SICOMORo-CM: Development of Trustworthy Systems via Models and Advanced Tools *

Elvira Albert[3], Pablo C. Cañizares[1], Esther Guerra[2], Juan de Lara[2], Esperanza Marcos[5], Manuel Núñez[1], Guillermo Román-Díez[4], Juan Manuel Vara[5], Damiano Zanardini[4]

[1] UCM-TER
Universidad Complutense de Madrid, Spain
pablocc@ucm.es, mn@sip.ucm.es
[2] UAM-miso
Universidad Autónoma de Madrid, Spain
Juan.deLara@uam.es, Esther.Guerra@uam.es
[3] UCM-COSTA
Universidad Complutense de Madrid, Spain
elvira@sip.ucm.es
[4] UPM-COSTA
Universidad Politécnica de Madrid, Spain
groman@fi.upm.es, damiano@fi.upm.es
[5] URJC-Kybele Research Group
Universidad Rey Juan Carlos
esperanza.marcos@urjc.es, juanmanuel.vara@urjc.es

Abstract. In this paper we present the SICOMORo-CM project. Its main aim is to advance the state of the art in the development of reliable and trustworthy systems by combining formal and model-based approaches. The project started on October 1st, 2014 and will last four years. The project consortium is integrated by five research groups based in Madrid (Spain) and it has been funded by the Regional Government of Madrid and the European Social Fund of the European Commission with a total of 635.088,65€.

1 Introduction

The main objective of SICOMORo-CM (Spanish acronym for Development of Trustworthy Systems via Models and Advanced Tools) consists in introducing methodologies, supported by tools, that allow the development of trustworthy and high quality software using a rigorous process that covers all its development phases. Therefore, SICOMORo-CM goes beyond partial steps that focus on isolated phases with the risk of wasting the results if there is no integrated framework for software development. SICOMORo-CM offers a joint scientific

* Research partially supported by the Comunidad de Madrid project *desarrollo de SIstemas COnfiables mediante MOdelos y herRamientas avanzadas* SICOMORo-CM (S2013/ICE-3006). The project web site is http://sicomoro-cm.es/.

program organized into 9 scientific-technological objectives. These objectives include work focused on every phase of the software development cycle (modelling, model verification, validation, and system verification); work in transversal lines that require all phases (in particular, we highlight the design of the SICOMORo-CM workflow and the implementation of a virtual colaboratory as our main expected results); and work, in cooperation with 9 partner companies, on the application of the developed methodologies and tools in industrial environments (with emphasis in transportation, automotive and cloud systems).

SICOMORo-CM is a program of high relevance since the methodology and tools developed in the project will allow software to be delivered in a more effective, efficient, and reliable manner than today, accelerating the development cycle and lowering the operational costs. This has the potential to significantly improve the competitiveness of companies using our developed technologies. In particular, it will be especially relevant for SICOMORo-CM industrial partners. It is a priority of SICOMORo-CM to show the applicability of the methodologies developed by its use, in cooperation with the industrial partners, both in the development of software systems for major industries (e.g. transportation and automotive industry) and in the definition of service and modelling operations, and of cloud systems. We also expect that SICOMORo-CM will have a relevant impact in academia since we propose an ambitious reach-out and dissemination program of the results, that comprises publications and presentations in the most relevant international events and the organization of summer schools and specialized workshops.

The SICOMORo-CM program brings together leading national research groups in the areas of formal modelling and analysis of complex software systems. The principal investigators of the five academic groups that form SICOMORo-CM consortium, in spite of their relative youth have a broad expertise in research, with very relevant publications and remarkable experience in project management, both at national and international levels. These groups work on different but complementary research areas, providing an interdisciplinary background to this challenging research agenda. Furthermore, the project's interdisciplinarity is reflected in the application fields of the partner companies, which tackle divergent areas of software development, like rail signaling systems, and infrastructure security and protection. Finally, it is worth mentioning that only a program like SICOMORo-CM gathers under the same umbrella groups that work in complementary areas but with a common objective: creating high quality software that can be more useful to society.

2 Consortium of the project

SICOMORo-CM is being implemented by five research groups located in the Madrid Region (Spain). Next, we briefly describe the main activities of each research group participating in the consortium.

