



## An Efficient Implementation of Refinements for JRuby+Truffle

Master thesis for ...

Matr.Nr.: ...

Email: ...@...

JRuby+Truffle [1] is an effort to use the Truffle AST interpreter framework and the Graal dynamic compiler to implement the Ruby programming language. It is part of the existing JRuby implementation of Ruby on the JVM.

The Ruby programming language allows methods to be redefined at any point in the program execution. This is often called monkey patching - a name which indicates the mischief that this can cause in complicated programs. One approach to restricting the impact of redefining methods is refinements [2], which allow redefinition to be constrained within a lexical scope.

Unusually for the Ruby programming language, this new feature has some related work in the literature, in the form of classboxes [3]. However support in alternative implementations of Ruby is limited - there is an experimental branch in JRuby that doesn't use Truffle.

We would like to create an efficient implementation of refinements. Conventional dispatch in Ruby has zero peak runtime overhead under most conditions, and we should aim to achieve the same for refinements if possible.

The scope of this thesis is as follows:

- A correct implementation of refinements with the same behavior as standard Ruby.
- Peak runtime overhead low enough for refinements to be useful.

Optional goals are:

- Improve the Ruby specifications for refinements [5] to prove correctness.
- Zero peak runtime overhead - a call using refinements should be no slower a conventional call.

The work's progress should be discussed with the supervisor at least every 2 weeks. Please note the guidelines of the Institute for System Software when preparing the written thesis.

Supervisor: Dipl.-Ing. Matthias Grimmer, Chris Seaton MEng, Oracle Labs

[1] <https://github.com/jruby/jruby/wiki/Truffle>

[2] [http://www.ruby-doc.org/core-2.1.1/doc/syntax/refinements\\_rdoc.html](http://www.ruby-doc.org/core-2.1.1/doc/syntax/refinements_rdoc.html)

[3] Classboxes: A Minimal Module Model Supporting Local Rebinding, Alexandre Bergel, Staephane Ducasse, and Roel Wuyts, JMLC 03

[4] <https://github.com/jruby/jruby/tree/refinements>

[5] <https://bugs.ruby-lang.org/projects/ruby-trunk/wiki/RefinementsSpec>

[6] <https://github.com/rubyspec/rubyspec/tree/refinements>