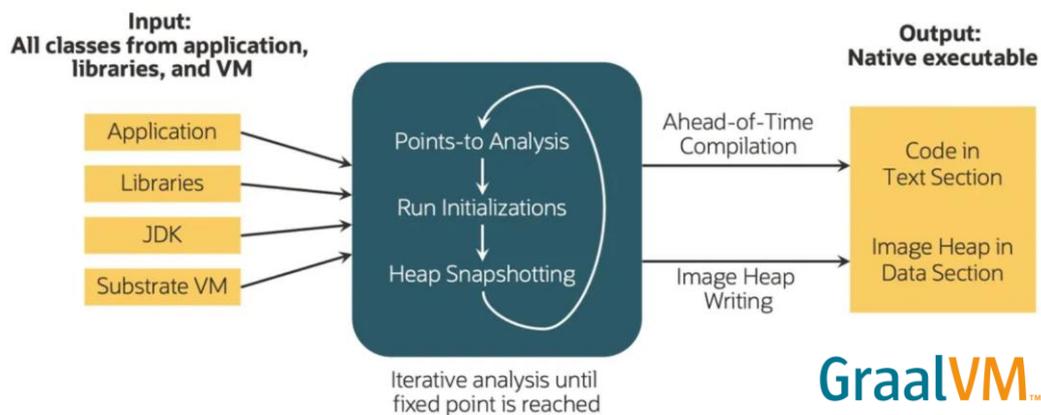


Tool Support for GraalVM Native Image

Master Project Proposal, Winter Term 2022/23
Software Architecture Group, Prof. Dr. Robert Hirschfeld

GraalVM Native Image [1] is a technology that enables ahead-of-time (AOT) compilation of Java applications. AOT compilation can reduce startup times and memory footprint significantly [2], allowing Java to be used in cloud functions and other setups that are usually dominated by static languages. On the one hand, this allows developers to use a high-level dynamic language for development. On the other hand, AOT compilation requires additional information about reflection and other dynamic mechanisms. While some of this additional information can be determined by the AOT analysis automatically, a manual configuration on the side of the developer is still required in some cases. Misconfiguration, however, can bloat the file size of AOT executables unnecessarily and can even lead to misbehavior and crashes at run-time.



In this project, we will explore tool support for GraalVM Native Image. The goal is to investigate tooling ideas and build prototypes that help developers in building AOT executables with GraalVM Native Image that are small, fast, and reliable. Students should be familiar with the Java programming language and interested in tool development and program analysis.

Possible tooling ideas that could be explored as part of this master's project include:

- A visualization tool for the points-to analysis [3] performed by GraalVM Native Image, allowing developers to understand dependencies within their applications interactively
- An implementation of the language server protocol (LSP) [4] that supports developers with immediate feedback from the points-to analysis and other aspects of GraalVM Native Image

Related Work

- [1] GraalVM Native Image (<https://www.graalvm.org/native-image/>), 2022.
- [2] Alina Yurenko, *Revolutionizing Java with GraalVM Native Image* (<https://www.infoq.com/articles/native-java-graalvm/>), InfoQ, 2022.
- [3] Wimmer et al., *Initialize Once, Start Fast: Application Initialization at Build Time* (<https://dl.acm.org/doi/10.1145/3360610>), OOPSLA, 2019.
- [4] Microsoft, *Language Server Protocol* (<https://microsoft.github.io/language-server-protocol/>), 2021.

Contact

Prof. Dr. Robert Hirschfeld, Dr. Jens Lincke, Fabio Niephaus,
Software Architecture Group, Potsdam
<http://www.hpi.uni-potsdam.de/swa>, {first.last}@hpi.uni-potsdam.de