JOT
JOURNAL OF
OBJECT TECHNOLOGY

Journal of Object Technology | EDITORIAL

Introduction to the Special Theme on Models and Evolution

Ludovico Iovino[*] **and Dalila Tamzalit**[†]
[*]Gran Sasso Science Institute, Italy
[†]University of Nantes, France

ABSTRACT In general, software artefacts and applications are subject to many kinds of changes at all levels, from requirements through architecture and design, to source code, documentation and test suites. They typically affect various kinds of models, and for this reason managing the evolution is therefore an essential discipline. This JOT theme issue takes directly his roots from the international workshop dedicated to the topic: “Models and Evolution” proposed to the ACM/IEEE Models conference since 2009 until now.

KEYWORDS MDE, Models and Evolution

1. Introduction

In general, software artefacts and applications are subject to many kinds of changes, which range from technical changes due to rapidly evolving technology platforms, to modifications in the applications themselves due to the natural evolution of the businesses supported by those software applications. These modifications include changes at all levels, from requirements through architecture and design, to source code, documentation and test suites. They typically affect various kinds of models (data, behaviour, domain, requirements…). Coping with and managing changes that accompany the evolution of software models is therefore an essential discipline of Software Engineering.

2. Models and Evolution: great challenges

Software artefacts constantly increase in complexity, variety and novelty. Environment and business constraints, user requirements and new insights put additional pressure on their adaptability, availability, reliability and quality: they continuously need to be up to date. But evolution issues are critical, complex and costly to manage. They concern requirements, architecture, design, source code, documentation, integration or deployment. They also typically affect various kinds of models (data, behavioural, domain, source code or goal models). Addressing and managing these varieties of changes is essential. Models and meta-models, the cornerstone of complex software systems’ abstractions, represent a powerful mean for facing software evolution challenges by ensuring a more abstract and expressive modeling of software evolution. They can help and guide software evolution and can enforce and reduce critical risks and important involved resources. This special theme puts the focus on Models and Evolution by considering two main sides:

1. Managing software evolution needs by relying on the high-level abstraction power of models and meta-models.

2. Managing model and metamodel evolution needs and the co-evolution of all related software artefacts by putting attention to their increasing evolution issues as they become primary artefacts.

3. Presentation of the special theme

This theme issue takes directly his roots from the international workshop dedicated to the topic: “Models and Evolution” (http://www.models-and-evolution.com). The workshop has first been the fusion of two different workshops (MoDSE and MCCM) and has been proposed to the ACM/IEEE Models conference since 2009 until now. ME is now a well-established

JOT reference format:
Ludovico Iovino and Dalila Tamzalit. *Introduction to the Special Theme on Models and Evolution*. Journal of Object Technology. Vol. 20, No. 1, 2021.

http://dx.doi.org/10.5381/jot.2021.20.1.e1

with IEEE/ACM Models conference and is in its fourteenth edition. This success shows the growing importance of issues around Models and Evolution, encouraging us to enforce it by proposing this dedicated theme issue. We received 6 intent to submit, 4 papers were submitted and, among them, 2 papers have been accepted for publication:

- *Patrick Stünkel, Harald König, Adrian Rutle and Yngve Lamo*, in their paper "Multi-Model Evolution through Model Repair", address the context of design and development of complex software systems that usually comprises multiple inter-related models, i.e. abstract representations of certain aspects of the underlying system. The relations between these models induce global consistency conditions which the models collectively must fulfil. At the same time, these models are subject to frequent changes, and as a result, maintaining their global consistency over time becomes an important issue in model management in general and Model-Driven Software Engineering in particular. They present a comprehensive feature model providing an overview of the current state of the art of model management. In this feature model, they further identify the central role of model repair as an implementation pattern for (multi-)model evolution.
- *Saheed Popoola, Xin Zhao, and Jeff Gray*, in their paper "Evolution of Bad Smells in LabVIEW Graphical Models", focus on bad smells that often indicate potential problems in software, which may lead to long-term challenges and expensive maintenance efforts. Although bad smells often occur in source code, bad smells also exist in representations of design descriptions and models. They have observed that many users of graphical modeling environments (e.g., LabVIEW) are systems engineers who may not be aware of core software engineering techniques, such as refactoring of bad smells. There exists a large body of research focused on analysing bad smells embedded in the source code of textual languages, but there has been limited research on bad smells in systems models of graphical languages. In this paper, they present a semi-automated approach for extracting design smells across versions of LabVIEW graphical models through user-defined queries.

4. About JOT

The Journal of Object Technology welcomes manuscripts describing theoretical, empirical, conceptual, and experimental results in the area of software and language engineering, including: programming paradigms, software language engineering, model-based and model-driven engineering, requirement engineering, software architecture, software validation & verification, software maintenance and evolution, software analytics, software development process and methodology. JOT Journal is currently indexed by DBLP, Google Scholar, Microsoft Academic Search, SCIRUS, Scientific Commons, DOAJ, Index of IS Journals, and SCOPUS. JOT is registered under ISSN 1660-1769.

About the Editors

Ludovico Iovino received his PHD from the University of L'Aquila, collaborating with the MDE Group. He is currently assistant professor at the GSSI – Gran Sasso Science Institute in L'Aquila in the Computer Science scientific area. His interests include Model Driven Engineering (MDE), Model Transformations, Metamodel Evolution, code generation and software quality evaluation. Currently, he's working on model-based artifacts and issues related to the metamodel evolution problem. He worked on model to code generation but also traditional code-centric development methods. He's been included in the program committees of numerous conferences and he is the co-organiser with Alfonso Pierantonio and Dalila of the Models and Evolution workshop from 3 years. He is involved in different academic projects about Model Repositories, model migration tools and Eclipse Plugins like the Edelta Project. You can contact him at ludovico.iovino@gssi.it or visit http://ludovicoiovino.com.

Dalila Tamzalit received her Ph.D. in computer science at the university of Nantes in 2000. She is an assistant professor at the University of Nantes in France since 2001. Her main research interest concerns software evolution foundations and methodologies. She published several peer-reviewed articles on this research topic in international journals and conferences. These last years, she focuses on Software Architecture Evolution. In addition to chairing the series of international workshops on Model-Driven Software Evolution (MoDSE) since 2007, Dalila Tamzalit is the head of the French CNRS research group on evolution, reuse and traceability of Information Systems (ERTSI) and co-founder and co-organiser of its dedicated workshop since 2002. She has also been co-organizer, program committee member and reviewer for French and international symposia, workshops, conferences and journals on software engineering and software evolution. Dalila Tamzalit (dalila.tamzalit@univ-nantes.fr) LS2N, University of Nantes, France. You can contact him at dalila.tamzalit@univ-nantes.fr.