Formal Foundations for the Unified Modeling Language

A. Funes, A. Dasso, D. Riesco, G. Montejano, R. Uzal

SEG

http://sel.unsl.edu.ar/

Universidad Nacional de San Luis

Ejército de los Andes 950

D5700HHW San Luis

Argentina

{afunes, arisdas, driesco, gmonte, ruzal}@unsl.edu.ar

Tel.: +54 (0) 2652 42 4027 ext. 251

Fax: +54 (0) 2652 43 0224

Abstract

We present in this work an outline of an ongoing research line in the framework of the Software Engineering Group (SEG) at the National University of San Luis. We describe here the previous work carried out by the group in formalizing UML using RSL, as well as the current and future work in the matter.

Introduction

The Unified Modeling Language (UML) [1][21][25] is a graphical language for modeling and specifying software systems and it consists of a set of constructs common to most object-oriented languages. However, although UML notations are easily communicated, their semantics are informal and –consequently– they can be ambiguous. There are an important number of theoretical works that deal with the integration of graphical notations and mathematically precise formalisms (see [3][13][17][26][23][15][5]). A good classification of the different proposals to carry out this integration can be found in [22]. Several efforts have been conducted to formalize the semantics and the syntax of different parts of UML (see [19][16][14][8][9][6][2][7]). Only a few use the RAISE Specification Language (RSL) as formal basis (see [24][18][4][10]).

RSL is a formal specification language [11], which receives its name from the RAISE method. The RAISE (Rigorous Approach to Industrial Software Engineering) [12] consists of a number of techniques and strategies for doing formal development and proofs. Its language, RSL, is a wide spectrum specification language. It allows the use of different styles of specifications: applicative or imperative; sequential or concurrent; direct (explicit) or axiomatic (implicit); algebraic (with abstract data types) or model-oriented (with concrete data types).

Past, Current and Future Work

In [24] we presented a first proposal for the semantics of a class using RSL. In this work only the basic syntactic elements were considered. Also binary associations were treated. All of them given through particular examples.

In [18] we gave a semantics in RSL for association class, aggregation and generalization through concrete examples.

In [4] we presented a proposal for using the translation of a class diagram applying the semantics given in the two previous works ([18] and [24]) as an initial applicative specification for the RAISE method.

In [10] an exhaustive and generic treatment of classes, n-ary associations, dependency, aggregation and composition, generalization, template classes, abstract classes and other syntactic elements were considered. Their semantics were given taking into account the integration with the other syntactic elements in the context of an integrated frame. The integration of the specifications was given by the semantics of the class diagram. Note that the semantics for associations, generalizations, association classes and aggregation were not given using the same RSL constructions than in [18] and [24]. Furthermore in [10], the semantics were given for the general cases. From this analysis a set of templates could be abstracted. These templates were used as the basis for the implementation of an automatic tool, which translate a class diagram into a RSL specification. The templates as well as the tool are also reported in [10]. Another important point in this work is that the syntax for the RSL-translatable class diagrams was formally specified using also RSL.

Current and future work includes the study of the integration between an specification obtained by following [10] and dynamics diagrams in UML. Currently, we are working specifically in an extension using state machines in UML to model the dynamic aspects of a class. This allows to model the life of an object, that is, the different states in which an object of the class can be and the transitions among these states. An interpretation in RSL for these states machines given with the class diagram will be embedded in the entire specification obtained for the class diagram.

References

[1] G. Booch, J. Rumbaugh, and I. Jacobson. *The Unified Modeling Language User Guide*. Addison-Wesley, 1999.

[2] R. Breu et al., Towards a formalization of the Unified Modeling Language. In *Proceedings of ECOOP'97*, number 1241 in *LNCS*, Springer-Verlag, 1997.

[3] S. Cook and J. Daniels, Let's get formal. Journal of Object-Oriented Programming (JOOP), July-August 1994.

