Research in Computer Science
Project Description

Alexandru Dima
adima@student.ethz.ch

March 16, 2010

A closure is a first-class function with free variables that are bound in its lexical environment. It is defined within the scope of its free variables, and the extent of those variables is at least as long as the lifetime of the closure itself. Closures make certain programming uses easier, while providing implicit state hiding. Verifying closures presents many challenges such as enabling specifications to allow describing the behavior of one method in terms of another, which makes program specification and verification more difficult.

Dynamic Frames Theory[1] enables decoupling of the alias control from the other formal constructs, like the class, the module or the object without any restrictions, by allowing the programmer to specify in a dynamic manner the frame of each method at the granularity of his choice.

Dafny[2], an experimental language for dynamic-frames specification, explores the dynamic frames style of specifications in an object-based sequential setting. The Dafny source code, via the intermediate verification language Boogie[3], is encoded for the SMT solver. This translation is done in such a way that the correctness of the resulting Boogie program implies the correctness of the original Dafny program.

The scope of this Research Project is to enhance Dafny's features by adding closures in an unobtrusive manner, meaning that the new syntactical constructs shall be consistent with the existing ones and that the source code added to the Dafny verification tool shall be consistent with the existing one. The resulting Boogie program generated from the Dafny source code shall translate closures into Boogie PL by loosely following the methodology described in the *Specification and Verification of Closures Technical Report*[4], but with the possibility of deviating from the methodology, given that the final verification of closures is correct.

The work can therefore be split in steps: defining the new Dafny syntactical constructs needed to express closures and their specifications in the programming language, adapting the *Specification and Verification of Closures Methodology*[4], implementing the first version of the closures verification (with static frames), and optionally extending this version to allow dynamic frames for closures.

The deliverables of this project shall be source code, accompanied by appropriate documentation, as well as tests which shall contain at least the three examples from [4]. The inclusion of the source code in the Dafny repository[3] is highly desirable.

References

[1] Ioannis T. Kassios. *Dynamic Frames: Support for framing, dependencies and sharing without restrictions*. In Jayadev Misra, Tobias Nipkow, and Emil Sekerinski, editors, FM 2006: *Formal Methods, 14th International Symposium on Formal Methods, volume 4085 of Lecture Notes in Computer Science*, pages 268-283. Springer, August 2006.

[2] K. Rustan M. Leino. *Specification and verification of object-oriented software*. In Manfred Broy, Wassiou Sitou, and Tony Hoare, editors, *Engineering Methods and Tools for Software Safety and Security, volume 22 of NATO Science for Peace and Security Series D: Information and Communication Security*, pages 231-266. IOS Press, 2009. Summer School Marktoberdorf 2008 lecture notes.

[3] http://boogie.codeplex.com

[4] [4] Ioannis T. Kassios and Peter Müller, *Specification and Verification of Closures*, Technical Report, ETH Zürich, 2010