International Journal on Software Tools for Technology (2006)
DOI 10.1007/s10009-005-0220-y

SPECIAL SECTION ON SPECIFICATION AND VALIDATION OF MODELS OF REAL TIME AND EMBEDDED SYSTEMS WITH UML

Susanne Graf · Oystein Haugen · Ileana Ober · Bran Selic

Preface of "Specification and Validation of Real Time and Embedded systems in UML"

Published online: 27 January 2006

Abstract The ideas of the papers in this special section have originally been presented at the first edition of the workshop on Specification and Validation of Real Time and Embedded Systems (SVERTS) that was held as a satellite workshop of the UML 2003 Conference in San Francisco.

The motivation for initiating this workshop was the fact that UML started to be used more and more also for modelling real-time and embedded software systems, whereas it was lacking features for appropriately modelling this kind of systems. Another motivation was the fact that UML was missing an accepted dynamic semantics, or at least a framework for defining profiles not just in terms of syntax, but also their semantics.

The six papers in this section reflect all parts of this problem.

Keywords UML · Modeling · Semantics · Real time · Formal verification · Validation · Software development

1 Introduction

All the papers of this special section are motivated by the very strong needs for tools for dealing with non-functional characteristics and requirements of embedded systems. Today's applications have often strong constraints with respect to non-functional, in particular time-related aspects. Moreover, overall systems may be huge, and even if the embedded hard real-time components are relatively small, there is some global interdependence and the existence of a global model in a uniform framework is an important issue. Handling the complexity induced by functional and non-functional requirements needs new approaches. UML can play a role in such a framework due to its broad adoption.

The definition of UML has been motivated by the need for a standard notation for modelling system architectures and behaviors at functional and implementation level. UML aims at providing an integrated modelling framework encompassing architecture descriptions and behavior descriptions. A first step to the integration of extra functional characteristics into the modelling framework has been achieved by the "UML profile for Schedulability, Performance, and Time". It provides a first attempt defining some basic concepts. Nevertheless, in order to be able to exchange models and to build validation tools, a common understanding of the semantics of the given notations is also needed. Other important issues in the domain of real-time are methodology and modeling paradigms allowing to break down the complexity, and tools which are able to verify well-designed systems.

Recently, and partly influenced by some of the work presented at this workshop, a request for proposal (RFP) addressing directly embedded systems has been issued by OMG: the *Profile for Modeling and Analysis of Real-Time and Embedded systems* [25].

The aim of this workshop was to bring together researchers and practitioners to discuss different time-related issues in the context of modeling, design and validation of real-time systems, such as notations for expressing time and related requirements, semantic issues and tools and modeling paradigms for real-time systems. At the time, when the workshop was held, the use of UML in the domain of real-time and embedded systems was still in its infancy. Whereas today its role is much more established partially based on ideas presented in the papers of this section.

S. Graf (✉)
Verimag, France
E-mail: susanne.graf@imag.fr, http://www-verimag.imag.fr/graf

O. Haugen
University of Oslo, Norway
E-mail: oysteinh@ifi.uio.no

I. Ober
IRIT, Toulouse, France
E-mail: ileana.ober@irit.fr

B. Selic
IBM Rational Software, Canada
E-mail: bselic@ca.ibm.com

2 The Contributions

2.1 Contribution of the IST OMEGA project

The first three papers present work done in the IST OMEGA project that completed in early 2005 and the results of this project were one of the motivations of this workshop (for an overview see [13, 14]). The aim of this project was

- to adapt the existing UML profile in order make it appropriate for modeling real-time embedded systems as well as the expression of properties to be validated;
- and to provide validation tools and methodology for allowing a smooth integration of validation and analysis in the development process and to validate them on case studies.

