

Paul Knight

An Introduction

My name is Paul Knight, and I'm a fourth year undergraduate student at the University of California, San Diego, majoring in Computer Science. I took Professor Steven Swanson's 240A Computer Architecture course last quarter and wanted to continue taking graduate courses, in part because I want a more thorough and academic approach to what I've learned in undergraduate courses, and in part because I'd like something from my major to balance the less challenging general education courses I'm taking.

I'd like to take graduate school, and though I've applied for Fall 2008 admittance, I'm still unsure when I want to pursue a Masters after graduation. In industry, I work part time for a Web 2.0 startup Eventful.com, currently a medium sized site that uses a network architecture approach to parallelism and load balancing. We've used distributed computing in several cases to process large amounts of data, but I'd like to explore a more tightly coupled architecture through this class. I've also had an internship with Apple, though the focus there was somewhat orthogonal to these ideas.

Though I have no specific project in mind, I think I would like something with a strong visual component. To be honest, I'm concerned about having trouble finding applications that are both interesting and obviously suitable for parallel programming; many of the ideas that seem interesting strike me as being better suited for a distributed model.

I'd like to work with a partner, and I consider myself a good teammate, technically and personally. I might work with one of the friends I'm taking the course with, but I would also like to partner with one of the graduate students. I'd be very happy to help others with the technical aspects of their collaboration in areas such as source control and team programming strategies, but I'm not sure how well I could help with the abstract.

A Parallel Application

The National Cancer Institute retooled a system for calculating correlation between a database of genome sequences using Interactive Supercomputing's Star-P system. The Genomic Correlation project gives NCI faculty and staff profiling data for exploring the relationship of genes, important for studying genetic factors in cancer research.

Star-P is a MATLAB environment for parallel computers. In some cases, MATLAB code can be parallelized automatically; for example, in the case of many matrix and vector operations. In other cases, additional functions provide parallel implementations for the standard library's serial counterparts. The core of the Genomic Correlation system can run unmodified on both standard desktops and in parallel on parallel computers.

The dataset size of the Genomic Correlation project, about 40,000 probe entries which could expand into a data matrix 100,000 elements by 100,000 elements, was running into memory issues

on desktop machines. The parallel approach was chosen to solve the memory problem, not necessarily a computational bottleneck.

The hardware itself is an SGI Altix server with a large amount of memory. Matrix operations are vectorized automatically, and the processing is tightly coupled. I/O is split in a more distributed fashion. On a four dual-core Opteron processor machine with 32GB of memory, NCI researchers saw a 200X speed increase over a Pentium-D single-processor single-core desktop with 2GB.

This sort of automatic parallelization, using the larger resources and programming language semantics to provide a better performing implementation with minimal programmer effort, is particularly interesting to me. Parallel and distributed programming is traditionally much more difficult than single threaded programming, and offloading the difficult semantics to a framework or language is very compelling.