在翻譯爪哇中介碼成 X86 組語環境下的動態呼叫實作

學生：陳俊元　　指導教授：楊 武 博士

國立交通大學資訊科學研究所

摘要

為了達到跨平台的特性，Java 程式會被編譯成一種以堆疊操作為導向，與硬體無關的中間型式，這種中間型式稱為 Bytecode，一般執行 Bytecode 的工具是直譯器。Java 程式語言雖然能符合在網路環境下發展程式的特性，但用直譯器執行 Bytecode 的過程十分緩慢，其速度無法和傳統語言如 C 和 C++相比。為了改善效能問題，因而衍生出幾種改進方式。如：JIT 編譯器、Native Compiler、Java chip。

本篇論文是屬於 Native Compiler 範疇，提出了一種將 Bytecode 轉成 X86 組合語言的方法，研發出來的翻譯器可以將類別檔轉成 X86 的組合語言，我們將詳細介紹 Java 的執行環境與 Native Compiler 的理論，並說明翻譯器如何把 JVM 的 stack machine 對應至 register machine，及如何翻譯各種類型的 JVM 指令（其中包括複雜的動態呼叫），及如何做一些基本的最佳化。最後會將此方法與 SUN 提出的 Interpreter 及 JIT 編譯器做效能比較，然後討論影響執行效能的原因。根據實驗結果，翻譯出來的組合語言有不錯的執行效能。

Dynamic Dispatch Implementation

Based On Translating Java Bytecode To X86 Assembly Environment

Student: Chun-Yuan Chen Advisor: Dr. Wuu Yang

Institute of Computer and Information Science

National Chiao Tung University

Abstract

For the purpose of platform-indepentdent, a Java program is compiled to an intermediate form, bytecode, which is stack-oriented and hardware-independent. The common tool of bytecode which is executed is interpreter. The performance of such an interpreter is less comparable to object coded generated by programming languages such as C and C++. To improve the performance, lead to some improvement types. For example, JIT compiler, Native Compiler, Java chip.

This paper belong to a category of Native Compiler. A method of translating bytecode to X86 assembly code is proposed. Translator which is researched can translate class file to X86 assembly code. We will introduce the execution environment and Native Compiler theory, and explain those translations of every type of bytecode by translator (including dynamic dispatch), and do some basic optimization. Finally, we will compare the performance of our translator with SUN interpreter and JIT, then discuss the factors that affect performance. Experimental results show that generated assembly code has good performance.

Acknowledgments

First of all, I deeply appreciate to my advisor, Dr. Wuu Yang, for all that he has done. Without his careful guidance and suggestion, it is impossible to accomplish this thesis.

Then, I would like to thank all the members in the PLAS laboratory for their encouragement. Their friendliness made me feel not lonely in the past two years.

Furthermore, I am grateful to my girl friend, Ching-Wen Chang. When I am discouraged by setbacks, she always spirits me up with her love and tolerance.

Finally, I would dedicate this thesis to my dear family, especially my parent. Without their support in spirit, I would not be able to get my M.S. degree. I want to offer my heartfelt thanks to them.

Contents

List of Figures

List of Tables