

Project Proposal

Distributive Memory benchmarking with OpenMosix Clustering

Prepared for: Professor Kaugars, CS 454, Spring 2006

Prepared by: Lucas Holt & Swaroop Atre

January 23, 2006

Summary

Objective

To create a linux cluster using OpenMosix and shared memory patches to the Linux 2.4 kernel and benchmark the performance characteristics with distributed memory enabled and disabled. Explore the usefulness of distributed memory in network environments.

Goals

Setup a linux cluster with the latest stable OpenMosix patches running on the Linux 2.4 kernel. Add shared memory support and run several memory benchmarks including steam, bonnie, and bonnie++ if they are compatible with OpenMosix. Also use standard linux facilities such as hdparm. In addition, create a few simulated loads to test the performance benefits of the cluster versus a standard SMP environment (2 cpus with and without hyperthreading).

An introduction to OpenMosix

OpenMosix is a project based on work done in Israel to allow linux processes to be distributed through a cluster environment, but in a manner that simulates SMP process execution. It uses kernel patches and userland utilities to control the cluster and provide the scheduling and monitoring of nodes. The focus is exclusively on giving processes 100% CPU utilization on a host. Without patches, the system can not run shared memory applications, networked applications like the Apache HTTPD server, or multithreaded applications. Ideal workloads are those which use a series of forked processors or when a series of processes needs to be executed like encoding mp3's in a batch.

The OpenMosix website is located at <http://openmosix.sourceforge.net>. A brief review of the shared memory extensions for the Linux kernel and OpenMosix can be found at <http://www.unixreview.com/documents/s=8989/ur0404/>.

Memory benchmarking may be done with the following:

<http://www.cs.virginia.edu/stream/>

<http://www.garloff.de/kurt/linux/bonnie/>

<http://www.coker.com/au/bonnie++/>