Journal of Universal Computer Science, vol. 20, no. 5 (2014), 583-586
submitted: 15/4/14, accepted: 25/4/14, appeared: 1/5/14 © J.UCS

Software Components, Architectures and Reuse: Software Product Line Engineering and Source Code Enhancements

J.UCS Special Issue

Marcelo Fantinato
(USP – University of São Paulo, Brazil
m.fantinato@usp.br)

Uirá Kulesza
(UFRN – Federal University of Rio Grande do Norte, Brazil
uira@dimap.ufrn.br)

Flavio Oquendo
(IRISA – University of South Brittany, France
flavio.oquendo@irisa.fr)

The aim of this Special Issue is to report the state of research and practice on the theme of Software Components, Architectures and Reuse – Software Product Line Engineering and Source Code Enhancements. This special issue is comprised of selected papers drawn from submissions from an open international Call for Papers and extended peer-reviewed versions of the best papers presented at the 6th Brazilian Symposium on Software Components, Architectures and Reuse (SBCARS 2012), held in Natal, RN, Brazil, September 24-25, 2012 (http://www.cbsoft.dimap.ufrn.br/sbcars_apresentacao.php?lang=en) and at the 7th Brazilian Symposium on Software Components, Architectures and Reuse (SBCARS 2013), held in Brasília, DF, Brazil, September 30 and October 1, 2013 (http://cbsoft2013.unb.br/en/sbcars-en).

The call for this Special Issue received 27 submissions. The submissions originated from co-authors of 13 countries (Austria, Brazil, Chile, China, France, Germany, Iran, Japan, Portugal, Republic of Korea, Spain, Tunisia, and USA).

Each submission was reviewed by at least three reviewers. The reviewing process was organized in two phases. First submissions were selected to provide revised versions based on the reviewers' recommendations or definitively rejected. The revised versions were then checked by the reviewers, and a second selection carried out out of which nine high-quality papers were finally accepted to be included in the special issue. In total, 75 reviewers participated in the process.

The special issue is composed by nine accepted papers and presents high-quality research carried out by co-authors from Austria, Brazil, France, Portugal, Spain and USA.

Contents of this Special Issue

The first paper, entitled “A Toolset for Checking SPL Refinements” (F. Ferreira, R. Gheyi, P. Borba, G. Soares) proposes and implements four tools for software product line (SPL) refinements that allow checking if refinement transformations preserve the behavior of the original SPL products. The tools are founded on a formal notion of SPL refinement, which guarantees that the observable behavior of products in the original SPL is preserved by corresponding products in the evolved SPL. To evaluate the proposed tools, the authors analyze 35 evolution scenarios of an SPL with 32 KLOC and compare the approaches with respect soundness, performance, and code coverage.

The second paper, entitled “Comparing Two Black-box Testing Strategies for Software Product Lines” (P. Accioly, P. Borba, R. Bonifácio) presents two controlled experiments conducted to assessing two different approaches for software product lines testing from the perspective of productivity and quality of the testing execution activities. The two evaluated testing approaches are: the Generic Technique (GT) that uses general test suites specifications without variability representation; and the Specific Technique (ST) that adopts product customized test suites. The results of both controlled experiments show that ST can improve the test execution process productivity by reducing test execution time and invalid change request rates.

The third paper, entitled “Consistency Checking in Early Software Product Line Specifications – The VCC Approach” (M. Alférez, R. E. Lopez-Herrejón, A. Moreira, V. Amaral, A. Egyed) addresses the challenge of checking, in a Software Product Line (SPL), that different models used in early SPL specification do not contain inconsistent information that may be propagated and generate inconsistent products that do not conform to its requirements. The approach proposed by these authors, called Variability Consistency Checking (VCC), relates information inferred from the relationships between features and from base models related to those features. Validating if all products in an SPL satisfy user-defined consistency constraints is based on searching for a satisfying assignment of each of the formulas generated by VCC. Results of case studies to validate VCC are presented in their paper.

The fourth paper, entitled “Defining and Validating a Feature-Driven Requirements Engineering Approach” (R. P. Oliveira, D. Blanes, J. Gonzalez-Huerta, E. Insfran, S. Abrahão, S. Cohen, E. S. Almeida) describes the Feature-Driven Requirements Engineering (FeDRE) approach, which provides support to the requirements engineering of software product lines. The approach focuses mainly on the specification of requirements at early stages of domain engineering, taking as input the scoping artifacts. The authors also present a case study involving the development of a mobile application for emergency notifications using the FeDRE approach. The study results show that the analysts perceived the approach as easy to use and useful for specifying the functional requirements for this particular SPL.

The fifth paper, entitled “Flexible Feature Binding with AspectJ-based Idioms” (R. Andrade, H. Rebêlo, M. Ribeiro, P. Borba) proposes AspectJ-based idioms to implement flexible feature binding, which address existing design deficiencies of recent research work. To evaluate their idioms, the authors present an empirical study that applies the proposed idioms to the extracted code of 16 features of five different software product lines. The study then compares those idioms through of: (i) a quantitative analysis using cloning, scattering, and tangling metrics; and (ii) a qualitative analysis discussing the code reusability, changeability, and instrumentation overhead. The results of the study show that one of the proposed idioms brings advantages with respect to both quantitative and qualitative assessments.

