Improving the Syntax and Semantics of Goal Modelling Languages

Raimundas Matulevičius

PReCISE, Computer Science Faculty, University of Namur, Belgium
rma@info.fundp.ac.be

Abstract. One major obstacle to requirements engineering (RE) is the growing complexity of today's systems. Such a complexity can only be fought efficiently by powerful abstraction mechanisms as incarnated by goal-modelling techniques. Unfortunately, the research efforts in this area are fragmented, which is a major impediment to a wide adoption by practitioners. In this work, we describe our approach how to aim the fragmentation by adopting a rigorous and novel approach for comparing and integrating goal modelling languages (GMLs). We investigate both syntax (using the principles for the effective communication) and semantics (using the UEML approach and the ISSRM reference model) and apply GMLs to solve domain specific problems (e.g. for security risk management). We hope to improve the coordination of research in this field, so that a comprehensive, sound, efficient, standard and tool-supported goal modelling language can emerge, be put into the hands of IS developers, and that the overall quality of IS developments can be improved.

1 Introduction

Goal-modelling languages (GMLs) have been a subject of research and experimentation for more than 15 years and have proved extremely valuable tools in a great number of situations. We can observe a host of GMLs and their variants– *i**, GRL, Tropos, NFR and KAOS. However, each GML comes with its own terminology, syntax, semantics, and process. In [1] Kavakli and Loucopoulos examined 15 GMLs and classified them along four dimensions: "usage", "subject", "representation" and "development". The authors identified that fragmentation appears at all levels. The languages have constructs that force developers to emphasise some aspects of the problem and neglect others. The more people work with one particular language, the more their thinking is influenced by this language, and their awareness of those aspects of the world that do not fit in, may consequently be diminished thus resulting in incomplete specification of the problem. Also, different issues within a problem situation may be relevant for different people at the same time, however not supported by the same GML.

Due to this fragmentation, we have not yet observed a widespread adoption of GMLs by practitioners. This is regrettable since RE is where GMLs are expected to have the highest payoff. In [1] authors have stressed the importance of more integration efforts to obtain a stronger GML that takes advantage of the many streams of goal-oriented research. In the literature we can find a number of attempts to unify GMLs at different levels, as well as to compare the meaning of their concepts following various approaches. However, none of them results in the systematic approach relating different GML aspects into the unified view.

In this paper we propose to yield a comparison and integration of GMLs. We present an on-going research, which analyses different GML quality aspects. The purpose is to develop an integrated and tool-supported GML, which would help improving the RE process. The structure of the paper is as follows: in Section 2 we introduce a research objective. Section 3 presents the recent contributions. Section 4 summarises our work and points out some future work.

2 Research Objective

The overall objective is the *comparison and integration of GMLs*. The objective is divided into four subgoals shown in Fig. 1. Firstly (*i*) we intend to assess the GMLs quality at the coarse-grained level using systematic evaluation frameworks. Secondly (*ii*) we evaluate the GML quality at the fine-grained level and define a precise syntax and semantics of GML constructs. Thirdly (*iii*) we compare tools that support modelling with GMLs. Finally (*iv*) we use the results of our comparisons to determine rules for language integration and model translation at both syntactic and semantic levels. The latter subgoal includes development of the integrated GML supported by a (prototype) tool.

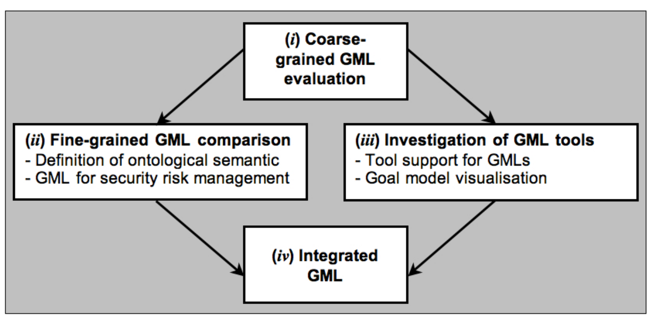

Fig. 1. Research method

3 Contribution

In this section we briefly present the contributions we achieved so far.

3.1 Coarse-grained GML Evaluation

In [2] we report on the experiment where two GMLs, namely *i** and KAOS, and models created using them are evaluated following the semiotic quality framework. This framework separates between different quality types (like semantic, syntactic, pragmatic and others). The experiment showed that the quality of individual goal models depends on particular language characteristics with respect to a given context. Even if one language is evaluated better than the other, this does not guarantee that the quality of the goal model would be better. Model quality much depends on the user's experience, the effort spent for model creation and the evaluator's subjective judgment.

