A First Attempt to Express KAOS Refinement Patterns with Event B

Abderrahman Matoussi, Frédéric Gervais, and Régine Laleau

LACL, Université Paris-Est
{abderrahman.matoussi,frederic.gervais,laleau}@univ-paris12.fr

It is now recognised that goals play an important role in requirements engineering process, and consequently in systems development process. Whereas specifications allow us to answer the question "WHAT the system does", goals allow us to address the "WHY, WHO, WHEN" questions [1]. Up to now, the development process associated with formal methods, including Event B, begins at the specification level. Our objective is to include requirements analysis within this process, and more precisely KAOS [2] which is a methodology to implement goal-based reasoning. Existing work [3, 4] that combine KAOS with formal methods generate a formal specification model from a KAOS requirements model. We aim at expressing KAOS goal models with a formal language (Event B), hence staying at the same abstraction level. Our work is based on a constructive approach in which Event B models are built incrementally from KAOS goal models, driven by goal refinement patterns [1]. Since a KAOS goal means that a property must be established, the main idea is to represent each goal as a B event and the property as the post-condition of this B event. Up to now, we consider refinement patterns defined with first-order logic. Patterns with LTL temporal logic will be studied in further work. Thus, the general form of the assertion associated to a goal G is $P \rightarrow Q$ (P and Q are predicates, $\rightarrow$ is the logical implication). The *THEN* part of the B event corresponding to G is the translation of this assertion into Event B. At the most abstract level, the guard of the event related to the parent goal is always set to *True* to express that the event is always feasible. The definitive guard is built during the refinement process; i.e. after processing the different sub-goals. Proof obligations of Event B allow most of the KAOS refinement conditions to be verified. However, for some KAOS patterns as the case-driven tactics [2], additional constraints must be identified. Our current work is still partial and we are working on its extensions.

References

1. R. Darimont and A. van Lamsweerde. Formal Refinement Patterns for Goal-Driven Requirements Elaboration. In *SIGSOFT '96*, pages 179–190, San Francisco, California, USA, October 1996. ACM SIGSOFT.
2. E. Letier. Reasoning About Agents in Goal-Oriented Requirements Engineering. Ph.D. Thesis, ftp://ftp.info.ucl.ac.be/pub/thesis/letier.pdf, 2001.
3. H. Nakagawa and K. Taguchi and S. Honiden. Formal specification generator for KAOS. In *ASE 2007*, pages 531–532, Atlanta, USA, November 2007, ACM.
4. C. Ponsard and E. Dieul. From Requirements Models to Formal Specifications in B. In *REMO2V'2006*, Luxembourg, June 2006.