A Self-Dual Distillation of Session Types: Mechanized Proofs (Artifact)

Jules Jacobs
Radboud University Nijmegen, The Netherlands

Abstract

This is the artifact description for the paper "A Self-Dual Distillation of Session Types". The artifact consists of mechanized proofs of the theorems listed in the paper, in the Coq proof assistant.

2012 ACM Subject Classification Software and its engineering → Concurrent programming languages

Keywords and phrases Linear types, concurrency, lambda calculus, session types

Digital Object Identifier 10.4230/DARTS.8.2.15

Related Article Jules Jacobs, "A Self-Dual Distillation of Session Types", in 36th European Conference on Object-Oriented Programming (ECOOP 2022), LIPIcs, Vol. 222, pp. 23:1–23:22, 2022. https://doi.org/10.4230/LIPIcs.ECOOP.2022.23

Related Conference 36th European Conference on Object-Oriented Programming (ECOOP 2022), June 6–10, 2022, Berlin, Germany

Evaluation Policy The artifact has been evaluated as described in the ECOOP 2022 Call for Artifacts and the ACM Artifact Review and Badging Policy.

1 Scope

All theorems listed in the paper are formally proved in the artifact, using the Coq proof assistant.

2 Content

The artifact consists of Coq source code in the form of .v proof files. The proof files are in the `theories` subdirectory. The files most relevant to the paper are:

lambdabar/langdef.v definition of the λ language, operational semantics, and type system.
lambdabar/definitions.v the definitions about deadlock freedom / reachability / global progress corresponding to those in the paper.
lambdabar/theorems.v the proofs of the theorems in the paper.
lambdabar/sessions.v the encoding of session types in λ.
lambdabar/compiler.v the translation from GV to λ and the simulation proof.

Additionally, these proofs make use of lemmas proved in the following files and directories:

lambdabar/langtools.v imports the required libraries
lambdabar/rtypesystem.v definition of the run-time type system.
lambdabar/invariant.v definition and proofs about the configuration invariant.
cgraphs/.v** definitions and lemmas about graphs and separation logic, a modified version of [1].

These files are internal details of the proofs, which are checked by Coq, so one does not need not check their correctness.

3 Getting the artifact

The artifact endorsed by the Artifact Evaluation Committee is available free of charge on the Dagstuhl Research Online Publication Server (DROPS). The latest version of the artifact is available at `https://github.com/julesjacobs/cgraphs`.

To test the artifact, you need a recent version of Coq, and an installation of Iris. The artifact has been tested with Coq 8.15.0 and Iris dev.2022-02-21.0.96883dbd.

The installation instructions for Coq can be found at: `https://coq.inria.fr/download`

The installation instructions for Iris can be found at: `https://gitlab.mpi-sws.org/iris/iris/`

If available on your platform, I advise using opam (OCaml package manager) to install Coq and Iris. On Unix-like platforms it can probably be installed with your OS' package manager, or on OS X with 'brew install opam'. In that case you do not need to install Coq separately; opam will install it when installing Iris.

After installing these, the development can be built with 'make'. This will let Coq check all the definitions and theorems.

4 Tested platforms

The artifact works on any platform supported by Coq. It has been tested on OS X.

5 License

The artifact is available under the BSD license.

6 MD5 sum of the artifact

2773d605bf6abf67325281499f3c5224

7 Size of the artifact

537 KB

References

1 Jules Jacobs, Stephanie Balzer, and Robbert Krebbers. Connectivity graphs: a method for proving deadlock freedom based on separation logic. *Proc. ACM Program. Lang.*, 6(POPL):1–33, 2022. `doi:10.1145/3498662`.