

A MDSD Approach for Adaptable Service Oriented Systems based on Domain Specific Language Engineering

Mohammed Lethrech, ENSIAS, Mohammed V University in Rabat, Rabat, Morocco

Adil Kenzi, Sidi Mohamed Ben Abdellah University, Fes, Morocco

Issam Elmagrouni, ENSIAS, Mohammed V University in Rabat, Rabat, Morocco

Mahmoud Nassar, ENSIAS, Mohammed V University in Rabat, Rabat, Morocco

Abdelaziz Kriouile, ENSIAS, Mohammed V University in Rabat, Rabat, Morocco

ABSTRACT

The simultaneous use of the new computing paradigms: Domain Specific Modeling, Context Oriented Computing and Service Oriented Computing, raises many challenges. Particularly, the challenge of engineering such systems, which consists of the definition of modeling approaches, processes, techniques and tools to facilitate their construction. The proposed MDSD approach for context-aware service oriented systems is based on Domain Specific Language Engineering. The Context-Aware, Domain Specific and Service Oriented (CADSSO) development approach is based on five models. The first one is domain specific context model; it symbolizes the services context of use. The second is domain specific services model; it takes care of services modeling. The third is service variability model; it formulates services variants. The fourth is adaptation rules model, which is the joint between service variability model and context model. The fifth is business rules model, used to model domain business. A code generator is in charge of the transformation of the five models to the final code.

KEYWORDS

Adaptation, Context Aware Computing (CAC), Domain Specific Language (DSL), Domain Specific Modeling (DSM), Model Driven Software Development (MDSD), Service Oriented Computing (SOC), Variability

1. INTRODUCTION

Communication and integration capacities, reactivity to business and technological changes, ease of use and adaptation of information systems, constitute the main challenges of enterprise competitiveness (Hirschfeld, Costanza, & Nierstrasz, 2008; Bettini et al., 2010; Strang, & Linnhoff-Popien, 2004; Boukadi, 2009). In order to meet these needs, software engineering did not stop providing new approaches and techniques. Currently, software engineering is particularly marked by the emergence of the following paradigms.

Adoption of Domain Specific Modeling (DSM) approach in software engineering provides a lot of advantages, essentially a better reactivity to business rules and technological changes in addition to a high productivity and an excellent quality of a generated code (Lethrech, Elmagrouni, Kenzi, Nassar, & Kriouile, 2012; Kelly & Tolvanen, 2008; Kelly & Pohjonen, 2009). DSM approach is mainly based on two principles. Firstly, elevation of abstraction level by modeling the solution with a Domain Specific Language (DSL); the DSL uses directly the concepts and business rules of a specific domain. Secondly, the full automatic generation of the final solution from the high level specifications.

Nowadays, with mobile devices propagation, new software engineering needs appeared, especially the challenge of context awareness (Madkour, El Ghanami, Maach, & Hasbi, 2013). In this direction, the ability of an application to change its behaviour according to the context of execution is a key characteristic (Kazhamiakin et al., 2010). Context Aware Computing (CAC) has recently emerged as a new computing paradigm promising adaptable systems development. It provides techniques for developing pervasive computing applications that are flexible, adaptable and can react to context changes (Bettini et al., 2010).

On the other hand, since the appearance of the software engineering discipline, the main concerns of researchers in this area are simplification of production, reuse and communication of software applications. The paradigm Service Oriented Computing (SOC) represents a promising solution to these preoccupations (Davis, 2009). SOC uses Service Oriented Architecture (SOA) which is an architectural style of building software applications that promotes loose coupling of independent services (Kumar, Narayan, & Ng, 2009).

The simultaneous use of these approaches raises many challenges. Particularly, the challenge of engineering such systems (Papazoglou, Traverso, Dustdar, & Leymann, 2008) which consists of the definition of modeling approaches, processes, techniques and tools to facilitate construction of these systems along with costs and time optimization.

In this paper, the researchers propose a Context Aware, Domain Specific and Service Oriented (CADSSO) Model Driven Software Development approach for CADSSO systems. The proposed approach is based on five models. The first one is the domain specific services model; it takes care of the services modeling. The second is services variability model; it formulates services variants. The third is the domain specific context model, which is responsible for modeling the system context of use. The fourth is adaptation rules model; it makes the join between services variability model and domain specific context model. The last one is domain specific business rules model, which is in charge of the domain specific business rules modeling. As we are thinking DSM, the language developer must produce the domain specific service meta-model as an extension of our generic service meta-model (abstract syntax), the language (concrete syntax), the domain framework and the code generator. The latter is in charge of the transformation of the five models to final source code.

This paper is organized as follows; the second section presents the related work, a DSL engineering process is presented in the third section, the fourth is an overview of our CADSSO MDSD approach, the CADSSO modeling stage and the code generation stage are respectively illustrated in the fifth and sixth sections. The proposed approach is evaluated in the seventh section using a tax case study. Finally, the article ends with a conclusion and an outlook.

2. RELATED WORK

In (Lethrech et al., 2012) a lot of SOA DSLs have been studied. These later address many concerns: security, orchestration, quality of service (QoS), etc. The main conclusion of this comparative study is no one of the studied SOA DSLs deals with adaptation.

Parra et al. (2009, 2012) that are part of the ADAM team propose a homogeneous Context-Aware Dynamic Service-Oriented Product Line (DSOPL) named CAPucine. Their goal is to define at the same time a service-oriented and context-aware Product Derivation (PD) that monitors the context evolution in order to dynamically integrate the appropriate assets in a running system. Their target platforms follow the service-oriented approach. They introduced the concept of context-aware assets, the assets that can be integrated at runtime, depending on the environment state. Context-aware asset is composed of several clauses. Each clause is composed of one test and one body. The clause test is a classical condition expressed on an observable that is an object abstracting a state of the environment. They treat the architecture kind of adaptation. Adaptation of the elements service is out of the scope of their work.

23 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage: www.igi-global.com/article/a-mdsd-approach-for-adaptable-service-oriented-systems-based-on-domain-specific-language-engineering/169721

Related Content

Ranking Barriers of Supply Chain Management by MCDM Method During Disaster Management: A Case Study of India

Sourav Roy, Avik Paul, Ashish Paul, Shresth Kashyap and Arunabha Jana (2021). *International Journal of System Dynamics Applications* (pp. 1-16). www.irma-international.org/article/ranking-barriers-of-supply-chain-management-by-mcdm-method-during-disaster-management/273176

An Approach in Big Data Analytics to improve the velocity of unstructured data Using Map Reduce: An Approach to improve the velocity of data

(2021). *International Journal of System Dynamics Applications* (pp. 0-0). www.irma-international.org/article//272245

Reflections

Lars Taxén (2010). *Using Activity Domain Theory for Managing Complex Systems* (pp. 30-52). www.irma-international.org/chapter/reflections/39670

A Comparative Study of Metaheuristic Methods for Transmission Network Expansion Planning

Ashu R. Verma, P. K. Bijwe and B. Panigrahi (2010). *International Journal of Applied Evolutionary Computation* (pp. 71-91). www.irma-international.org/article/comparative-study-metaheuristic-methods-transmission/49137

Reality Classification System: A Product Line of the EIS UFO

Azamat Abdoullaev (2008). *Reality, Universal Ontology and Knowledge Systems: Toward the Intelligent World* (pp. 305-320). www.irma-international.org/chapter/reality-classification-system/28320