

# TIL-Script Typechecker

Filip Peterek

Technical University of Ostrava

June 2022

# Project goals

- ▶ Create a working typechecker for TIL constructions
- ▶ Plot typechecking graph to an image
- ▶ Write extensible code to allow further development

# Accomplished goals

- ▶ Working typechecker
- ▶ Graph plotting
- ▶ TIL-Script parser
- ▶ Context recognition

# Implementation

- ▶ Kotlin
- ▶ Gradle
- ▶ Antlr

# Parser Implementation

- ▶ Existing TIL-Script grammar has been utilized
  - ▶ Slight tweaks were necessary to make it work with Antlr
- ▶ Parser is generated using Antlr
- ▶ Resulting AST is converted to a custom structure

# Typechecker Implementation

- ▶ For typechecking to succeed, all names must be defined
- ▶ This is ensured in a separate step
- ▶ If at least one undefined name is found, processing ends
- ▶ Errors are printed to stdout

# Typechecker Implementation

- ▶ Recursive implementation
- ▶ Aliases are expanded, expansions are compared
- ▶ Double execution is ignored
- ▶ Trivializations force construction of independent scopes
- ▶ Type deduction is unsupported
- ▶ Errors are, yet again, printed to stdout

# Context Recognition

- ▶ Context recognition is also supported
- ▶ Recursive implementation
- ▶ No errors can be detected at this point



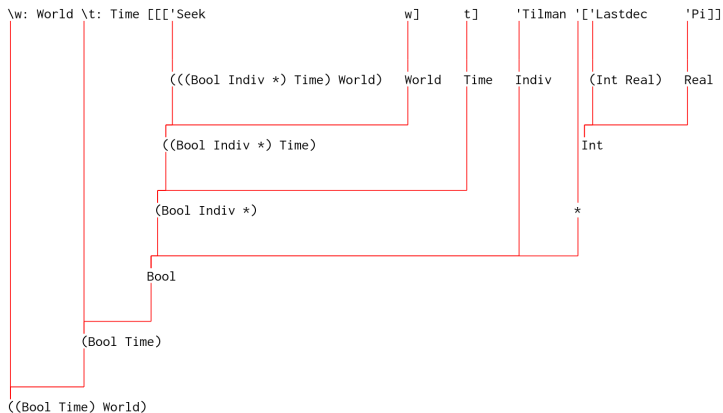
# Program Output

- ▶ Errors are printed to stdout
- ▶ Script information is written to a JSON file
  - ▶ Constructions
  - ▶ All subconstructions of said constructions
  - ▶ Definitions
  - ▶ Constructed type
  - ▶ Type of TIL-Script statement
  - ▶ Context
- ▶ Typechecking graph is created as an SVG image

# Graph Plotting

- ▶ No third party libraries
- ▶ Indentation is necessary due to lengths of type ids
- ▶ Monospace font

# Graph Plotting



# Possible Improvements

- ▶ LSP
  - ▶ Text editor integration
  - ▶ Code completion
  - ▶ Error reporting
  - ▶ Refactoring
  - ▶ Supported by all major editors (Vim, Emacs)

# Possible Improvements

- ▶ Interactive browser application
  - ▶ Web application available online
  - ▶ Allows online text editing
  - ▶ Provides on-the-fly syntax and type coherence checking
  - ▶ Opportunity to utilize graphs

# The End