

# LLVM

Kamil Niewiara

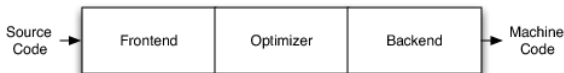
Wydział Matematyki, Fizyki i Informatyki  
Politechnika Krakowska im. Tadeusza Kościuszki

5 lipca 2019

LLVM (skrót od Low Level Virtual Machine) to pierwotnie maszyna wirtualna napisana w języku C++, która została zaprojektowana w 2000 roku przez Vikrama Advego oraz Chrisa Lattnera. Jednak celem projektu było przede wszystkim stworzenie narzędzi ułatwiających optymalizację i analizę kodu.



Rysunek: Logo projektu.

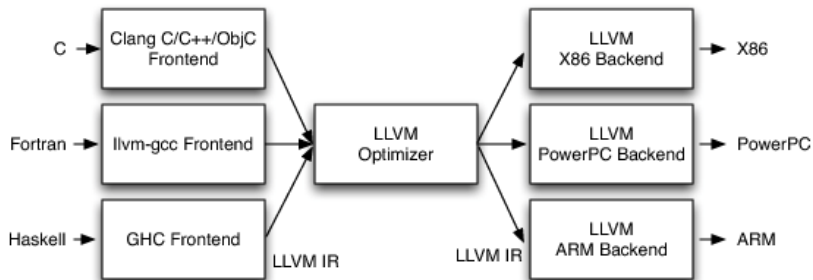


Rysunek: Budowa kompilatora. Schemat można znaleźć w [1].

Kompilator wykonuje trzy etapy:

- front-end - przyjmowany jest tutaj kod źródłowy
- middle-end - generowany jest odpowiadający instrukcjom języka kod maszynowy, dołączane są biblioteki, dokonuje się optymalizacja
- back-end - tworzony jest plik wykonywalny

Wirtualna maszyna LLVM, zajmuje się tylko drugim i trzecim etapem, front-endem jest tzw. Clang.



Rysunek: Architektura LLVM. Schemat można znaleźć w [1].

- Clang to front-end dla projektu LLVM, kompilujący języki C, C++ i Objective C. Jego zgodność z interfejsem GCC sprawia, że może go zastąpić. Do jego zalet należy generowanie rozbudowanych komunikatów o błędach, co umożliwia ich szybsze wykrywanie i usuwanie, a także szybsza kompilacja i bardziej zoptymalizowany kod. Clang ma również możliwość dynamicznej kompilacji. Liczne rozszerzenia i modułarna budowa umożliwiają lepszą integrację ze środowiskami IDE.
- Architektura LLVM cechuje się dużą elastycznością. Front-endem mogą być również języki takie jak Haskell, Go, czy Fortran.

Optymalizacja kodu ma za zadanie poprawić szybkość jego wykonywania oraz zmniejszyć rozmiar pliku wynikowego. Optymalizacja LLVM przeprowadzana jest we własnym formacie kodu, co sprawia, że niezależnie z jakiego języka pochodzi kod, mechanizmy optymalizacji są takie same. Następnie kod jest tłumaczony na kod maszynowy konkretnej architektury, czyli na back-end.

Intermediate Representation (reprezentacja pośrednia) to język maszyny wirtualnej LLVM. Jest on dostępny w czytelnej postaci, co ułatwia pisanie front-endów i back-endów.

- IDE Xcode firmy Apple ma u swojej podstawy LLVM. Firma ta wykorzystując dynamiczną kompilację i modularność stworzyła również framework OpenCL, który umożliwia pisanie kodu niezależnie od środowiska sprzętowego.
- Google Chrome na Androidzie, Linusie i macOS, a od 2018 także na Windows 10 korzysta z Clanga.
- Z LLVM korzystają również takie firmy jak Qualcomm Innovation Center i Intel oraz ośrodki naukowe, takie jak CERN i Argonne National Laboratory.



- szybkość kompilacji
- czytelność komunikatów o błędach
- czytelna i łatwa w zrozumieniu reprezentacja pośrednia(IR)
- wiele gotowych modułów optymalizacji, łatwość modyfikacji

- [1] <http://www.aosabook.org/en/llvm.html>
- [2] <https://llvm.org/>
- [3] <https://grzesieklog.blogspot.com/2013/03/llvm-i-linux.html>