The **UCM-TER** *Testing and Performance Evaluation* research group at Universidad Complutense Madrid (http://antares.sip.ucm.es/testing/) was founded by Manuel Núñez, Coordinator of the SICOMORo-CM project. The group has 17 members, with a good balance between senior researchers, fresh doctors and doctoral students. Although an important part of the work of the group concentrates on the Mathematical Foundations of Computer Science, the applicability of the results is also a priority, with an important focus on the development of tools to support the theory.

The **UAM-miso** *Modelling and Software Engineering Research Group* at the Universidad Autónoma de Madrid (http://miso.es) was founded by Juan de Lara in 2013. Its current members include 3 professors, 4 PhD students, and 3 research associates. The main focus of the group is on the development of methods and tools for Model-Driven Engineering (MDE) and Domain-Specific Languages (DSLs).

The **COSTA** research group (http://costa.ls.fi.upm.es) is split between Universidad Complutense de Madrid (UCM-COSTA) and Universidad Politécnica de Madrid (UPM-COSTA). **UCM-COSTA** has 9 members, spanning from well-known experienced researchers to students. Elvira Albert is the co-founder and coordinator of the group. Group members have their Master and PhD degrees in Mathematics or Computer Science. The main focus of the group from its beginning has been to bring to practice theoretical results in program analysis. This has been obtained by developing a number of techniques with the goal of applying them to large-scale problems, and implement tools working on state-of-the-art programming languages and systems.

The **UPM-COSTA** research group is an emerging research group coordinated by Damiano Zanardini. Currently, this group has 3 members, two of them being staff researchers with a PhD Degree in Computer Science. Apart from the research lines shared with the UCM part of the COSTA group, research interest has been devoted to analysis of heap data structures (e.g. reachability and cyclicity) in Java and termination analysis of multithreaded Java.

The **URJC-Kybele** *Service Science, Management and Engineering and Software Engineering Research Group* at Universidad Rey Juan Carlos in Madrid (http://www.kybele.es/) was founded by Esperanza Marcos in 1998. The group has now 14 researchers (11 of them doctors) who collaborate also in Kybele Consulting, the group's spin-off. Modelling has been one of the main areas of interest for Kybele from its inception, even before the advent of Model-Driven Engineering. In fact, the most relevant projects run by the group since 1999 to date have been related with the provision of methods, tools and techniques based on models for different engineering purposes, like the development of information systems or the evolution of services. Indeed, Kybele was one of the first Spanish groups working in the area of service science management and engineering, which has become later the main area of interest of the group.

3 Objectives of the project and current achievements

The project is structured around 9 objectives, which we briefly describe next, summarizing the results obtained so far.

Objective 1. Executable and trustworthy models. *UAM-miso.*
A first objective is to be able to specify DSLs (both syntax and semantics) in a cost-effective way, with the possibility of analysing the DSL semantics. In the project, we consider both denotational and operational semantics. The former are specified using (model-to-model) transformations into a semantic domain, while the latter are specified using in-place model transformations. For this purpose, we are currently investigating processes to facilitate the creation of (graphical) DSLs. These are based on deriving the DSL concrete and abstract syntax based on examples [20], and include techniques to evaluate the effectiveness of the concrete syntax [15]. We have developed reusability mechanisms for model-to-model transformations and in-place transformations so that we can map families of DSLs (e.g., workflow languages) into semantic domains (Petri nets) [29,12], or reuse in-place model transformations describing the execution semantics of the DSL [11]. This will allow the construction of libraries of reusable DSL semantics.

Objective 2. Verification of models and transformations. *UPM-COSTA.*
In this objective, we will analyse properties of both models and transformations, including model-to-model, model-to-text and in-place transformations. We will consider a case study in the verification of railway controllers provided by an industry partner.

Regarding model-to-model transformations, we are developing several techniques, for example based on static analysis of transformation definitions (using ATL) [10] and traceability analysis [18]. Regarding in-place model transformations, we are developing techniques based on backwards reasoning [8] to verify whether different model executions can violate given properties. Regarding models, we are working on analysing constraints [16], deriving techniques for slicing [22], and developing DSLs for an integral validation and verification of meta-models [21].

