KaffeOS: Isolation, Resource Management and Sharing in Java as presented by Godmar Back

Christopher E. Davis - chris2d@cs.unm.edu
University of New Mexico
Computer Science Colloquia Spring 2004

March 29, 2004

Abstract

Java is widely used for applications, servlets, mobile agents, etc. The main motivation of KaffeOS is to provide a robust environment for these applications. While doing this we would like to make efficient use of resources, increase scalability and run on small hand-held systems.

1 Introduction

Our current options for JVM execution structure are severely limited - run one JVM with all applications in that JVM or run multiple JVM (one per application). The single JVM allows memory sharing, but lacks resource management. The multiple JVM model allows you to more finely manage resources, but you loose shared memory.

Mr. Back suggests that we create a JVM OS that runs in a single JVM but provides the services that motivate this problem. He suggests that KaffeOS is different from the JavaOS in that JavaOS was billed as a full featured OS written in Java, whereas KaffeOS is simply a simplistic operating system for the JVM.

2 Discussion

KaffeOS seemed to perform nicely under all of the benchmarks that Mr. Back presented. I particularly appreciated it's performance under load with a mali-

cious memory or CPU hog. I would like to see KaffeOS benchmarks comparing the baseline JVM performance versus the enhanced version performance.
The memory management system seemed quite straight forward. I would appreciate hearing why IBM and other leading JVM manufacturers have not included this feature yet. One major flaw I felt existed was the necessary trust of kernel code. Mr. Back glossed over the difficulty of code verification, when in fact I feel it is a hard and important problem to incorporate into his system. Mr. Back seemed to make code verification sound simple.
I would also like to know why KaffeOS leaks memory. Mr. Back stated that the leakage was minimal (bytes) - but never stated WHY it leaked. It was unclear if this was due to the process of memory reclamation or if this was a leak introduced by the base JVM code he was using.

3 Conclusion

KaffeOS sounds like a very interesting innovation. Systems and Java are not my area of study, and as such I am sure much of the nuance of this lecture was lost on me. It is obvious that resource management is important to the future of JVMs.

4 Comments

Mr. Back handled Professor William's question well, and his answer was nicely presented. His flow was pleasant and his presentation seemed better polished than his predecessors. Again though, I have to say that his area of study is of no interest to me. I hope to see some more varied applicants.