The profile that has been chosen is based on a slight generalization of the profile found in a number of UML CASE tools used in this application area for the operational part. It introduces a notion of *activity group*, a set of objects related by containment relationships defining a *thread* – which is similar to SDL *processes* or ROOM *capsules* – and is based on communication through either synchronous operation calls or asynchronous signal exchanges. For the expression of requirements, an extension of OCL 2.0 [23, 26] with event histories were introduced, as well as live sequence charts (LSC) [9, 11, 21] as a means to replace sequence diagrams and, finally, in [22], *observers* are introduced, – an extension of state machines for the definition of language acceptors. Timing extensions are defined in [15]. Paper [28] provides a formalization of the semantics of the main concepts in PVS.

The verification tools developed for this profile, which were applied to several case studies[1] are based on the above mentioned PVS formalization [20] or on model checking techniques, where BDD based techniques [27] and enumerative techniques have been considered [22].

2.2 The individual contributions

The papers in this section all relate to some UML profile concerned with usages of UML for distributed and/or embedded systems, where time-related and other non-functional properties are of primary importance, and appropriate profiles in this domain.

The paper on "*A Semantics of Communicating Reactive Objects with Timing*" [28] provides an overview on the operational kernel model of the OMEGA UML profile in a relatively abstract setting. The paper considers the abstract concepts of inheritance, communication and the run-to-completion execution strategy defined by the notion of *activity group* of the OMEGA profile. It considers time at an abstract level by an extension of state machines to timed automata [2].

The main aim of this paper is discussing some of the semantic decisions of the profile presented in the form of a global operational semantics in [10] and to provide a formal underpinning by means of a formalization in PVS. This formalization is also available in the form of a translator taking a UML specification specified as an XMI file. Moreover, a package of PVS theories which can be used for proving properties is available [3]. This package has been used in some small case studies [19].

The paper on "*Timed annotations in UML*" [15] presents the OMEGA timing extension in more detail. The basic idea is to lift the expressive concepts for handling time in the context of timed automata and from scheduling analysis to the UML level by taking into account the specifics of UML and usability requirements from users.

At UML level, such timed automata are represented by so called ≪observer≫ classes, defined by state machines. They use an ≪event≫ allowing to syntactically characterize any state change and associated time point in the semantic models together with a means for selecting particular sets of occurrences into a user defined "event".

A notion of resource is introduced for the definition of mutual exclusion constraints and powerful dynamic priority rules for the definition of execution and scheduling policies. Patterns for timing constraints, scheduling policies, as they are defined in the UML profile for Scheduling Performance and Time [24] are derived concepts in this profile. Some concepts from this profile will be proposed as response to the call for a UML Profile for Modeling and Analysis of Real-Time and Embedded systems.

The paper on "*Validating timed UML models by simulation and verification*" [22] presents a tool for abstractly executing and verifying models using the OMEGA profile with the above mentioned time extensions. It provides an alternative semantics for the profile by means of a mapping into the IF formalism [6] which is much closer to UML than PVS, but has a well-defined semantics. IF is also connected to a number of state-of-the-art validation tools [7].

The semantic mapping is defined in such a way to make adaptation to variants of the considered UML profile as easy as possible. In particular, the definition of the run-to-completion semantics of *activity groups* by means of two dynamic priority rules shows the expressiveness of the proposed priority concept.

The profile and the use of the tools is demonstrated on a case study in which coordination and scheduling properties play a crucial rule. Also, the case study uses priorities, but in a much more classical setting.

The paper "*Towards a "Synchronous Reactive" UML profile*" [12] uses some interesting examples to identify the need for some semantic changes to UML to allow a synchronous interpretation of UML, where "synchronous" is to be understood in the sense of synchronous languages, such as Esterel, Lustre and Signal [4, 8].

In particular, for this purpose it is important to allow the simultaneous consumption of all signals present in the signal

[1] An overview on the case studies can be found at http//www-omega.imag.fr. Some results are presented in [22] but more complete publications are forthcoming

queue. Obviously, it is possible to provide a workaround for obtaining the same effect also in presence of the run-to-completion rule, but it can not be done in way acceptable to the users. Similar usability problems have also been reported by some users in the OMEGA project. An interesting point is that the time-independent point of view proposed in this paper is compatible with the time extensions proposed in [15].

