
Introduction

Word budget: ~500–600 words

Explain the main motivation for the project
Show how the work fits into the broad area of surrounding computer science
brief survey of previous related work
-> original emscripten (compiler LLVM to JS)
-> various compilers to wasm, including from LLVM which would do C

1.1 Background and Motivation

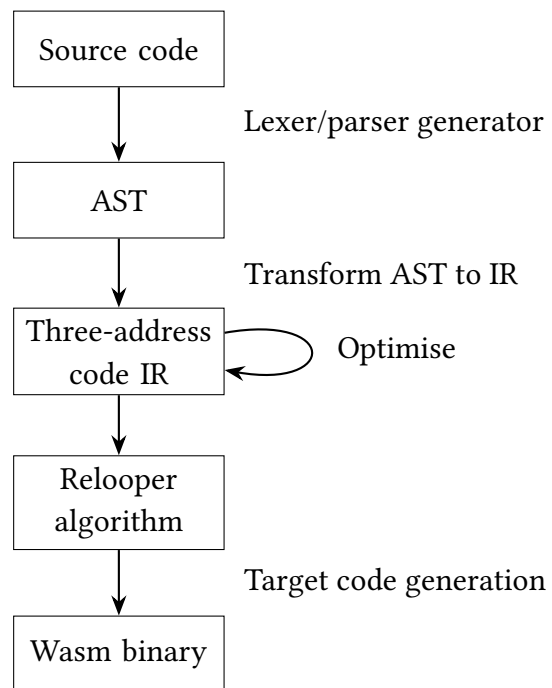
Increasingly in modern society, more and more applications are shifting to cloud computing as one of the primary ways of interacting with computers. We are experiencing a transition away from traditional native applications and towards performing the same tasks in an online environment. However, the standard approach of building web apps with JavaScript fails to deliver the performance necessary for many intensive applications.

WebAssembly aims to solve this problem by bringing near-native performance to the browser space. It is a virtual instruction set architecture, which executes on a stack-based virtual machine. Per the Introduction section of the WebAssembly specification [1], it is designed to have “fast, safe, and portable semantics” and an “efficient and portable representation”. The next two paragraphs expand on what this means.

The semantics are designed to be able to be executed efficiently across different hardware, be memory safe (with respect to the surrounding execution environment), and to be portable across source languages, target architectures, and platforms.

The representation is designed with the primary target of the web in mind. It is designed to be compact and modular, allowing it to be efficiently transmitted over the Internet without slowing down page loads. This also includes being streamable and parallelisable, which means it can be decoded while still being received. It supports both just-in-time and ahead-of-time compilation.

1.2 Project Objectives



1.3 Survey of Related Work