The sixth paper, entitled "Verification of Software Product Line Artefacts: A Checklist to Support Feature Model Inspections" (R. M. Mello, E. N. Teixeira, M. Schots, C. M. L. Werner, G. H. Travassos) presents the results of a quasi-systematic review of the technical literature that point to a lack of techniques to support the inspection of Software Product Line artifacts, including the feature models that are largely used in domain modeling. As a result, these authors developed a checklist-based inspection technique, called FMCheck, to support the detection of defects on feature models. FMCheck is configurable and applicable to different feature model notations. In their paper, authors present results of an empirical evaluation, comparing FMCheck to ad-hoc techniques.

The seventh paper, entitled "A Catalogue of Refactorings to Remove Incomplete Annotations" (F. Medeiros, M. Ribeiro, R. Gheyi, B. Fonseca) addresses the needs to efficiently remove incomplete annotations used by developers. In this paper, the authors propose a catalogue of refactorings that converts incomplete annotations into complete ones without cloning code and hence without increasing Lines of Code (LOC), differently from other existing solutions. They implemented an Eclipse plug-in to support the proposed approach. To evaluate the proposed catalogue, the authors performed a study to analyze questions related to code cloning, LOC, and number of directives. And, to answer their research questions, they analyze releases of 12 program families of different domains ranging from 4.9 thousand to 1.5 million LOC. The results of this study are presented in their paper.

The eighth paper, entitled "Thesaurus-Based Tag Clouds for Test-Driven Code Search" (O. A. L. Lemos, A. C. Paula, G. Konishi, S. Bajracharya, J. Ossher, C. Lopes) proposes the adoption of thesaurus-based tag clouds to improve test-driven code search, which consists on the usage of code search and reuse that makes use of more semantic information available on test cases. The thesaurus-based tag clouds allow showing developers terms that are more frequently used in the code repository to improve their search. Terms are generated by looking up words similar to the initial keywords on a thesaurus. Tag clouds are then formed based on the frequency in which these terms appear in the code base. The authors have implemented the approach as an extension to CodeGenie – a Java-based test-driven code search tool. The paper also presents the approach evaluation conducted through an applicability study and a controlled experiment, which bring evidences of the approach's benefits.

The ninth paper, entitled "What should I code now?" (L. L. N. Silva Jr, A. Plastino, L. G. P. Murta) addresses the relevant knowledge that can be extracted from the huge amount of data related to the documentation and to the source code during software development. In their paper, they present a new approach for code completion based on sequential patterns mined from previous developed source code. Using data mining as tool, their approach can consider what is being coded to provide suggestions of new code sequences based on the mined patterns. The proposed approach was implemented as a plug-in for the Eclipse IDE, named Vertical Code Completion, and applied over widely known open source systems. Their paper presents the results of these studies.

Reviewers of the special issue

Adenilso Simao, Brazil	Jobson Massollar, Brazil
Alberto Costa Neto, Brazil	Johannes Koskinen, Finland
Alexandre Alvaro, Brazil	John Grundy, Australia
Alexandre Correa, Brazil	Jose Miguel Horcas, Spain
Aline Vasconcelos, Brazil	José Pérez-Alcazar, Brazil
Andre Endo, Brazil	Kecia Ferreira, Brazil
Andres Diaz-Pace, Argentina	Kiev Gama, Brazil
Antonio Francisco Prado, Brazil	Luciano Digiampietri, Brazil
Antony Tang, Australia	Marcelo Eler, Brazil
Bedir Tekinerdogan, Turkey	Márcio Barros, Brazil
Camila Nunes, Brazil	Marco Antonio Pereira Araujo, Brazil
Carlos Cuesta, Spain	Marco Tulio Valente, Brazil
Carlos da Silva, Brazil	Martin Becker, Germany
Carlos Ramón López Paz, Cuba	Maurizio Morisio, Italy
Cecilia Rubira, Brazil	Mehdi Mirakhorli, USA
Claudia Werner, Brazil	Nelio Cacho, Brazil
Cláudio Sant`Anna, Brazil	Nelson Rosa, Brazil
Daniel Lucrédio, Brazil	Oliver Hummel, Germany
Danny Weyns, Belgium	Padraig O'Leary, Ireland
Darko Huljenic, Croatia	Paris Avgeriou, The Netherlands
Edson Oliveira Junior, Brazil	Patricia Machado, Brazil
Eduardo Almeida, Brazil	Paulo Pires, Brazil
Elder Cirilo, Brazil	Raffaela Mirandola, Italy
Elisa Yumi Nakagawa, Brazil	Regina Braga, Brazil
Ellen Francine Barbosa, Brazil	Rick Rabiser, Austria
Fernando Castor, Brazil	Roberta Coelho, Brazil
Fernando Figueira Filho, Brazil	Roberto Bittencourt, Brazil
Flavia Delicato, Brazil	Roberto dos Santos Rocha, Brazil
Francisco Dantas, Brazil	Rodrigo Bonifacio, Brazil
Frederico Durao, Brazil	Rogerio de Lemos, Great Britain
Frederico Lopes, Brazil	Rosana Braga, Brazil
Genaina Rodrigues, Brazil	Sarajane Peres, Brazil
Gibeon Soares de Aquino Junior, Brazil	Tiago da Silva, Brazil
Gledson Elias, Brazil	Uwe Zdun, Austria
Henry Muccini, Italy	Vander Alves, Brazil
Ingrid Nunes, Brazil	Vanilson Buregio, Brazil
Isela Macía, Brazil	Vinicius Garcia, Brazil
Ivica Crnkovic, Sweden	

Marcelo Fantinato
Uirá Kulesza
Flavio Oquendo
Brazil/France, April 2014