An Analysis of Operating System Behavior on a Simultaneous Multithreaded Architecture

Joshua A. Redstone, Susan J. Eggers and Henry M. Levy

University of Washington

{redstone, eggers, levy}@cs.washington.edu

Abstract

This paper presents the first analysis of operating system execution on a simultaneous multithreaded (SMT) processor. While SMT has been studied extensively over the past 6 years, previous research has focused entirely on user-mode execution. However, many of the applications most amenable to multithreading technologies spend a significant fraction of their time in kernel code. A full understanding of the behavior of such workloads therefore requires execution and measurement of the operating system, as well as the application itself.

To carry out this study, we (1) modified the Digital Unix 4.0d operating system to run on an SMT CPU, and (2) integrated our SMT Alpha instruction set simulator into the SimOS simulator to provide an execution environment. For an OS-intensive workload, we ran the multithreaded Apache Web server on an 8-context SMT. We compared Apache's user- and kernel-mode behavior to a standard multiprogrammed SPECInt workload, and compared the SMT processor to an out-of-order superscalar running both workloads. Overall, our results demonstrate the micro-architectural impact of an OS-intensive workload on an SMT processor and provide insight into the OS demands of the Apache Web server. The synergy between the SMT processor and Web and OS software produced a greater throughput gain over superscalar execution than seen on any previously examined workloads, including commercial databases and explicitly parallel programs.