

# Benefits of Implementing A Query Language in Purely Functional Style

Artúr Poór, Gábor Páli, Tamás Kozsik, Melinda Tóth

Department of Programming Languages and Compilers  
Eötvös Loránd University

poor\_a@inf.elte.hu, pgj@elte.hu, kto@elte.hu, tothmelinda@elte.hu

Software maintenance can be significantly facilitated with a source code analyzer tool, especially if the gained information is made available through a query language. Thus intuition, consistency, and reliable implementation of query languages are becoming important, therefore it is recommended to develop a good practice that could be applied in their design and implementation in order to address those aims. In this paper, we present such a practice through the redesign of the query language of the static analyzer tool RefactorErl, as well as examples of semantic queries. The characteristics of the new language are its purely functional style, easy-to-learn syntax, and its formal definition in Haskell.

## References

- [1] Zoltán Horváth, László Lövei, Tamás Kozsik, Róbert Kitlei, Anikó Nagyné Víg, Tamás Nagy, Melinda Tóth and Roland Király: Modeling semantic knowledge in Erlang for refactoring. In Knowledge Engineering: Principles and Techniques, Proceedings of the International Conference on Knowledge Engineering, Principles and Techniques, KEPT 2009. Volume 54, Studia Universitatis Babeş-Bolyai, Series Informatica, pages 7–16, July 2009.
- [2] Paul Hudak and Joseph H. Fasel: A Gentle Introduction to Haskell. SIGPLAN Notices, 27(5), pages 1–52, May 1992.