[4] N. Debnath, D. Riesco et al., "Using UML class diagram for RAISE Applicative Specification", ACIS International Journal of "Computer & Information Science", Vol. 3, Pages 84-93, March 2002. ISBN/ISSN 1525-9293.

[5] R. Duke et al., The Object-Z Specification Language, Technology of Object-Oriented Languages and Systems: TOOLS 5. Prentice-Hall, 1991.

[6] A. Evans et al., Developing the UML as a formal modeling notation, UML'98 Beyond the notation, Muller and Bezivin editors, number 1618 in *LNCS*, Springer-Verlag, 1998.

[7] A. Evans et al., Towards a core metamodelling semantics of UML, Behavioral Specifications of Businesses and Systems, H. Kilov editro, Kluwer Academic Publishers, 1999.

[8] R. France. A Problem-Oriented Analysis of Basic UML Static Requirements Modeling Concepts. In Proceedings of OOPSLA ' 99, Denver, CO, USA, Nov. 1999.

[9] R. France and B. Rumpe, editors. Proceedings of the UML´99 conference, Beyond the Standard, Colorado, USA, number 1723 in LNCS, Springer-Verlag, 1999.

[10] A. Funes and C. George. , Chapter 8: "Formalizing UML class diagrams" in "UML and the Unified Process", ISBN 1931777446, Idea Group Publishing; April 1, 2003.

[11] C. George et al. The RAISE Specification Language. Prentice-Hall International (UK) Limited, 1992.

[12] C. George et al. The RAISE Development Method. Prentice-Hall International (UK) Limited, 1995.

[13] S. Goldsack and S. Kent, Chapter 3: LOTOS in the Object-oriented analysis process, Formal Methods and Object Technology, Springer-Verlag, 1996.

[14] S. Kim and D. Carrington, A Formal Specification Mapping between UML Models and Object-Z Specifications. In LNCS, number 1878, pages 2–21. Springer-Verlag, 2000.

[15] K. Lano, Z++, An object-oriented extension to Z. In John Nicholls editor, Z user workshop, Oxford, 1990, Workshops in Computing, Springer-Verlag, 1991.

[16] K. Lano and J. Biccaregui, Formalizing the UML in Structured Temporal Theories, 2nd ECOOP Workshop on Precise Behavioral Semantics, TUM-I9813, Technische Universitat Munchen, 1998.

[17] Meyer, E. and Souquieres, J. (1999). A Systematic Approach to Transform OMT Diagrams to a B Specification. In *Proceedings of FM '99*, volume I of *LNCS*, pages 875–895.

[18] G. Montejano et al., "RAISE Formalization of UML Class Associations", In Proceedings of CAINE 2000, Honolulu, Hawai, USA, Nov 1-3, 2000.

[19] P. Muller and J. Bezivin, editors. Proceedings of the UML´98 conference, Beyond the notation, Mulhouse, France, number 1618 in LNCS , Springer-Verlag, 1998.

[21] OMG. OMG-Unified Modeling Language v1.4, chapter UML Notation Guide. http://www.omg.org/technology/documents/formal/uml.htm, September 2001.

[22] C. Pons and G. Baum. Formal Foundations of Object-Oriented Modeling Notations. In Proceedings of 3rd IEEE International Conference on Formal Engineering Methods (ICFEM'00), September 04 - 07, 2000, York, England.

[23] G. Reggio and M. Larosa, A graphic notation for formal specification of dynamic systems. In Proceedings of FME'97, number 1313 in LNCS, Springer-Verlag, 1997.

[24] D. Riesco et al., "UML Class Structure Interpretation using RAISE Abstract Applicative Specification", In Proceedings of CAINE 2000, Honolulu, Hawai, USA, Nov 1-3, 2000.

[25] J. Rumbaugh, I. Jacobson, and G. Booch. The Unified Modeling Languge Reference Manual. Object Technology Series. Addison Wesley, 1999.

[26] M. Weber, Combining Statecharts and Z for the Design of Safety-Critical Control Systems. In *Proceedings of the 3rd International Symposium of FME'96.* Oxford, 1996.