

Baskiyar, S.**Efficient execution of pure object-oriented programs by follow-up compilation.** (English)[Zbl 1239.68022](#)[Computing 69, No. 4, 273-289 \(2002\).](#)

Summary: Dynamic type checking and method binding slows pure object-oriented programs such as those written in Smalltalk. Dynamic method lookup is needed to support method overriding and type-method conformance. We have developed a platform independent technique that implements method overriding efficiently for pure object-oriented languages. The technique involves binding non-overridden method calls at compile time. The overridden method calls are also resolved (and bound) statically using simple inferences. Overridden method bindings are corrected dynamically using an overridden method dictionary. Using simulations, we demonstrate that this technique offers 35%–70% reduction in the average method lookup delay over the Berkeley Smalltalk implementation. This technique has very low compile time overhead, memory overhead, does not need specialized hardware and allows shared code pages in concurrent programs. For programs which may be re-used, it is proper to have a follow-up compilation to generate efficient code after the program development is complete.

MSC:[68N20](#) Theory of compilers and interpreters[68N19](#) Other programming paradigms (object-oriented, sequential, concurrent, automatic, etc.)[68N15](#) Theory of programming languages[68N25](#) Theory of operating systems**Keywords:**[dynamic binding](#); [object-oriented](#); [overriding](#); [smalltalk](#); [type checking](#)**Software:**[Smalltalk](#)**Full Text:** [DOI](#)