NUMAtic Project and the DUnX OS

Department of Computer Science
Duke University

Personnel

Principal investigators	Carla Ellis	carla@cs.duke.edu
	Mark Holliday	holliday@cs.duke.edu
Other contributors	Rick LaRowe	rlarowe@encore.com
	David Kotz	dfk@cs.duke.edu
Students	Vick Khera	khera@cs.duke.edu
	Steve Owen	so@cs.duke.edu
	Chris Connelly	jcc@cs.duke.edu

Contact

Carla Ellis
Department of Computer Science
Duke University
Durham NC 27243

(919) 660-6523
carla@cs.duke.edu

Project Description

Non-uniformity of memory access is an almost inevitable feature of the memory architecture in any shared memory multiprocessor design that can scale to large numbers of processors. The NUMAtic project is a program of experimental research exploring memory management on NUMA machines.

The scope of the project covers a full range of policies and mechanisms for controlling the location and movement of data including

1. The migration of code and data among the distinct memory modules that collectively comprise the main memory of a NUMA machine.
2. The replication (or caching) of copies of pages within main memory.
3. Interactions among issues of dynamic memory placement, processor allocation, and thread scheduling.
4. The organization and management of secondary memory based on parallel independent disks.

The DUnX (pronounced "ducks") Operating System on BBN Butterfly GP-1000 multiprocessor has been developed to study issues 1-4. DUnX has been in daily use at Duke's Computer Science Department since November 1989. The RAPID Transit file system testbed has been designed to investigate item 5.

Report Availability

Copies of Duke Technical Reports are available from

Kathy Redding
Department of Computer Science
Duke University
Durham, NC 27706
kathy@cs.duke.edu

There is a $5.00 charge for Ph.D. theses.

Selected reports are available through anonymous FTP from cs.duke.edu [128.109.140.1] in compressed PostScript format in the directory pub/NUMAtic.

References

[1] R. Adema, "DUSTy - a programming environment for distributed shared memory multiprocessors," Master's thesis, Duke University, May 1991.

[2] R. Adema and C. Ellis, "Memory allocation constructs to complement NUMA memory management." submitted, June 1991.

[3] C. Ellis and D. Kotz, "Prefetching in file systems for MIMD multiprocessors," in *Proceedings of the 1989 International Conference onParallel Processing*, pp. 306–314, August 1989.

[4] M. Holliday, "Page table management in local/remote architectures," in *ACM SIGARCH Int. Conf. on Supercomputing*, pp. 1–8, July 1988.

[5] M. Holliday, "Reference history, page size, and migration daemons in local/remote architectures," in *Proceedings, Architectural Support for Programming Languages and Operating Systems*, pp. 104–112, April 1989.

[6] M. Holliday, "Reference history, page size, and page scanners in distributed shared memory machines." submitted, June 1990.

[7] M. Holliday, "On the effectiveness of dynamic page placement," Tech. Rep. TR-1989-19, Duke University, September 1989.

[8] M. Holliday, "Techniques for cache and memory simulation using address reference traces," *Int. Journal of Computer Simulation*. to appear.

[9] M. Holliday and C. Ellis, "Accuracy of memory reference traces of parallel computations in trace-driven simulation," *IEEE Transactions on Parallel and Distributed Systems*. to appear, also available as Duke University TR CS-1990-8.

[10] M. Holliday and C. Ellis, "Scalable shared memory architectures," ch. An Example of Correct Global Trace Generation, Kluwer Academic Publishers, 1991.

[11] M. Holliday, "A program behavior model and its evaluation," Tech. Rep. CS-1990-9, Duke University, April 1990. submitted.

[12] M. Holliday, "Trace-driven simulation of distributed shared memory environments," in *ISCA 90 Workshop on Scalable Shared Memory Architectures*, May 1990.

[13] D. Kotz, *Prefetching and Buffering Techniques in File Systems for MIMD Multiprocessors*. PhD thesis, Duke University, May 1991.

[14] D. Kotz and C. Ellis, "Practical prefetching techniques for parallel file systems." submitted, May 1991.

[15] D. Kotz and C. Ellis, "Caching and writeback policies in parallel file systems." submitted, June 1991.

[16] R. LaRowe, *Page Placement for NUMA Shared Memory Multiprocessors.* PhD thesis, Duke University, May 1991.

[17] R. LaRowe, "Addressing the NUMA problem at the operating systems level," in *CMG'91 International Conference*, December 1991. to appear.

[18] R. LaRowe and C. Ellis, "Experimental comparison of memory management policies for NUMA multiprocessors," Tech. Rep. CS-1990-10, Duke University, April 1990. submitted.

[19] R. LaRowe and C. Ellis, "Page placement policies for NUMA multiprocessors," *Journal of Parallel and Distributed Computing*, vol. 11, pp. 112–129, February 1991.

[20] R. LaRowe, C. Ellis, and L. Kaplan, "The robustness of NUMA memory management," in *13th Symp. on Operating Systems Principles (SOSP)*, October 1991. also available as TR CS-1991-05.

[21] R. LaRowe, J. Wilkes, and C. Ellis, "Exploiting operating system support for dynamic page placement on a NUMA shared memory multiprocessor," in *Symposium on Principles and Practice of Parallel Programming (PPoPP)*, pp. 122–132, April 1991.

[22] R. LaRowe and C. Ellis, "OS experimentation and a user community coexist under the DUnX kernel," in *Proceedings of International Conference on Parallel Processing (ICPP)*, August 1991. to appear.

[23] R. LaRowe and C. Ellis, "Dynamic page placement in a NUMA multiprocessor virtual memory system," Tech. Rep. CS-1989-21, Duke University, October 1989.

[24] R. LaRowe, M. Holliday, and C. Ellis, "An analysis of dynamic page placement on a NUMA multiprocessor." May 1991.