

Statement of Research

Martin Schulz

`schulzm@in.tum.de`

Parallel and distributed architectures are becoming increasingly commonplace, with parallelism being exposed at various levels ranging from Instruction Level Parallelism (ILP) and Chip MultiProcessors (CMP) to Grid infrastructures. The questions posed at all these levels, however, are often the same and require similar approaches. The ultimate goal is to fully exploit the given capabilities. To this end, both architectural enhancements and optimized software techniques, as well as the potential synergy between them should be investigated. This leads to integrative approaches which are capable of exploiting the (potentially even multi-level) parallelism in a manner that is adjusted to the target system and hence efficient, as well as transparent and architecture independent.

My research, therefore, focuses on this question in a general way and applies both hardware and software concepts already established in one particular level of parallelism to others, as well as across a hierarchy of parallelism. Of special interest thereby are the questions of a synergetic cooperation between hardware and software components and of an adaptability to varying degrees and levels of parallelism, potentially even dynamically at runtime. A good example for the latter is the programming of SMP clusters, for which currently no simple and unified programming approach exists which exploits both levels of parallelism in a consistent and yet optimized manner. Additionally, I want to focus on the issue of consistent global data representations with relaxed consistency models, as my current research has shown that this provides significant performance potentials for the future.

Besides this core research, I believe that any concept should be evaluated using real-world or industrial applications in addition to traditional academic benchmarks. Only then can realistic assessments of the concepts developed be made and hence ensure a long term impact.

Early Work

During the course of my work, I had the chance to be involved in many projects dealing with different types of parallelism. Early work for my master's degree focused on an application study for a parallel programming language. At the University of Illinois at Urbana-Champaign under the supervision of Dr. Andrew Chien (now UCSD), I developed a parallel volume rendering code based on iso-surface extraction [8] for the Illinois Concert C++ environment [2], a semi-explicit, object-oriented parallel programming approach based on C++ as the base language. This work focused on the investigation of overheads associated with this kind of high-level programming of parallel systems and how to overcome them. Especially issues like data placement and locality as well as synchronization and locking were among the most important issues explored.

Also during my time at U of I, and together with Dr. A. Chien and Dr. R. Gupta, I worked on the Morph project [1]. It focused on a reconfigurable architecture for massively parallel processing (it was actually part of the Petaflop initiative started in 1997). The main idea was to introduce small amounts of reconfigurable logic along the memory hierarchy in order to achieve an adaptation of the architecture to the memory behavior of the target application. The adaptation can go as far as reserving private storage areas in caches, allowing application specific data packing in caches, and pointer chasing inside the memory modules. I was involved in this project at a very early stage, together with other graduate students, and we focused on establishing a comprehensive simulation environment for this novel architecture using MINT [17]. Using this, we then showed the potential benefit of this approach using small benchmarks, the most interesting ones dealing with sparse matrices [18]. Unfortunately, the project has not progressed further since then, due more to personnel changes than to lack of research potential, but I consider it to be promising for the future.

SMiLE: Shared Memory in a LAN-like Environment

During the last five years, I have been actively involved in the SMiLE (Shared Memory in a LAN-like Environment) project¹ investigating the potentials and challenges of clustering and parallel processing using NUMA architectures, mainly based on SCI (the Scalable Coherent Interface) [4, 5]. In this project, my responsibilities significantly exceeded the pure research and implementation work of my own Ph.D. project; it also included other aspects within SMiLE as well as many administrative, planning, and research management tasks, including grant writing.

Within the SMiLE project, we are developing both SCI hardware and software. The hardware aspect includes a reconfigurable PCI-SCI adapter for research in communication architectures and networking and a hardware monitor capable of observing all memory accesses in a manner which allows relating the observed transactions back to addresses and data structures. On the software side, we have developed a comprehensive and integrated environment which in its implementation is closely oriented on the hardware aspects and capabilities of the underlying architecture while still providing a high-level, powerful abstraction for programmers.

The work on this software infrastructure has been supported by several grants from the European Union, in which I was directly involved (both from the administrative and the research side). The most important were SISCO [3] and NEPHEW [6], two large-scale projects on SCI-based cluster computing with partners and collaborators from both academia and industry across Europe.