Note also that the ideas presented in this paper have strongly influenced the call for a UML Profile for Modeling and Analysis of Real-Time and Embedded systems.

The paper "*On a time enriched OCL liveness template*" [18] is concerned with the description of global timing constraints in real-time distributed systems which should be analyzable independently of the existence of a detailed specification. For this purpose, the paper proposes an extension of OCL with temporal patterns. From a syntactic point of view, the proposal consists in replacing or enriching post conditions to express ≪eventuality≫ constraints and ≪time-frames≫.

A semantic underpinning for these OCL notation is provided in terms of logic of knowledge [16]. The knowledge related modalities are used here to distinguish between local and global constraints. This is an interesting usage of logic of knowledge which might also be profitable for the timing constraints defined in [15], which are more general, but more closely related to a given detailed specification.

The paper "*Hybrid profile for UML 2.0*" [5] proposes extending the use of UML for modelling hybrid systems. In this context, hybrid systems combine the use of continous and discrete variables, where the value of *continuous variables* changes with time according to some law (e.g., a differential equation).

The concrete proposal made in this contribution is based on use of the CHARON language [1] and thus directly provides a formal semantics in terms of hybrid automata [17] and tool support by the CHARON tool.

Relatively little work exists on extending UML to hybrid systems, so that this is the main contribution of the paper. In this context, the choice of CHARON is an interesting setting, defining a number of concepts essential for hybrid systems. The concrete approach taken here is extending UML by using stereotypes grouped in a UML profile, to make possible the use of an UML editor as a convenient graphical editor for Charon. The paper ends with an interesting discussion of the UML profile as well as lessons learned that can be useful to both UML and CHARON.