Objective 3. Transformations as a service. *UAM-miso.*
Based on our previous experience [9], our goal is to create a system able to optimize and execute transformations-as-a-service in the cloud, and its use in advanced scenarios (e.g., distributed and streaming transformations). We foresee integrating this system with the virtual collaborative environment of Objective 6, and the use of the modelling and verification techniques for cloud systems of Objective 8. We have currently developed a DSL to describe and generate infrastructure for MDE services [5], and we are collaborating with external groups to define transformation services for verification [26] and distributed transformations [3].

Objective 4. Verification and validation of systems. *UCM-COSTA.*
While the previous objectives dealt with models, the project also considers the verification of systems. This includes the verification of complex properties on sequential systems and concurrent programs, the development of scalable techniques for systems validation and the validation of concurrent programs. In particular, we are developing new techniques and tools to reason automatically on the behaviour of concurrent systems and understand all potential task interleavings that may arise along the execution. This is

essential to prove both liveness and safety properties of the concurrent systems, like absence of deadlocks and absence of data races, or the termination of all loops in the program. Among the contributions of the project to this objective we can mention [1,2,30].

Objective 5. Model-Based systems validation. *UCM-TER.*

To complement formal verification techniques, the project considers validation based on formal testing. This includes the definition of implementation relations, the design of algorithms for automatic generation of test cases and the proposal of passive testing techniques.

In the medium term, we will have new implementation relations and a tool to ensure its ease of use. At the end of the project, we expect to achieve all objectives. In particular, we will derive a full set of implementation relations, test case selection criteria, a formal methodology to perform passive testing for synchronous and asynchronous systems and tools that support these frameworks. We have already contributed to a state-of-the-art paper on formal testing [6] and developed new techniques for passive testing of systems with time information [23] and asynchronous communications [17].

Objective 6. Virtual Collaborative Environment. *UCM-COSTA.*

As the project aims at producing practical tools that can be used in combination along the development process, an objective is to develop supporting infrastructure for the flexible combination of tools, and their cloud-based execution. For this purpose, we have describe generic interfaces for tool integration and developed a prototype [13]. In the rest of the project, this prototype will be used to integrate the developed tools.

Objective 7. Modelling Service Operations. *URJC-Kybele.*

This objective has started in the second phase of the project. In order to support the development and analysis of service-oriented applications, we are currently developing a tool to support families of notations to model service operations, and we will support their formal analysis.

The first release of the tool has been delivered and can be downloaded at http://kybele.es/innovaserv/. It supports several notations for business modelling like Canvas [25], e3value [14], Service Blueprint [4] and Process Chain Network [28]. Since the tool has just been delivered only preliminary results are available [15] but some publications have already been submitted for consideration to high-impact conferences.

Objective 8. Cloud systems: model, verification and validation. *UCM-TER.*

We will apply the developed tools and techiques to model and analyse cloud systems. In particular, we have proposed specification techniques based on multi-level modelling [27], and we are developing an environment for the model-based analysis of cloud systems, including both expert rules and simulation for performance prediction [7]. We are also developing a methodology based on metamorphic testing for the validation of cloud systems [24].

Objective 9. Dissemination and exploitation. *UCM-TER.*

We are disseminating the project results primarily in academic conferences and journals, but we aim at disseminate and evaluate results in the software development community. For this purpose, during the second year we organized an “industry day” with the industrial partners and invited software companies in the Madrid region, showcasing the different developed tools. We have also organized seminars around the project topics.

4 SICOMORO-CM: the road ahead

While we have obtained promising scientific and technical results – which have been published in international journal and conferences – there are still different remaining challenges, which will be tackled until the end of the project.

We expect to produce a methodology for the systematic and formal development of *all phases* of the *software-development* process. In addition to the underlying theoretical framework, we will provide tools that allow a smooth transition between the different phases of the development and the technologies used in them. While several individual tools have been developed, we aim at integrating them, using the virtual collaborative environment described in Objective 6. In particular, the environment will serve to *cloudify the different tools* and deliver their functionality adopting a software-as-a-service approach. This will serve to enable the use of MDE techniques and future integration with other services and tools. Techniques for the efficient and distributed execution of model transformations, as well as the development of streaming transformation techniques will be developed as well.

