



## Truffle/Clojure: An AST-Interpreter for Clojure

Master thesis for Thomas Feichtinger

Matr.-Nr.: 0855190

Email: t.feichtinger@gmail.com

Truffle is a novel modeling language for implementing managed languages in Java. The language implementer writes an abstract syntax tree (AST) interpreter, which is integrated into the Truffle framework.

The goal of the Truffle/Clojure project is to write a Clojure interpreter for Truffle. Clojure will be the first functional language implemented with Truffle. This thesis focuses on identifying functional programming concepts (for example tail recursion) which are not yet supported by the Truffle framework and implementing them.

This thesis should also determine the set of nodes that is needed for modeling a useful part of the Clojure language.

Specific sub-goals are:

- Identifying functional programming concepts not yet available to Truffle based interpreters and implementing them.
- Applying object-oriented design techniques when modeling the interpreter.
- Implementing a subset of Clojure such that the interpreter can be tested on established Clojure benchmarks.
- Exploring which Java language constructs are convenient for implementing a scripting language interpreter.

Explicit non-goals are:

- Completeness with respect to the standard Clojure implementation.
- Peak performance of executed code comparable to the standard Clojure implementation.

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: Dr. Thomas Würthinger, Dipl.-Ing. Matthias Grimmer