

発表概要

RISC 向けの高性能中間コードによるマルチプラットフォーム 実行環境の実現

黄 顔 上[†] 鈴木 貢^{††} 渡 邊 坦^{††}

ネットワークの発展により、プログラムを非ソース形態により異機種間で配布、共有する需要が高まっている。仮想マシンを定義して、それによってプログラムを配布するのが一つの解であるが、従来はスタックマシンを用いるのが定例であった。Java は上記の環境を実現しており、コード量が小さい代わりにレジスタ情報を持たないため、現在のアーキテクチャの一つの流れである RISC 方式には不向きで、マシンコードへの変換に時間がかかり、実行効率が悪い。本論文では、レジスタ数に制限を設けない、コンパクトな抽象レジスタマシン CompactParmCode (Parallel Abstract Register Machine Code) をそのための中間言語として提案する。転送する中間コードのファイルには、記号情報、操作情報以外にレジスタ情報を含むため、マシンコードを生成する際、レジスタ最適化作業を大部分省略でき、生成時間が短く優れたコードを生成できる。byte code に比べて中間コードの情報量が多いが、ファイル全体を静的に走査し、命令列の類似パターンを検出し、それを小サイズ命令へ変換することで縮小する。転送ファイルはソースプログラムと比べて 2 倍以内に収まり、実行速度はソースプログラムから直接コンパイルしたものと同等の性能を得られることを確認した。

Experimental Development of Multi-Platform Execution Environment Using High Performance Intermediate Code for RISC Architectures

HUANG YEN SAN,[†] MITUGU SUZUKI^{††} and TAN WATANABE^{††}

Recent progress of network systems has increased a requirement for using same programs on many platforms in non-source code style. Java realized such environment by using a virtual stack machine independent on target platforms. Java byte code has no register information, so it is slow in compiling and execution for RISC architecture. In this paper, we propose a compact abstract register machine code (CompactParmCode) for realizing the multi-platform environment. CompactParmCode is fast in machine code generation and execution because of its optimized register information. It is made compact by finding same pattern composed of several codes and replacing them to one code of small size. The size of CompactParmCode file is less than 2 times compared to source file size. Its execution speed is the same to that of the code directly generated from source code.

(平成 11 年 3 月 23 日発表)

[†] 電気通信大学大学院情報工学専攻
Computer Science, University of Electro-Communications

^{††} 電気通信大学情報工学科
University of Electro-Communications