potential future prospects have been discussed.

1. Introduction

Component-based software engineering (CBSE) (Cai et al., 2000) has been a transition from the conventional development process to promote software development in an economical, effective and faster manner through the reuse and integration of software packages named as component or COTS (Commercial Off-The-Shelf). Ample amount of research has been done on the various aspects, phases, and characteristics of the CBSE process since its provenance in the industry. This is because of the numerous advantages (Debayan, 2011) that it has over the traditional methods of software development like reusability, replaceability, extensibility, context independence, minimal inter-component dependency, etc. all of which lead to the increased productivity and decreased cost of development.

Even though component development has vast benefits, component-based systems have been generally afflicted with different issues concerning system integration and composition, due to which it has not been able to achieve its full potential:

- Firstly, there is no clear protocol to specify a third-party component's description, configuration, integration, and modification, so that it can be accommodated within distributed development environments or within third-party system requirements (Crnkovic, 2001).
- Secondly, due to the absence of proper query methods and techniques, off-the-shelf component integration also involves the problem of locating and retrieving reusable components which fulfil the user's requirements (Vitharana, 2003).
- Lastly, there have been methods discussed in the literature that automatically build composite components. However, when there is an absence of pre-conditions and effects in the developer's requirements, the traditional techniques may generate composite components which contradict to the developer's needs (Rosa and Lucena Jr, 2011).

Thus, component-based development has still not been able to acquire its full capability as a result of these few major hurdles. A suitable solution to this lies in Semantic Web technologies (Cardoso, 2007) which "help machines comprehend additional information located on the Web to make them support automation of tasks and richer data discovery, data navigation and integration" . In other words, Semantic Web contains the ability to methodically express the intended semantics and aids in automated reasoning, supporting, sharing, integration and management of information from heterogeneous sources (Cardoso and Sheth, 2006). These capabilities of Semantic Web perfectly satisfy all those general requirements of exploring, retrieving and describing the source-code

* Corresponding author.
E-mail addresses: loveleen.kaur@thapar.edu (L. Kaur), ashutosh.mishra@thapar.edu (A. Mishra).

http://dx.doi.org/10.1016/j.jss.2016.11.028

Table 1
Research questions for literature evaluation.

Research question	Main motivation
RQ1:What has been the year wise status of publications since the inception of the amalgamation of Semantic Web and CBSE ?	Identify the time frame in which most relevant and large number of research articles have been published.
RQ2:Which aspect of CBSE has been most catered to by Semantic Web ?	Identify the focus of the Semantic Web technologies in CBSE process.
RQ3a:What Semantic Web technique has been used the most?	Identify the most popular technique of Semantic Web for each CBSE task catered to.
RQ3b:What are the frequently used tools and languages for the techniques in RQ3a?	Frequently used languages and tools are suggestive of ease of use and a larger application domain.
RQ4: What have been the research gaps identified from the literature survey?	To identify significant research prospects that have not been explored yet.

component relationships in large object-oriented reuse frameworks or libraries for the efficient working of the component-based development process.

This Systematic Literature Review (SLR) aims to describe the latest state-of-the-art Semantic Web technologies employed to ease and improve the process of CBSE. The systematic review conducted in this research work is performed in accordance with the guidelines proposed by Kitchenham, 2004. To identify the most important studies in accordance with relevance and quality, we have performed the search in four primary digital libraries. We posed certain research questions, stated in Section 3, which supported us to gather the essential information from the research articles in our review process.

The discussion proceeds as follows: the background of Semantic Web technologies is described in Section 2. Section 3 analyses the research method undertaken to perform the review process, stating the research questions formulated. Section 4 researches and reviews the status of Semantic Web in CBSE and explains the papers investigated in this study in a clear and concise tabular format. Section 5 discusses the findings of this paper in terms of results and attempts to answer the research questions stated in Table 1 of Section 3. Finally, Section 6 concludes the paper, stating the additions which could be made to the current scenario of research related to the area of concern.

