This repository
Search
Explore
Gist
Blog
Help
mhucka
Watch 16
Star 13
Fork 7
Code
Issues 19
Pull Requests 0
Wiki
Pulse
Graphs
HTTPS clone URL
https://github.com/!
Clone in Desktop
Download ZIP
branch: master
README.md

Sable / mclab

Compiler toolkit for MATLAB http://www.sable.mcgill.ca/mclab/

2,619 commits | 6 branches | 7 releases | 7 contributors

mclab / +

Switch Travis build to Oracle JDK 8.

isbadawi authored on May 23 latest commit 7a2fdd6849

languages	Simplify code using Java 8 features.	5 months ago
lib	Update Guava to 16.0.1.	7 months ago
.classpath	Update Guava to 16.0.1.	7 months ago
.gitignore	Add ctags files to .gitignore.	5 months ago
.project	Rename eclipse project to McLab instead of Matlab Source	2 years ago
.travis.yml	Switch Travis build to Oracle JDK 8.	5 months ago
LICENSE	.gitignore, copyright & license	2 years ago
README.md	Make build status image a link [ci skip]	2 years ago

You can clone with HTTPS, SSH, or Subversion.

McLab/Natlab

build passing

This is the Java-based infrastructure of the McLab project, which aims to provide compiler tools and infrastructure for MATLAB (and potentially other scientific languages in the future). In practice this repository houses development for the project's "static" branch, which includes the frontend (parsing, static analysis, refactoring, etc.), and work towards static compilation. (The "dynamic" branch's McVM project will be made available soon.)

Overview

Proper developer documentation will eventually be here. A quick tour of the code:

- `matlab` includes the Matlab to Natlab translator. Natlab (nice Matlab) is a simplified version of Matlab that is easier to parse. (There are only syntactic differences between the two.)
- `natlab` includes the generated Natlab parser, as well as `natlab.Main`, the entry point.
- `natlab.toolkits.rewrite` is a simple framework for AST transformations and simplifications. Some useful simplifications, such as conversion to three-address code, are provided.
- `natlab.toolkits.analysis` is an (intraprocedural) dataflow analysis framework, and subpackages provide various common analyses. Of particular interest is the kind analysis, which lives in `natlab.toolkits.analysis.varorfun`.
- `natlab.refactoring` includes implementations of a few different refactorings, such as function and script inlining.
- `natlab.tame` and its subpackages comprise the tamer framework, which aims to make Matlab more suitable to static compilation. It provides an intermediate representation (`natlab.tame.tir`), machinery for analyses to handle Matlab builtins, and an interprocedural analysis framework, among other things.
- `natlab.backends` houses the Fortran and X10 backends. These build on top of the tamer, and are a work in progress.

Working with the code

There exists an eclipse project (`.project`, `.classpath`) which can be imported in eclipse.

There exist ant files (`build.xml`). These can either build the projects on the command line, or in eclipse (import in the `ant-view`) using the `eclipse.*` targets. The ant targets generate necessary code (like the Natlab AST). The command line ant targets also compile everything, while the eclipse targets do not -- eclipse will compile the code.

(The build process contains a lot of cruft, duplication, some things are broken, etc. There is an ongoing effort to clean it up.)

It is useful for analyses to have access to the library files of a Matlab installation. After `Natlab.jar` has been built with `ant jar`, you can run the `registerMatlabPathWithNatlab.m` script in Matlab; it just calls the Matlab `path` function and feeds its output to the jar.

Disclaimer

Some of this code is confusing, poorly documented, and not well understood, as the grad students who wrote much of it have moved on to bigger and better things. (This is fairly typical of research software.) Most of the documentation can be found in the various McLab publications, but in many cases there are discrepancies between terminology used in papers and what's in the code. There is an ongoing effort to clean up the code and the build process, and document everything, so please bear with us as we work through this.

Copyright and License