zbMATH Open
THE FIRST RESOURCE FOR MATHEMATICS

Chen, Zhenbang; **Liu, Zhiming**; **Ravn, Anders P.**; **Stolz, Volker**; **Zhan, Naijun**
Refinement and verification in component-based model-driven design. (English)
Zbl 1178.68158
Sci. Comput. Program. 74, No. 4, 168-196 (2009).

Summary: Modern software development is complex as it has to deal with many different and yet related aspects of applications. In practical software engineering this is now handled by a UML-like modelling approach in which different aspects are modelled by different notations. Component-based and object-oriented design techniques are found effective in the support of separation of correctness concerns of different aspects. These techniques are practised in a model-driven development process in which models are constructed in each phase of the development. To ensure the correctness of the software system developed, all models constructed in each phase are verifiable. This requires that the modelling notations are formally defined and related in order to have tool support developed for the integration of sophisticated checkers, generators and transformations.

This paper summarises our research on the method of Refinement of Component and Object Systems (rCOS) and illustrates it with experiences from the work on the Common Component Modelling Example (CoCoME). This gives evidence that the formal techniques developed in rCOS can be integrated into a model-driven development process and shows where it may be integrated in computer-aided software engineering (CASE) tools for adding formally supported checking, transformation and generation facilities.

MSC:

68N99 Theory of software
68Q60 Specification and verification (program logics, model checking, etc.)

Cited in **5** Documents

Keywords:

formal methods; multi-view modelling; rcos; software design process; tool design; UML

Software:

Circus; Eiffel; ESC/Java; rCOS; Spec#; SPIN; TCOZ; UNITY; Uppaal; Z

Full Text: DOI

References:

[1] R.-J. Back, L. Petre, I.P. Paltor, Formalising UML use cases in the refinement calculus, Tech. Rep. TUCS-TR-279, Turku Centre for Computer Science and åbo Akademi University, Finland, May 1999

[2] Back, R. -J.; Von Wright, J.: Trace refinement of action systems, Lecture notes in computer science 836 (1994)

[3] Back, R. -J.; Von Wright, J.: Refinement calculus: A systematic introduction, Graduate texts in computer science (1998) · Zbl 0949.68094

[4] Barnett, M.; Leino, K. M.; Schulte, W.: The spec# programming system: an overview, Lecture notes in computer science 3362 (2005)

[5] Borba, P.; Sampaio, A.; Cornélio, M.: A refinement algebra for object-oriented programming, Lecture notes in computer science 2743, 457-482 (2003)

[6] Cavalcanti, A.; Naumann, D.: A weakest precondition semantics for an object-oriented language of refinement, Lecture notes in computer science 1709, 1439-1460 (1999) · Zbl 0953.68081

[7] Cavalcanti, A.; Sampaio, A.; Woodcock, J.: A refinement strategy for circus, Formal aspects of computing 15, No. 2–3, 146-181 (2003) · Zbl 1093.68555 · doi:10.1007/s00165-003-0006-5

[8] Chalin, P.; Kiniry, J. R.; Leavens, G. T.; Poll, E.: Beyond assertions: advanced specification and verification with JML and ESC/java2, Lecture notes in computer science 4111 (2006)

[9] Chandy, K.; Misra, J.: Parallel program design: A foundation, (1988) · Zbl 0717.68034

[10] Chen, X.; He, J.; Liu, Z.; Zhan, N.: A model of component-based programing, Lecture notes in computer science 4767 (2007)

© 2021 FIZ Karlsruhe GmbH

[11] Chen, X.; Liu, Z.; Mencl, V.: Separation of concerns and consistent integration in requirements modelling, Lecture notes in computer science 4362 (2007) · Zbl 1131.68414

[12] Z. Chen, A.H. Hannousse, D. Van Hung, I. Knoll, X. Li, Z. Liu, Y. Liu, Q. Nan, J.C. Okika, A.P. Ravn, V. Stolz, L. Yang, N. Zhan, Modelling with relational calculus of object and component systems - rCOS, in: CoCoME, 2007, pp. 116–145. doi:10.1007/978-3-540-85289-6_6

[13] Chen, Z.; Li, X.; Liu, Z.; Stolz, V.; Yang, L.: Harnessing rcos for tool support – the cocome experience, Lecture notes in computer science 4700, 83-114 (2007) · Zbl 1151.68380 · doi:10.1007/978-3-540-75221-9_5

[14] Z. Chen, Z. Liu, A. Ravn, V. Stolz, N. Zhan, Refinement and verification in component-based model driven design, Tech. Rep. 388, UNU/IIST, P.O. Box 3058, Macao, 2007 · Zbl 1178.68158

[15] Z. Chen, Z. Liu, V. Stolz, The rCOS tool, in: Modelling and Analysis in VDM: Proceedings of the Fourth VDM/Overture Workshop, Technical Report, No. CS-TR-1099, Newcastle University, 2008

[16] The Concurrency Workbench. URL: http://homepages.inf.ed.ac.uk/perdita/cwb/

[17] Dürr, E.; Dusink, E.: The role of VDM++ in the development of a real-time tracking and tracing system, Lecture notes in computer science 670 (1993)

[18] Flanagan, C.: Extended static checking for Java, (2002)

[19] Fowler, M.; Beck, K.; Brant, J.; Opdyke, W.; Roberts, D.: Refactoring: improving the design of existing code, (1999)

[20] Gamma, E.: Design patterns, (1995)