References

1. Alur, R., Dang, T., Esposito, J.M., Fierro, R.B., Hur, Y., Ivancic, F., Kumar, V., Lee, I., Mishra, P., Pappas, G. J., Sokolsky, O.: Hierarchical hybrid modeling of embedded systems. In: Embedded Software, First International Workshop, EMSOFT 2001, Tahoe City, CA, USA, October, 8–10, 2001. LNCS 2211 pp. 14–31 (2001)
2. Alur, R., Dill, D.: A theory of timed automata. Theor. Comp. Sci. **126**, 183–235 (1994)
3. Arons, T., Hooman, J., Kugler, H., Pnueli, A., van der Zwaag, M.: Deductive verification of UML models in TLPVS. In: Proceedings UML 2004, pp. 335–349. LNCS 3273, Springer-Verlag (2004)
4. Benveniste, A., Caspi, P., Edwards, S.A., Halbwachs, N., Le Guernic, P., de Simone, R.: The synchronous languages 12 years later. In: Proceedings of the IEEE, **91**(1) (2003)
5. Berkenkötter, K., Bisanz, S., Hennemann, U., Peleska, J.: Hybrid profile for UML 2.0. STTT, Int. J. Softw. Tools Technol. Transf. (this volume) (2005)
6. Bozga, M., Graf, S., Mounier, L.: IF-2.0: A validation environment for component-based real-time systems. In: Proceedings of Conference on Computer Aided Verification, CAV'02, Copenhagen. LNCS 2404, Springer Verlag (2002)
7. Bozga, M., Graf, S., Ober, I., Ober, I., Sifakis, J.: The IF toolset. In: SFM-04:RT 4th Int. School on Formal Methods for the Design of Computer, Communication and Software Systems: Real Time. LNCS 3185 (June 2004)
8. Caspi, P., Benveniste, A., Le Guernic, P., Halbwachs, N.: A decade of concurrency, reflexions and perspeptives. In: REX Symposium, LNCS 803 (1993)
9. Damm, W., Harel, D.: LSCs: Breathing life into Message Sequence Charts. In: Ciancarini, P., Fantechi, A., Gorrieri, R. (eds.), FMOODS'99 IFIP TC6/WG6.1 Third International Conference on Formal Methods for Open Object-Based Distributed Systems. Kluwer Academic Publishers, 1999. J. Formal Methods Syst. Des. (July 2001)
10. Damm, W., Josko, B., Pnueli, A., Votintseva, A.: Understanding UML: A formal semantics of concurrency and communication in real-time UML. In: de Boer, F., Bonsangue, M., Graf, S., de Roever, W.-P. (eds.), Proceedings of the 1st Symposium on Formal Methods for Components and Objects (FMCO 2002), pp. 70–98. LNCS Tutorials 2852 (2003)
11. Damm W., Westphal, B.: Live and let die: LSC based verification of UML models. Science of Computer Programming (2004) (to appear) available at http//www.sciencedirect.com/
12. de Simone, R., André, Ch.: Towards a "Synchronous Reactive" UML profile. STTT, Int. J. Softw. Tools Technol. Transf. (this volume) (2005)
13. Graf, S., de Boer, F.S., Combes, P., Hooman, J., Kugler, H., Kyas, M., Lesens, D., Ober, I., Votintseva, A., Yushtein, Y., Zenou, M.: Omega final project report. Deliverable of the Omega IST project (2005)
14. Graf, S., Hooman, J.: Correct development of embedded systems. In: European Workshop on Software Architecture: Languages, Styles, Models, Tools, and Applications (EWSA 2004), co-located with ICSE 2004, St. Andrews, Scotland, LNCS 3047, pp. 241–249. Springer-Verlag (2004)
15. Graf, S., Ober, I., Ober, I.: Timed annotations in UML. STTT, Int. J. Softw. Tools Technol. Transf. (this volume) (2005)
16. Halpern, J.Y.,Vardi, M.Y.: The complexity of reasoning about knowledge and time. i. lower bounds. J. Comp. Syst. Sci. **38**(1), 195–237 (1989)
17. Henzinger, T.A.: The theory of hybrid automata. In: Conference on Logic in Computer Science, LICS, pp. 278–292 (1996)
18. Küster-Filipe, J., Anderson, S.: On a time enriched OCL liveness template. STTT, Int. J. Softw. Tools Technol. Transf. (this volume) (2005)
19. Kyas, M.: A compositional proof of the sieve of Eratosthenes in PVS. Technical report, University of Kiel (2004)
20. Kyas, M., Fecher, H., de Boer, F.S., van der Zwaag, M., Hooman, J., Arons, T., Kugler, H.: Formalizing UML models and OCL constraints in PVS. In: Workshop on Semantic Foundations of Engineering Design Languages, SFEDL 2004, ENTCS 115, pp. 39–47. Elsevier (2004)
21. Marelly, R., Harel, D., Kugler, H.: Multiple instances and symbolic variables in executable sequence charts. In: Proceedings of 17th Annual ACM Conference on Object-Oriented Programming, Systems, Languages and Applications (OOPSLA'02), pp. 83–100. Seattle, WA (2002)

22. Ober, I., Graf, S., Ober, I.: Validating timed UML models by simulation and verification. STTT, Int. J. Softw. Tools Technol. Transf. (this volume) (2005)
23. OMG Unified Modeling Language Specification—Object Constraint Language Version 2.0 (2003)
24. OMG.: Response to the OMG RFP for Schedulability, Performance and Time, v. 2.0. OMG document ad/2002-03-04 (March 2002)
25. OMG.: UML profile for modeling and analysis of real-time and embedded systems (February 2005)
26. Richters, M., Gogolla, M.: A metamodel for OCL. In: France, R., Rumpe, B. (eds.), UML'99—The Unified Modeling Language. Beyond the Standard. Second International Conference, Fort Collins, CO, USA, October 28–30. 1999, Proceedings, LNCS 1723. Springer (1999)
27. Schinz, I., Toben, T., Mrugalla, C., Westphal, B.: The Rhapsody UML verification environment. In: 2nd International Conference on Software Engineering and Formal Methods (SEFM 2004), pp. 174–183. IEEE Computer Society (2004)
28. van der Zwaag, M., Hooman, J.: A semantics of communicating reactive objects with timing. STTT, Int. J. Softw. Tools Technol. Transf. (this volume) (2005)