Regarding Objective 7, we will work on the integration of DSLs to support the modelling of service operations, bundled into a (collaborative and virtual) modelling environment, supporting formal verification of properties, value analysis and processes, as well as import/export operations of service operations models from/to other process modelling notations.

Another project goal is to develop *verification and validation techniques* applicable to several domains – like cloud, services, concurrent applications – to ensure that the verified systems satisfy some quality guarantees (e.g., deadlock-freeness, termination of all processes, existence of upper-bounds on resource consumption, etc). We will continue working on techniques and tools in this direction. In particular, we will provide an *MDE framework* to support the verification of both models and model transformations. The framework is being developed as an open-source framework atop of Eclipse/EMF. An official Eclipse project proposal will be elaborated around the framework in order to enhance its visibility. This will contribute also to foster adoption by the industry due to the popularity of Eclipse among professional developers.

SICOMORo-CM is aimed to attract interested companies and organizations in order to enable technology transfer. The adoption of the techniques and tools delivered by the project is expected to have a verifiable impact in terms of improving the quality of the software developed and production cost-cutting. In-depth studies with program partners on the benefits provided by the methodologies and tools developed in SICOMORo-CM are also planned.

5 References to related projects

There are several projects, both at the national and international level, related to SICOMORo-CM. Next, we mention two of the FP7 European projects where members of SICOMORo-CM took part. Both projects have recently finished and some of their results have been used as inputs for SICOMORo-CM. MONDO [19] (http://www.mondo-project.org/) (*Scalable Modelling and Model Management on the Cloud*) focused on a very relevant research line of SICOMORo-CM, modelling, and on techniques to make modelling scalable. Instead, our focus in SICOMORo-CM is more on developing trustworthy systems. Envisage (http://www.envisage-project.eu/) (*Engineering Virtualized Services*) focused on applying formal approaches to services, having in mind that virtualized services can be used in the cloud. Again, SICOMORo-CM shares research interest with this project, in particular concerning services and the cloud.

References

1. E. Albert, P. Arenas, and M. Gómez-Zamalloa. Testing of concurrent and imperative software using CLP. In *18th Int. Symposium on Principles and Practice of Declarative Programming, PPDP'16*, pages 1–8. ACM Press, 2016.
2. E. Albert, A. Flores-Montoya, S. Genaim, and E. Martin-Martin. May-happen-in-parallel analysis for actor-based concurrency. *ACM Transactions on Computational Logic*, 17(2):11:1–11:39, 2016.
3. A. Benelallam, M. Tisi, J. Sánchez Cuadrado, J. de Lara, and J. Cabot. Efficient model partitioning for distributed model transformations. In *Proc SLE*, pages 226–238. ACM, 2016.
4. M. J. Bitner, A. L. Ostrom, and F. N. Morgan. Service blueprinting: a practical technique for service innovation. *California management review*, 50(3):66–94, 2008.
5. C. Carrascal, J. Sánchez, and J. de Lara. Building MDE cloud services with distil. In *CloudMDE@MODELS, CEUR Workshop Proceedings 1563*, pages 19–24, 2015.
6. A. R. Cavalli, T. Higashino, and M. Núñez. A survey on formal active and passive testing with applications to the cloud. *Annales of Telecom.*, 70(3-4):85–93, 2015.
7. P. Cerro-Cañizares, A. Nuñez, and J. de Lara. MAGICIAN: model-based design for optimizing the configuration of data-centers. In *Proc. SEKE*, pages 602–607, 2017.
8. R. Clarisó, J. Cabot, E. Guerra, and J. de Lara. Backwards reasoning for model transformations: Method and applications. *Journal of Systems and Software*, 116:113–132, 2016.
9. J. Sánchez Cuadrado and J. de Lara. Streaming model transformations: Scenarios, challenges and initial solutions. In *Proc ICMT*, volume 7909 of *Lecture Notes in Computer Science*, pages 1–16. Springer, 2013.
10. J. Sánchez Cuadrado, E. Guerra, and J. de Lara. Static analysis of model transformations. *IEEE Trans. Software Eng.*, 43(9):868–897, 2017.
11. J. de Lara and E. Guerra. *A Posteriori* typing for model-driven engineering: Concepts, analysis, and applications. *ACM Trans. Softw. Eng. Methodol.*, 25(4):31:1–31:60, 2017.
12. J. de Lara, J. Di Rocco, D. Di Ruscio, E. Guerra, L. Iovino, A. Pierantonio, and J. Sánchez Cuadrado. Reusing model transformations through typing requirements models. In *Proc. FASE*, volume 10202 of *LNCS*, pages 264–282. Springer, 2017.