In order to stay as versatile as possible, we designed SMiLE to include equal support for messaging and shared memory. For the shared memory, my main area of research in the last few years and also the topic of my dissertation [10], I designed a framework that maintains the choice of concrete programming models for the programmer while still guaranteeing the highest possible performance by directly exploiting the remote memory facilities present in NUMA architectures. The result is HAMSTER (Hybrid-DSM based Adaptive and Modular Shared memory archiTectuRe)², a framework for shared memory programming on NUMA characterized clusters [7, 10]. It enables the low-complex implementation of almost arbitrary shared memory programming models on top of a single efficient DSM core using a novel Hybrid-DSM technique [11]. In addition, the framework proposes a solution for solving the issue of missing cache coherency in these architectures while keeping the programmability for the end user. The result is a consistent, yet optimized and efficient infrastructure enabling a global data representation. This topic is currently being extended to a more general architectural model and tries to explore the potentials and challenges connected with dropping the often complex cache coherency protocols and directly leveraging on non-coherent memory while keeping the impact for the programmer at a minimum.

While this work on the software infrastructure itself has been more or less completed and is currently the topic of further evaluation, this work has produced several important offspring which form my current area of research. Most important among them are the questions of efficient and application adaptive parallel I/O for cluster environments [9], a tighter integration of memory and consistency management of relaxed memory schemes [11] in NCC-NUMA environments, and easy-to-use performance tools for shared memory environments. Significant work has been done within the latter topic on a portable and versatile monitoring infrastructure and a tool set for locality optimizations in NUMA environments [15, 14].

For the future, our group is extending the results achieved in two new directions: cache and memory performance monitoring in SMPs and CMPs using the SMiLE monitoring approach and data and resource management in Grid-like ubiquitous computing environments using DSM-like techniques to achieve a global view on the data. Both directions are important issues for future research, and I would like to stay involved in a cooperative role.

Cross-Disciplinary Evaluation

An important aspect of any architecture or systems project is the evaluation of its concepts using large-scale real-world applications. This must be done to drive the evaluation beyond the usual small and simple benchmarks which, despite their undoubted academic merit of finding particular performance problems and aiding in detailed performance analysis, do not allow an assessment of the investigated concepts in realistic scenarios.

¹More information also at <http://smile.in.tum.de/>

²More information also at <http://hamster.in.tum.de/>

An essential prerequisite to enable this evaluation is a tight cooperation with partners in various fields with computationally demanding applications. This has been done extensively within the SMiLE project. Specifically, I worked in cooperation with the department of nuclear medicine from TUM's university clinic on applications from the area of PET imaging (Positron Emission Tomography) [13, 12] and with ABB Corporate Research, Heidelberg, on a code to compute electric and electromagnetic fields for high voltage transformers and switchgear [16]. In addition, we recently started evaluating codes for both phylogenetic analyses (in cooperation with the chair for microbiology at TUM) and for image analysis (in cooperation with the Max Planck Institute for Neuropsychological Studies in Leipzig). All of these cooperations are proving mutually beneficial and we find them quite insightful in terms of evaluating and optimizing the SMiLE infrastructure.

Future Directions

In the future, I would like to maintain this diverse and adaptive approach to parallel architectures in my research. My goal is to continue designing software frameworks which are both hardware and architecture oriented in order to exploit the capabilities of the underlying system while concurrently providing a high-level and easy-to-use abstraction for the end user. These frameworks should therefore not only provide an abstraction focused on encapsulating parallel tasks, but instead contain a useful global data abstraction which allows a consistent view on data potentially stored in a relaxed consistency model. Target systems potentially range from threading in CMP or SMT systems to Grid or Grid-like infrastructures.

In addition, I would like to continue to work on real-world applications with a medical and biological background as realistic test scenarios for the systems under development. These two fields currently have a high demand for introducing new computational methods and therefore offer ample opportunities for cross-disciplinary cooperation.

