

Formalizing the Java Memory Model for Multithreaded Program Correctness and Optimization

*Yue Yang, Ganesh Gopalakrishnan, and
Gary Lindstrom*

UUCS-02-011

School of Computing
University of Utah
Salt Lake City, UT 84112, USA

April 2, 2002

Abstract

Standardized language level support for threads is one of the most important features of Java. However, defining and understanding the Java Memory Model (JMM) has turned out to be a big challenge. Several models produced to date are not as easily comparable as first thought. Given the growing interest in multi-threaded Java programming, it is essential to have a sound framework that would allow formal specification and reasoning about the JMM.

This paper presents the Uniform Memory Model (UMM), a formal memory model specification framework. With a flexible architecture, it can be easily configured to capture different shared memory semantics including both architectural and language level memory models. Based on guarded commands, UMM is integrated with a model checking utility, providing strong built-in support for formal verification and program analysis. A formal specification of the JMM following the semantics proposed by Manson and Pugh is presented in UMM. Systematic analysis has revealed interesting properties of the proposed semantics. In addition, several mistakes from the original specification have been uncovered.