Final Report for the Castle Project
1 May 1994 - 30 April 1998

www.cs.berkeley.edu/projects/parallel/castle/

Tom Anderson, David Culler, James Demmel, Jerry Feldman, Susan Graham, Paul Hilfinger, Katherine Yelick

Computer Science Division
University of California, Berkeley

16 February 2000

The goal of the Castle project was to provide a parallel programming environment that enables the construction of high performance applications that run portably across many platforms. Our approach was to design and implement a multilayered architecture, with higher levels building on lower ones to ensure portability, but with care taken not to introduce abstractions that sacrifice performance. Our layers include the following (listed from bottom to top, with highlights):

1. Interprocessor communications. This includes work on Active Messages and MPI. The work on Active Messages was part of the basis of the new industry communication standard VIA, adopted by Intel and other computer companies.

2. System support. This includes threads, scheduling, load balancing, and I/O support. This work led to the world's record disk-to-disk sort in 1997 (www.cs.berkeley.edu/NowSort/index.html).

3. Libraries. This includes work on ScaLAPACK, SuperLU, Multipol, PHIPAC, and studies of the impact of heterogeneous computing. ScaLAPACK has been widely adopted by many computer companies as part of their standard parallel library, and is part of the ASCI Red standard system build.

4. Languages. This includes Split-C, PSather and Titanium.

5. Applications. This includes a large number of studies, including climate modeling, circuit simulation, phylogeny tree computation, cell simulation, semiconductor modeling, computational fluid dynamics, connected components, sorting, connectionist networks, computing Gröbner bases, and web searching.

The symmetric eigensolver in ScaLAPACK was incorporated in a material science code at SNL, which was runner up in the 1998 Gordon Bell Competition at Supercomputing.

The distributed memory version of the SuperLU sparse direct linear system solver was used to solve an open problem in quantum chemistry at NERSC, resulting in the cover of the 24 Dec 1999 issue of Science.
DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, make any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.
DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.
The web searching application, initially implemented on the NOW using Active Messages, was made into a commercial product called Inktomi which was worth over $1B after going public in 1998.

In addition, we made the following contributions:

1. Performance Modeling and Tuning. In addition to performance modeling in the context of the above applications, various studies were performed to identify bottlenecks on different kinds of parallel architectures.

2. Education. Two courses were developed. “Applications of Parallel Computing” has extensive on-line notes and is taught regularly at Berkeley, with the notes used at several other universities and for short courses. “Parallel Computer Architecture” is also taught at Berkeley, and resulted in a published textbook.

3. Collaborations. We collaborate extensively with NERSC and other groups at Lawrence Berkeley National Lab on various issues related to high performance computing, including linear algebra libraries, and language support for computational fluid dynamics. Some of our software (ScaLAPACK and SuperLU) is also appearing in the ACTS Toolkit, funded by DOE 2000. This software as well as the Titanium language is part of the NSF NPACI Parallel Tools and Environments effort.

The NOW was part of the NSF NPACI SuperComputer Center, one of the supercomputers NSF users could elect to use.

We also received a $6M grant from Intel Corporation, as well as other large grants from SUN Microsystems, Microsoft, and the National Science Foundation to build a parallel computing infrastructure for 18 departments across the Berkeley campus. This system, called Millennium (www.millennium.berkeley.edu) draws heavily on our experience in Castle.

We divide our report into several sections which lists the publications for the above topics.

1 Interprocessor Communications


2. S. Lumetta, A. Mainwaring, D. Culler, "Multi-Protocol Active Messages on a Cluster of SMP's." Supercomputing '97, San Jose, CA, November 1997


2 System Support


2. S. Lumetta, "Mantis: A Debugger for the Split-C Language." University of California at Berkeley, Computer Science Division, 1994 (MS Thesis)

3. S. Luna, "Implementing an Efficient Portable Global Memory Layer on Distributed Memory Multiprocessors." University of California at Berkeley Computer Science Division, 1994 MS Thesis


5. D. Culler, A. Arpaci-Dusseau, R. Arpaci-Dusseau, B. Chun, S. Lumetta, A. Mainwaring, R. Martin, C. Yoshikawa, F. Wong, "Parallel Computing on the Berkeley NOW." JSPP '97 (9th Joint Symposium on Parallel Processing)


3 Libraries


4 Languages


5 Applications


2. A. Arpaci-Dousseau, R. Arpaci-Dousseau, Culler, Hellerstein, Patterson, "Searching for the Sorting Record: Experiences in Tuning the NOW-Sort." The 1998 Symposium on Parallel and Distributed Tools (SPDT '98), Welches, Oregon, August 3-4, 1998.

3. A. Arpaci-Dousseau, R. Arpaci-Dousseau, Culler, Hellerstein, Patterson, "High-Performance Sorting on Networks of Workstations." SIGMOD '97, Tuscon, AZ, May 1997

6


6 Performance Modeling and Tuning


7 Education

We developed extensive course material in conjunction with this grant. First, we developed CS 267, "Applications of Parallel Computing." This course is taught annually at Berkeley, usually with 50% enrollment from within CS and 50% from other engineering and physical science departments (and the occasional economist). It has been taught by PIs Culler, Demmel, and Yelick. Demmel developed an extensive set of on-line notes concentrating on algorithms (www.cs.berkeley.edu/~dummell/cs267), and later augmented by Culler (now.CS.Berkeley.EDU/cs267/) and Yelick (www.cs.berkeley.edu/~dmartin/cs267/). In addition to being taught at Berkeley, it was taught in short course form at a NSF-CBMS Lecture on Parallel Numerical Linear Algebra (June 1995, Demmel as main lecturer and organizer), at a NATO Advanced Summer Institute in July 1996, at ETH Zurich in 1996, and at an NPACI Training Institute in September 1997.

Second, we wrote and published a textbook for CS 258, "Parallel Computer Architecture". Culler and J.P. Singh are principal authors, with Anoop Gupta of Stanford and Microsoft. See www.cs.berkeley.edu/~culler/book.alpha/index.html. This book explains the forces behind the convergence of shared-memory, message-passing, data parallel, and data-driven computing architectures. It then examines the design issues that are critical to all parallel architecture across the full range of modern design, covering data access, communication performance, coordination of cooperative work, and correct implementation of useful semantics. It not only describes the hardware and software techniques for addressing each of these issues but also explores how these techniques interact in the same system.