Symunix-2
(Highly Parallel Operating System)

NYU Ultracomputer Project
Ultracomputer Research Laboratory
New York University

Personnel

Principal investigator	Allan Gottlieb	gottlieb@nyu.edu
Researchers	Jan Edler	edler@nyu.edu
	Eric Freudenthal	freudent@nyu.edu
	David Wood	wood@nyu.edu
	Jim Lipkis	lipkis@nyu.edu
Student	Miriam Ayala	ayala@lab.ultra.nyu.edu

Contact

Jan Edler
NYU Ultracomputer Research Laboratory
715 Broadway, 10th floor
New York, NY 10003

edler@nyu.edu
(212) 998-3353

Project Description

Our operating systems work is currently focused on design and implementation of a new UNIX system for highly-parallel shared-memory MIMD computers. Our original effort, known simply as *Symunix*, has many highly parallel attributes and has been running reliably since 1985. The new system is known as *Symunix-2*; the following paragraphs briefly describe some of its most interesting features.

Avoidance of serial bottlenecks due to critical sections is important because their cost increases with the size of the machine. Our approach relies on Fetch&Phi operations, chiefly Fetch&Add, which are amenable to combining in some architectures. Many critical section free algorithms and data structures have been already been developed and tested in the earlier Symunix effort, including a variety of general-purpose list and synchronization types. For Symunix-2, we are expanding the use of those algorithms, and also designing new ones to solve particular operating system problems.

Efficient support for applications with highly-volatile fine-grained parallelism is key to broadening the applicability of large scale parallel machines. The key technique used for this purpose in Symunix-2 is user-mode implementation of threads, which are not known to the OS kernel. Efficiency for user-mode thread implementations is achieved by supporting asynchronous system calls and asynchronous page faults, and providing fast and reliable mechanisms for user-level control of preemption.

We are also exploring a new virtual memory system with powerful and flexible support for memory sharing. All user-accessible virtual memory is implemented by mapping of files; there is no “anonymous” memory. This simple mechanism simultaneously provides both backing store and naming, allowing for the support of a wide variety of parallel programming models.

Report Availability

Most of the references cited below are available in compressed postscript form via anonymous FTP from cs.nyu.edu, in the directories under pub/local/ultra/UCN.

References

[1] Lawrence Rudolph, *Software Structures for Ultraparallel Computing*, Ph.D. thesis, Courant Institute, New York University, February, 1982.

[2] Allan Gottlieb, B. D. Lubachevsky, and Larry Rudolph, *Basic Techniques for the Efficient Coordination of Very Large Numbers of Cooperating Sequential Processors*, ACM Transactions on Programming Languages and Systems, April 1983, p164-189.

[3] Allan Gottlieb, *An Overview of the NYU Ultracomputer Project*, in "Experimental Parallel Computing Architectures", Jack J. Dongarra (Ed.) North Holland, 1987, p25-95. A revised version is also available as Ultracomputer Note #100, New York University.

[4] James Wilson, *Operating System Data Structures for Shared-Memory MIMD Machines with Fetch-and-Add*, Ph.D. thesis, Courant Institute, New York University, June, 1988.

[5] Jan Edler, Jim Lipkis, and Edith Schonberg, *Memory Management in Symunix II: A Design for Large-Scale Shared Memory Multiprocessors*, Proc. USENIX Workshop on UNIX and Supercomputers, Sept. 1988. Also available as Ultracomputer Note #135, New York University.

[6] Jan Edler, Jim Lipkis, and Edith Schonberg, *Process Management for Highly Parallel UNIX Systems*, Proc. USENIX Workshop on UNIX and Supercomputers, Sept. 1988. Also available as Ultracomputer Note #136, New York University.