References

- [1] A. Chien and R. Gupta. Morph: A system architecture for robust high performance using customization. In *Proceedings of Frontiers*, 1996.
- [2] A. Chien, U. Reddy, J. Plevyak, and J. Dolby. ICC++ — A C++ Dialect for High Performance Parallel Computing. Technical report, Department of Computer Science, University of Illinois at Urbana–Champaign, 1996.
- [3] M. Eberl, H. Hellwagner, B. Herland, and M. Schulz. SISCO — Implementing a Standard Software Infrastructure on an SCI Cluster. In W. Rehm, editor, *Tagungsband zum 1. Workshop Cluster Computing*, number CSR-97-05 in Chemnitzer Informatik–Berichte, pages 49–61, November 1997.
- [4] H. Hellwagner and A. Reinefeld, editors. *SCI: Scalable Coherent Interface. Architecture and Software for High-Performance Compute Clusters*, volume 1734 of *LNCS State-of-the-Art Survey*. Springer Verlag, October 1999. ISBN 3-540-66696-6.
- [5] IEEE Computer Society. *IEEE Std 1596–1992: IEEE Standard for Scalable Coherent Interface*. The Institute of Electrical and Electronics Engineers, Inc., 345 East 47th Street, New York, NY 10017, USA, August 1993.
- [6] W. Karl, M. Schulz, M. Völk, and S. Ziegler. NEPHEW: Applying a Toolset for the Efficient Deployment of a Medical Image Application on SCI-based clusters. In A. Bode, T. Ludwig, W. Karl, and R. Wismüller, editors, *Euro-Par 2000 — Parallel Processing*, volume 1900 of *Lecture Notes of Computer Science (LNCS)*, pages 851–860. Springer Verlag, Berlin, September 2000.
- [7] Martin Schulz. Efficient deployment of shared memory models on clusters of PCs using the SMiLEing HAM-STER approach. In A. Goscinski, H. Ip, W. Jia, and W. Zhou, editors, *Proceedings of the 4th International Conference on Algorithms and Architectures for Parallel Processing (ICA3PP)*, pages 2–14. World Scientific Publishing, December 2000.

- [8] M. Schulz. Application study for the Illinois Concert C++, A Parallel Volume Renderer. Master's thesis, University of Illinois at Urbana–Champaign, January 1997.
- [9] M. Schulz. DIOM: Parallel I/O for Data Intensive Applications on Commodity Clusters. In *Parallel and Distributed Computing and Systems (PDCS)*. ACTA Press, August 2001.
- [10] M. Schulz. *Shared Memory Programming on NUMA-based Clusters using a General and Open Hybrid Hardware/Software Approach*. PhD thesis, Technische Universität München, July 2001.
- [11] M. Schulz and W. Karl. Hybrid-DSM: An Efficient Alternative to Pure Software DSM Systems on NUMA Architectures. In L. Iftode and P. Keleher, editors, *Proceedings of the Second International Workshop on Software Distributed Shared Memory*, May 2000.
- [12] M. Schulz, M. Völk, W. Karl, and S. Ziegler. Effiziente iterative PET-Bild Rekonstruktion auf einem Cluster von PCs. *Journal of Radiationoncology. Biology. Physics – Abstraktband des gemeinsamen Jahreskongresses der DEGRO, ÖGRO, DGMP*, 176(1), October 2000.
- [13] M. Schulz, Martin Völk, Wolfgang Karl, Frank Munz, and Sibylle Ziegler. Running a spectral analysis code on top of SCI shared memory using the TreadMarks API. In Geir Horn and Wolfgang Karl, editors, *Proceedings of SCI-Europe '99, The 2nd international conference on SCI-based technology and research*, pages 35–43. SINTEF Electronics and Cybernetics, September 1999. ISBN: 82-14-00014-9, Also available at <http://www.bode.in.tum.de/events/>.
- [14] J. Tao, W. Karl, and M. Schulz. Memory Access Behavior Analysis on NUMA-based Shared Memory Programs. *Scientific Computing, Special Issue on Performance Oriented Application Development for Distributed Architectures*. to appear.
- [15] Jie Tao, Wolfgang Karl, and Martin Schulz. Visualizing the Memory Access Behavior of Shared Memory Applications on NUMA Architectures. In *Proceedings of the International Conference on Computational Science (ICCS)*, 2001.
- [16] C. Trinitis, M. Schulz, M. Eberl, and W. Karl. SCI-based LINUX PC-Clusters as a Platform for Electromagnetic Field Calculations. In *Proceedings of the 6th Conference on Parallel Computing Technologies (PaCT)*, number 2127 in LNCS. Springer Verlag, Berlin, September 2001.
- [17] J. Veenstra and R. Fowler. *MINT Tutorial and User Manual*. University of Rochester, Computer Science Department, Rochester, New York 14627, Technical Report 452 edition, June 1994.
- [18] X. Zhang, A. Dasdan, M. Schulz, R. Gupta, and A. Chien. Architectural adaptation for application-specific locality optimizations. In *Proceedings of the International Conference on Computer Design ICCD*. IEEE, October 1997.