Performance Characterization of the Pentium II Using Web Workloads

Shuchi Chawla, Rose Hoberman, Bianca Schroeder
{shuchi, roseh, bianca}@cs.cmu.edu

October 20, 2001

Project Web Page

`http://www.cs.cmu.edu/~bianca/architecture/project.html`

Project Description

Recent work [1], [2], [3] shows that the characteristics of commercial applications can be significantly different from the numerical and scientific applications that are typically used in benchmarks (e.g. the SPEC suite). All of this work, however, has focused on commercial database systems. We propose to do a similar study for another important non-scientific application: web servers.

Our goal is to measure the performance of modern architectures when running a web server. In particular, we plan to set up a web server on a Pentium II hardware platform running Linux. We will then generate web requests and measure the performance of hardware in terms of stall times, cache misses and branch mispredictions.

Based on our measurements we hope to gain some insights on how web servers could be designed to make better use of today's hardware or how the hardware could be better optimized for web servers.

One interesting issue is how different workloads (e.g. static versus dynamic requests) affect the architectural performance. This might give some insights on how to design/configure a machine depending on what workload will run on it. Similarly, there are two different types of web servers: those that use multiple threads or processes to handle requests and those that are based on the "select()" call. It would be interesting to see how these web server design choices affect the architectural performance.

However, due to the limited time we will probably have to focus on one type of workload and one web server in this class project.

Resources

We have access to six Pentium II machines running Linux 2.2.14 that are directly connected through a switch. One of these machines has the Apache web server and the Flash web server installed. The other 5 machines are set up as workload generators for static web workload.

Since a web server is interactive we cannot use simulation to obtain meaningful performance numbers. Instead we will have to use hardware counters. After talking to other people who have used hardware counters in their work we decided to use the perfctr tool in PAPI. This tool works with Linux 2.2.14 and is freely available at `http://icl.cs.utk.edu/projects/papi/`.

Plan of Attack

Since the application software is already installed and running, the "critical path" in our schedule will be setting up the hardware counter measurements.

Week 1: Set up the perfctr software with our kernel (Rose and Bianca)
Understand the web server source code (Shuchi)

Week 2: Learn the perfctr API (Rose and Bianca)

Week 3: Learn the meanings of the processor counters (Rose and Bianca)
Instrument the web server source code (Shuchi)

Week 4: Run trial experiments and debug.

Week 5: Run final experiments.

Week 6: Evaluate results and writeup.

By the project milestone we plan to have the the counters working, the web server fully instrumented, and be ready to run our experiments.

Literature

L. A. Barroso, K. Gharachorloo, and E. Bugnion. *Memory System Characterization of Commercial Workloads.* Proceedings of the 25th Annual International Symposium on Computer Architecture, June 1998.

K. Keeton, D. A. Patterson, Y. Q. He, R. C. Raphael, and W. E. Baker. *Performance Characterization of a Quad Pentium Pro SMP Using OLTP Workloads.* Proceedings of the 25th International Symposium on Computer Architecture, June 1998.

A. Ailamaki, D. J. DeWitt, M. D. Hill, and D. A. Wood. *DBMSs on a Modern Processor: Where Does Time Go?.* Proceedings of the 25th International Conference on Very Large Data Bases (VLDB), Edinburgh, UK, September 1999 .