[21] Gosling, J.; Joy, B.; Steele, G.: The Java language specification, (1996) · Zbl 0865.68001

[22] He, J.; Li, X.; Liu, Z.: Component-based software engineering, Lecture notes in computer science 3722 (2005) · Zbl 1169.68366

[23] He, J.; Li, X.; Liu, Z.: Rcos: A refinement calculus for object systems, Theoretical computer science 365, No. 1–2, 109-142 (2006) · Zbl 1118.68049

[24] He, J.; Li, X.; Liu, Z.: A theory of reactive components, Entcs 160 (2006)

[25] Hoare, C.: Communicating sequential processes, (1985) · Zbl 0637.68007

[26] Hoare, C.: Verified software: theories, tools, experiments, Lecture notes in computer science 4171, 21-29 (2007)

[27] Hoare, C.; He, J.: Unifying theories of programming, (1998) · Zbl 1005.68036

[28] Holzmann, G.: The SPIN model checker: primer and reference manual, (2003)

[29] Jones, C.: Systematic software development using VDM, (1990) · Zbl 0743.68048

[30] Kruchten, P.: The rational unified process–an introduction, (2000)

[31] Lamport, L.: Specifying systems: the TLA+ language and tools for hardware and software engineers, (2002)

[32] Larman, C.: Applying UML and patterns: an introduction to object-oriented analysis and design and the unified process, (2001)

[33] Larsen, K.; Pettersson, P.; Yi, W.: UPPAAL in a nutshell, Sttt 1, No. 1–2, 134-152 (1997) · Zbl 1060.68577 · doi:10.1007/s100090050010

[34] Leavens, G.: Jml's rich, inherited specification for behavioural subtypes, Lecture notes in computer science 4260 (2006)

[35] Li, X.; Liu, Z.: Prototyping system requirements model, Entcs 207, 17-32 (2008)

[36] Z. Liu, V. Mencl, A.P. Ravn, L. Yang, Harnessing theories for tool support, in: Proc. International Symposium on Leveraging Applications of Formal Methods, Verification and Validation, ISoLA06, IEEE Computer Society, 2006, pp. 371–382. Full version as UNU-IIST Technical Report 343, http://www.iist.unu.edu

[37] Mahony, B.; Dong, J.: Deep semantic links of TCSP and object-Z: TCOZ approach, Formal aspects of computing 3, No. 2, 146-160 (2002) · Zbl 1063.68603 · doi:10.1007/s001650200004

[38] Meyer, B.: Eiffel: the language, (1992) · Zbl 0779.68013

[39] Milner, R.: A calculus of communicating systems, (1980) · Zbl 0452.68027

[40] Morgan, C.: Programming from specifications, (1994) · Zbl 0829.68083

[41] NoMagic, Inc., MagicDraw. URL: http://www.magicdraw.com/

[42] Object Management Group, MOF QVT final adopted specification, ptc/05-11-01. http://www.omg.org/docs/ptc/05-11-01.pdf, 2005

[43] Object Management Group, Unified Modeling Language: Superstructure, version 2.0, final adopted specification, 2005. URL: http://www.omg.org/cgi-bin/doc?formal/05-07-04

[44] Object Management Group, XML Metadata Interchange, 2005. URL: http://www.omg.org/cgi-bin/doc?formal/2005-09-01

[45] Olderog, E. -R.; Wehrheim, H.: Specification and (property) inheritance in CSP-OZ, Science of computer programming 55, 227-257 (2005) · Zbl 1075.68051 · doi:10.1016/j.scico.2004.05.017

[46] Plasil, F.; Visnosky, S.: Behavior protocols for software components, IEEE transactions on software engineering 28, No. 11, 1056-1070 (2002)

[47] , Lecture notes in computer science 5153 (2008)

[48] Roscoe, A.: The theory and practice of concurrency, (1997)

[49] Rumbaugh, J.; Jacobson, I.; Booch, G.: The unified modelling language reference manual, (1999)

[50] Schneider, A.: The B-method, (2001)

© 2021 FIZ Karlsruhe GmbH

[51] Smith, G.: The object-Z specification language, (2000) · Zbl 0944.68124

[52] Tata Consultancy Services, Mastercraft, http://www.tata-mastercraft.com

[53] Topcased–Open Source Engineering Workshop, http://topcased.org

[54] Woodcock, J.; Davies, J.: Using Z: specification, refinement, and proof, (1996) · Zbl 0855.68060

[55] Woodcock, J.; Morgan, C.: Refinement of state-based concurrent systems, Lecture notes in computer science 428 (1990)

[56] L. Yang, V. Mencl, V. Stolz, Z. Liu, Automating correctness preserving model-to-model transformation in MDA, in: Proc. of Asian Working Conference on Verified Software, UNU-IIST Technical Report 348, 2006

[57] Yang, L.; Stolz, V.: Integrating refinement into software development tools, Entcs 207, 69-88 (2008)

[58] L. Zhao, X. Liu, Z. Liu, Z. Qiu, Graph transformations for object-oriented refinement, Formal Aspects of Computing, Springer, Published online: 8 January 2008

This reference list is based on information provided by the publisher or from digital mathematics libraries. Its items are heuristically matched to zbMATH identifiers and may contain data conversion errors. It attempts to reflect the references listed in the original paper as accurately as possible without claiming the completeness or perfect precision of the matching.

© 2021 FIZ Karlsruhe GmbH