3.2 Fine-grained GML Comparison

Definition of Ontological Semantics. In [3] we have applied the UEML approach [4] to investigate meaning of the GRL and KAOS constructs. Here a language construct can be described by (*i*) decomposing it to the represented classes, properties, states and transformations and by (*ii*) mapping them to the common UEML ontology. The study introduces a set of correspondences between two analysed languages. These correspondences can be used to translate GRL and KAOS models to each other based on their explicit semantics. This can help improve the traceability between models and tools used at different development stages. For instance, between early requirements elicitation using GRL, and late requirements specification using KAOS (or Tropos).

GML for Security Risk Management. We also investigated how GMLs are applicable for specific domains. We applied the reference model [5] of the information system security risk management (ISSRM) in order to check if concepts of Secure Tropos (which utilises *security constraints*) are adequate and sufficient for security risk management. The results indicate that Secure Tropos, firstly, has to be provided with guidelines as to when and how to use the constructs to avoid misinterpretations of ISSRM. Secondly, it should be improved with additional constructs to cover ISSRM better. In addition to Secure Tropos we have analysed Misuse cases [6]. We envision that after analysing a number of security languages it will be possible to facilitate model transformation to represent system security using different perspectives.

3.3 Investigation of GML supporting Tools

Tool support for GMLs. In [7] we have investigated goal modelling tools (e.g. OME, TAOM4E). We have observed that most of them are prototypes, thus requiring serious improvements before acquiring them to practice. To become more mature tools should be able to prepare and maintain not only the goal models, but the requirements specifications, too.

Model Visualisation. The problem with the goal models is that for the humans they quickly become difficult to comprehend the displayed information. In [8] we considered how to reduce the complexity of KAOS models using principles for effective communication. The current ongoing research involves analysis of the *i** framework languages and their supporting tools (TAOM4E, OME, and ST-Tool). We investigate scenarios which modellers could apply to create effectively communicating goal models. We also look for the visual cues (and supporting tool functionality) that facilitate preparation of the effective goal models and comprehension of the concepts provided in the *i** framework faster.

4 Conclusion and Future work

This paper presents an on-going research which aims to create the integrated GML. Currently we develop a metaCASE tool [9] using which we intend to generate a prototype tool supporting the integrated GML. The overall expected results of our study would contribute with (*i*) a thorough systematic scientific investigation and comparison of GMLs; and (*ii*) an integrated and tool-supported GML. The expected long-term benefits of GML analysis are improvement of the quality of the RE process. We hope to drive the research community towards a more rigorous way to define and extend (goal) modelling languages.

Acknowledgment. The author would like to acknowledge prof. Patrick Heymans and all the Requirements Engineering group at the University of Namur for contributing to and supporting this research.

References

1. Kavakli, E., Loucopoulos, P.: Goal Modeling in Requirements Engineering: Analysis and Critique of Current Methods. In: Information Modeling Methods and Methodologies. IDEA Group Publishing (2005)
2. Matulevičius, R., Heymans, P.: Comparison of Goal Languages: an Experiment. In: Proceedings of REFSQ 2007, Springer-Verlag Berlin Heidelberg (2007) 18–32
3. Matulevičius, R., Heymans, P., Opdahl, A.L.: Comparing GRL and KAOS Using the UEML Approach. In: Enterprise Interoprability II. New Challenges and Approaches. Springer-Verlag (2007) 77–88
4. Anaya, V., Berio, G., Harzallah, M., Heymans, P., Matulevičius, R., Opdahl, A.L., Panetto, H., Verdecho, M.J.: The Unified Enterprise Modelling Language – Overview and Further Work. In: Accepted at the 17th IFAC World Congress as invited keynote paper. (2008)
5. Mayer, N., Heymans, P., Matulevičius, R.: Design of a Modelling Language for Information System Security Risk Management. In: Proceedings of RCIS 2007. (2007) 121–131
6. Matulevičius, R., Mayer, N., Heymans, P.: Alignment of Misuse Cases with Security Risk Management. In: Proceedings of SREIS 2008, IEEE Computer Society (2008)
7. Matulevičius, R., Heymans, P., Sindre, G.: Comparing Goal-modelling Tools with the RE-tool Evaluation Approach. Journal of Information Technology and Control **35A**(3) (2006) 276–284
8. Matulevičius, R., Heymans, P.: Visually Effective Goal Models using KAOS. In: Proceedings of RIGiM 2007, Springer-Verlag Berlin Heidelberg (2007) 265–275
9. Englebert, V., Heymans, P.: Towards More Extensive MetaCASE Tools. In: Proceedings of CAiSE 2007, Springer-Verlag Berlin Heidelberg (2007) 454–468