13. J. Doménech, S. Genaim, E. Broch Johnsen, and R. Schlatte. Easyinterface: A toolkit for rapid development of GUIs for research prototype tools. In *Proc, FASE'17, LNCS 10202*, pages 379–383. Springer, 2017.
14. J. Gordijn, H. Akkermans, and J Van Vliet. Designing and evaluating e-business models. *IEEE Intelligent Systems*, 16(4):11–17, 2001.
15. D. Granada, J. M. Vara, M. Brambilla, V. Andrea Bollati, and E. Marcos. Analysing the cognitive effectiveness of the webml visual notation. *Software and System Modeling*, 16(1):195–227, 2017.
16. E. Guerra and J. de Lara. Automated analysis of integrity constraints in multi-level models. *Data & Knowledge Engineering*, 107:1–23, 2017.
17. R. M. Hierons, M. G. Merayo, and M. Núñez. An extended framework for passive asynchronous testing. *Journal of Logical and Algebraic Methods in Programming*, 86(1):408–424, 2017.
18. Á. Jiménez, J. M. Vara, V. A. Bollati, and E. Marcos. Metagem-trace: Improving trace generation in model transformation by leveraging the role of transformation models. *Science of Computer Programming*, 98:3–27, 2015.
19. D. S. Kolovos, L. M. Rose, R. F. Paige, E. Guerra, J. Sánchez, J. de Lara, I. Ráth, D. Varró, G. Sunyé, and M. Tisi. MONDO: Scalable modelling and model management on the cloud. In *STAF Projects Showcase, CEUR Workshop Proceedings 1400*, pages 44–53. CEUR-WS.org, 2015.
20. J. J. López-Fernández, A. Garmendia, E. Guerra, and J. de Lara. Example-based generation of graphical modelling environments. In *Proc. ECMFA*, volume 9764 of *LNCS*, pages 101–117. Springer, 2016.
21. J. J. López-Fernández, E. Guerra, and J. de Lara. Combining unit and specification-based testing for meta-model validation and verification. *Information Systems*, 62:104–135, 2016.
22. I. Mastroeni and D. Zanardini. Abstract program slicing: An abstract interpretation-based approach to program slicing. *ACM Transactions on Computational Logic*, 18(1):7:1–7:58, 2017.
23. M. G. Merayo and A. Núñez. Passive testing of communicating systems with timeouts. *Information and Software Technology*, 64:19–35, 2015.
24. A. Núñez and R. M. Hierons. A methodology for validating cloud models using metamorphic testing. *Annales of Telecommunications*, 70(3-4):127–135, 2015.
25. A. Osterwalder and Y. Pigneur. *Business model generation: a handbook for visionaries, game changers, and challengers.* John Wiley & Sons, 2010.
26. J. Di Rocco, D. Di Ruscio, A. Pierantonio, J. Sánchez, J. de Lara, and E. Guerra. Using ATL transformation services in the mdeforge collaborative modeling platform. In *Proc. ICMT'16, LNCS 9765*, pages 70–78. Springer, 2016.
27. A. Rossini, J. de Lara, E. Guerra, and N. Nikolov. A comparison of two-level and multi-level modelling for cloud-based applications. In *Proc. ECMFA*, LNCS, pages 18–32. Springer, 2015.
28. Scott E Sampson. Visualizing service operations. *J. Service Research*, 15(2):182–198, 2012.
29. J. Sánchez, E. Guerra, and J. de Lara. Reusable model transformation components with bento. In *Proc. ICMT'2015, LNCS 9152*, pages 59–65. Springer, 2015.
30. D. Zanardini, E. Albert, and K. Villela. Resource-usage-aware configuration in software product lines. *Journal of Logic and Algebraic Methods in Programming*, 85(1):173–199, 2016.