2. Semantic Web: a brief background check

A brief overview of the basics of Semantic Web is presented in this section. Semantic Web was introduced by Tim Berners-Lee and was made popular with the help of the World Wide Web consortium (W3C). It was due to Semantic Web that computers could deal with the information on WWW, understand and couple it, to help humans find required knowledge.

This study does not attempt to reproduce precursory studies about the progression of Semantic Web. Nevertheless, in order to promote comprehension, only the key concepts of Semantic Web are reviewed in this section, that are relevant to the CBSE domain and the literature studied in this research article.

2.1. Ontology

Often considered as the premise of the Semantic Web, an ontology (Guarino, 1998) can be interpreted as a collection of logical axioms constructed to explain the intended meaning of a vocabulary, to express something significant within a particular area of interest and promote sharing of information (Gruber, 1991). XML (Wang et al., 2005) is the commonly used format for data and documents on the Web. Also proposed by W3C, is the Resource Description Framework(RDF) (Quan et al., 2007), a data model based on the concepts of graph which employs URIs and has several different syntax, including XML.

2.1.1. Ontological languages

Ontology makes use of ontological languages (Sireteanu, 2013) like OIL (Ontology Inference Layer), DAML+OIL, CycL, OWL(Web Ontology Language), etc. for representation of knowledge. Among all the ontological languages presented, OWL (McGuinness et al., 2004) has been the most popular one. OWL provides an additional vocabulary with the help of formal representation of semantic. This makes it support greater interpretability of the content on the Web by the machine than that provided by RDF, XML and RDF Schema (RDF-S). It inherits the advantages of DAML+OIL and provides the same level of reasoning and sharing capabilities. Contrary to other languages, OWL also provides built-in versioning functionalities. It supports a rich expressive power and achieves scalability due to a layered architecture.

2.1.2. Ontology editing tools

Ontology employs a variety of tools (Kapoor and Sharma, 2010) for ontology editing and development. However, Protégé (Noy et al., 2000), an integrated and platform-independent system for maintenance and development of knowledge-based systems, has been used repeatedly for ontology editing. Protégé consists of a frame-based knowledge model, ensuring compatibility with OKBC (the Open Knowledge-Base Connectivity protocol) which enables interoperability with other knowledge-representation systems.

2.1.3. Reasoners

Reasoners determine the quality and correctness of an ontology. High quality and correct ontology is devoid of inconsistency and uncertainty. Out of all the ontology reasoners, RACER (Haarslev and Möller, 2001), the first OWL reasoner to be developed; Pellet (Parsia and Sirin, 2004), an OWL-DL reasoner based on Java; FACT ++ (Tsarkov and Horrocks, 2006) (an updated version of FAst Classification of Terminologies) which supports OWL and uses the principles of description logic; and OWLJessKB, a description logic reasoner for OWL, have gained larger acceptance. On the basis of ontology languages and reasoners described above, many query systems and languages have already been developed. nRQL is the query language for RACER. SPARQL (Sirin and Parsia, 2007) is an RDF-based query language which uses the SQL syntax.

When ontologies are employed in collaboration with Semantic Web rules, SQWRL (O' Connor and Das, 2009) is employed as the query engine.

2.2. Semantic Web services

Semantic Web services (Fensel and Bussler, 2002a) are used for machine-to-machine communication through the World Wide Web, much like the usual Web services. Semantic services employ markup that renders data machine readable in a sophisticated and detailed way, making them a vital component of the Semantic Web. The fundamental idea behind Web services is rapid software formation with the help of reusable software components which enables the delivery and payment of software as services instead of packaged products. This proves to be extremely useful for the decentralization and distribution of business services.

Semantic Web services employ the following technology (Nacer and Aissani, 2014) and languages for its functioning: Universal Description, Discovery, and Integration (UDDI) is a technique to en-