Enabling Multi- and Cross-Language Verification with LLVM

Jack J. Garzella, Marek Baranowski, Shaobo He, Zvonimir Rakamarić

• Many verifiers are made to support only C programming language
• Nowadays programs are written in many different languages and their combinations
• New languages are being invented (e.g., Rust, Swift) to improve ease of development, security, and/or performance
• Verification tooling for these new languages does not exist

Strengths
• Modular architecture that uses LLVM avoids the need for front-end development
• New language can be added with modest amount of development effort

Challenges
• Modeling of standard libraries and large runtimes is challenging and time-consuming

SMACK Toolflow Diagram

This work was supported by the University of Utah Office of Undergraduate Research and National Science Foundation