A Compilation and Optimization Model for Aspect-Oriented Programs

Hidehiko Masuhara[†] Gregor Kiczales[‡§]
Chris Dutchyn[§]
[†]Graduate School of Arts and Sciences, University of Tokyo
[‡]Intentional Software Corporation
[§]Department of Computer Science, University of British Columbia

Abstract

This paper presents a semantics-based compilation model for an aspect-oriented programming language based on its operational semantics. Using partial evaluation, the model can explain several issues in compilation processes, including how to find places in program text to insert aspect code and how to remove unnecessary run-time checks. It also illustrates optimization of calling-context sensitive pointcuts (cflow), implemented in real compilers.

Keywords

Aspect-Oriented Programming, Aspect SandBox, dynamic join point model, partial evaluation, Futamura projection, compile-time weaving

This paper will appear in *Proceedings of 12th International Conference on Compiler Construction (CC 2003)*, Lecture Notes in Computer Science, Springer-Verlag, April 2003.