

異種実行環境へのスクリプト言語の抽象コード生成

松村哲郎 志田俊介 井出真広 倉光君郎
横浜国立大学

近年、スクリプト言語は、自前のインタプリタ上だけでなく、JavaVM, CLR や LLVM, JavaScript など様々な実行環境上で実行する要望が増えている。従来、これらの要望に応えるため、JRuby や IronRuby のように各ターゲットに向けて個別の処理系を実装してきた。本ポスターでは、MiniKonoha を実装例にして、ひとつのパース実装処理系から ETRoboCon 向け小型 VM, JavaScript, Common Language Runtime へのコード生成とその抽象化を紹介する。

Multi-target code generation for scripting language

Tetsuro Matsumura Shunsuke Shida Masahiro Ide Kimio Kuramitsu
Yokohama National University

1. はじめに

近年、スクリプト言語を独自のインタプリタ上だけでなく、JVM、CLI、LLVM、JavaScript など、様々な実行環境で実行する要望が増えている。従来、これらの要求に応えるため、JRuby や IronRuby のように各ターゲットに向けて個別の処理系を実装してきた。しかし、このような場合、同じスクリプト言語のパース系であっても実装ごとに用意する必要がある。

2. 提案

本研究では、ソースコードを抽象構文木(AST)の形に変換し、1つのASTを元に複数の環境へ向けたコードを生成する。この手法は、各ターゲットに向けて個別の処理系を実装することに比べて、処理の前段階が共通化できるため、実装がより簡単に行える利点がある。

2.1 MiniKonoha

本ポスターで扱うスクリプト言語である MiniKonoha 言語について述べる。

MiniKonoha 言語処理系は機能をモジュール化し、利用者が拡張しやすいよう設計を行なっている。MiniKonoha 言語処理系を構成するモジュールを以下に示す。

- ・コード評価器
- ・コード生成器
- ・型検査器
- ・パーサ
- ・メモリ管理機構

本研究では MiniKonoha のコード生成器、コード評価器を切り替えることで、様々な環境へ向けたコード生成を実現する(図1)。

2.2 中間表現

我々はソースコードから各環境のコードを生成する際の中間表現として MiniKonoha の AST を利用している。これは、他言語のソースコードを生成する際に元のプログラムの構造を残すことができるためである。

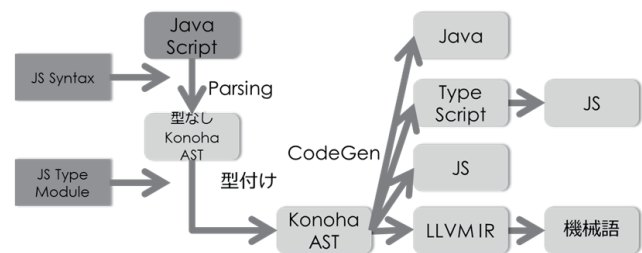


図1 MiniKonoha の処理の流れ

2.3 コード生成、評価

コード生成器は MiniKonoha のコード生成モジュールとして実装される。コード生成器の出力は、大きく分けてバイトコードとソースコードの2つがあり、前者はJVM、後者はJavaScriptのコード生成などが該当する。ソースコードは、中間表現にASTを利用しているため、元のプログラムの構造を残した物が生成可能である。

コード評価器もモジュールとして実装される。JavaScriptのコード生成を例にとると、V8などのJavaScriptエンジンをコード評価器として組み込み、実行させることが可能である。

実装例として、組み込み機器で動作する ETRoboCon 向け小型 VM、JavaScript、CLR のコード生成器があり、同一のスクリプトが各環境で動作している。

3. まとめ

我々はコード生成・評価モジュールを切り替え、単一のパーサから各環境向けのコード生成を行う方法を提案した。

謝辞 本研究は、JST/CREST「実用化を目指した組み込みシステム用ディペンダブル・オペレーティングシステム」領域の研究助成を受けて行われた。

参考文献

- 1) KonohaScript, <http://konohascript.org>
- 2) Kuramitsu, K.: Konoha = MiniKonoha + Sugar, 情報処理学会・プログラミング研究会, Information Processing Society of Japan (IPSJ).
- 3) Kuramitsu, K.: Konoha - Implementing a Static Scripting Language with Dynamic Behaviors, Workshop on Self-sustaining Systems 2010, S3, The University of Tokyo, Japan, September 2010.
- 4) 志田 俊介, 菅谷 みどり, 倉光 君郎: Tiny Konoha: ET ロボコン向けのスクリプト処理系の簡素化, 情報処理学会, Information Processing Society of Japan (IPSJ).

