Results of metric based transformations[*]

Roland Király

Eszterházy Károly Collage
kiraly.roland@aries.ektf.hu

Abstract

In this paper I am going to present how to create an analyzer that reveals inadequate programming style or overcomplicated erlang [8, 9] program constructs during the whole lifecycle of the code using complexity measures describing the program.

The algorithm [4, 6, 7], which I present here is also based upon the analysis of the semantic graph built from the source code (which, in this case, is naturally based on the measurements of structural complexity), but at this stage we can define default complexity measures, and these defaults are compared to the actual measured values of the code, and so the differences can be indicated.

The impact analysis of the transformations is theoretical but based on conclusions, as these statements are proven by running tests and interpreting the results on the implemented analyzing prototype.

On the other hand I show the algorithm measuring code complexity in Erlang programs, that provides automatic code transformations based on these measures. I created a script language that can calculate the structural complexity of Erlang source codes, and based on the resulting outcome providing the descriptions of transformational steps. With the help of this language we can describe automatic code transformations based on code complexity measurements.

I define the syntax [5] of the language that can describe those series of steps in these automatic code refactoring that are complexity measurement [2, 3] based, and present the principle of operation of the analyzer and runtime providing algorithm. Besides the introduction of the syntax and use cases, I present the results we can achieve using this language.

After defining and implementing the language [12, 13, 14], I create several scripts solving source code transformational problems that are based on complexity measures, and present their run-time results, proving their usability.

The transformation scripts created by me were applied on the source code of large programs, too. In this paper I present the results I measured during

---

the run time of transformation scripts to prove the validity of the analysing algorithm and language.

Keywords: software metrics, complexity metrics, source code, semantic graph, refactorerl

References

[1] McCabe T. J. A Complexity Measure, *IEE Trans. Software Engineering, SE-2(4), pp.308-320 (1976)*

[2] Zoltán Porkoláb, Ádám Sipos, Norbert Pataki, Structural Complexity Metrics on SDL Programs. *Computer Science, CSCS 2006, Volume of extended abstracts, (2006)*

[3] Zoltán Porkoláb Programok Strukturális Bonyolultsági Méröszámai. *PhD thesis Dr Töke Pál, ELTE Hungary, (2002)*

[4] Zoltán Horváth, Zoltán Csörnyei, Roland Király, Róbert Kitlei, Tamás Kozsik, László Lövei, Tamás Nagy, Melinda Tóth, and Anikó Víg.: Use cases *for refactoring in Erlang, To appear in Lecture Notes in Computer Science, (2008)*

[5] Csörnyei Zoltán *Fordítóprogramok* Typotex Kiadó, Budapest, 2006. 3

[6] R. Kitlei, L. Lövei, M Tóth, Z. Horváth, T. Kozsik, T. Kozsik, R. Király, I. Bozó, Cs. Hoch, D. Horpácsi.: Automated Syntax Manipulation in RefactorErl. *14th International Erlang/OTP User Conference. Stockholm, (2008)*

[7] Lövei, L., Hoch, C., Köllö, H., Nagy, T., Nagyné-Víg, A., Kitlei, R., and Király, R.: Refactoring Module Structure *In 7th ACM SIGPLAN Erlang Workshop, (2008)*

[8] Lövei, L., Horváth, Z., Kozsik, T., Király, R., Víg, A., and Nagy, T.: Refactoring in Erlang, a Dynamic Functional Language *In Proceedings of the 1st Workshop on Refactoring Tools, pages 45–46, Berlin, Germany, extended abstract, poster (2007)*

[9] Erlang - Dynamic Functional Language
http://www.erlang.org

[10] T. Kozsik, Z. Horváth, L. Lövei, T. Nagy, Z. Csörnyei, A. Víg, R. Király, M. Tóth, R. Kitlei.. Refactoring Erlang programs. *CEFP'07, Kolozsvár (2007)*

[11] Király Roland, *Funkcionális programozási nyelvek* EKF TTK TAMOP412 2010 120 oldal.

[12] Király, R., Kitlei R.: *Complexity measurments for functional code* 8th Joint Conference on Mathematics and Computer Science (MaCS 2010) refereed, and the proceedings will have ISBN classification July 14-17, 2010

[13] KIRÁLY, R., KITLEI R.: *Implementing structural complexity metrics in Erlang.* '10 ICAI 2010 – 8th International Conference on Applied Informatics to be held in Eger, Hungary January 27-30, 2010

[14] KIRÁLY, R. AND KITLEI, R.: *Implementing structural complexity metrics for Erlang* Poster on the 8th International Conference on Applied Informatics, ICAI 2010, 2010

Roland Király
Eszterházy Károly Collage, Institute of Mathematics and Informatics