References

[1]

[2]

[3] David Abrahams, Beman Dawes, Jeff Garland, and Douglas Gregor. The future of boost. Panel at BoostCon 2007, Aspen, Colorado, May 2007.

[4] David Abrahams, Beman Dawes, Jeff Garland, and Douglas Gregor. Future of Boost 2008. Panel at BoostCon 2008, May 2008.

[5] David Abrahams and Jeremy Siek. Policy adaptors and the boost iterator adaptor library. In *Second Workshop on C++ Template Programming*, October 2001.

[6] David Abrahams, Jeremy Siek, and Thomas Witt. Iterator facade and adaptor. Technical Report N1476=03-0059, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, 2003.

[7] David Abrahams, Jeremy Siek, and Thomas Witt. New iterator concepts. Technical Report N1477=03-0060, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, 2003.

[8] T. Aittokallio, M. Gyllenberg, J. Järvi, O. Nevalainen, and O. Polo. Detection of high-frequency respiratory movements during sleep. *Computer Methods and Programs in Biomedicine*, 61(3):171–185, 2000.

[9] Andreas Albrecht, Peter Gottschling, and Uwe Naumann. Logarithmic simulated annealing for optimal derivative code. Technical report, Technical Report 372, University of Hertfordshire, UK, 2002.

[10] Andreas Albrecht, Peter Gottschling, and Uwe Naumann. Markowitz-type heuristics for computing jacobian matrices efficiently. In *Internat. Conference on Computational Science, St. Petersburg*, number 2658 in LNCS, 2003.

[11] Thomas Panas Dan Quinlan Zhendong Su Andreas Saebjoernsen, Jeremiah Willcock. Detecting code clones in binary executables. In *International Conference on Software Testing and Analysis (ISSTA)*, July 2009.

[12] B. Barrett, J. M. Squyres, A. Lumsdaine, R. L. Graham, and G. Bosilca. Analysis of the component architecture overhead in open mpi. In *Proceedings, 12th European PVM/MPI Users' Group Meeting*, Sorrento, Italy, September 2005.

[13] Brian Barrett, Jeff Squyres, and Andrew Lumsdaine. Integration of the LAM/MPI environment and the PBS scheduling system. In *Proceedings, 17th Annual International Symposium on High Performance Computing Systems and Applications*, Quebec, Canada, May 2003.

[14] Brian W. Barrett. Return of the MPI datatypes. *ClusterWorld Magazine, MPI Mechanic Column*, 2(6):34–36, June 2004.

[15] Brian W. Barrett. *One-Sided Communication for High Performance Computing Applications*. PhD thesis, Indiana University, 2009.

[16] Brian W. Barrett, Galen M. Shipman, and Andrew Lumsdaine. Analysis of implementation options for mpi-2 one-sided. In *Proceedings, Euro PVM/MPI*, Paris, France, October 2007.

[17] Brian W. Barrett, Jeffrey M. Squyres, and Andrew Lumsdaine. Implementation of open mpi on red storm. Technical Report LA-UR-05-8307, Los Alamos National Laboratory, Los Alamos, New Mexico, USA, October 2005.

[18] Jean-Philippe Bernardy, Patrik Jansson, Marcin Zalewski, Sibylle Schupp, and Andreas Priesnitz. A comparison of C++ concepts and Haskell type classes. In *Proc. ACM SIGPLAN Workshop on Generic Programming*, pages 48–37. ACM, 2008.

[19] L. Susan Blackford, James Demmel, Jack Dongarra, Iain Duff, Sven Hammarling, Michael Heroux, Linda Kaufman, Andrew Lumsdaine, Antone Petitet, Roldan Pozo, Karin Remington, and R. Clint Whaley. An updated set of basic linear algebra subprograms (blas). *ACM Transactions on Mathematical Software (TOMS)*, 28(2):135–151, 2002.

[20] Alex Breuer, Peter Gottschling, Douglas Gregor, and Andrew Lumsdaine. Effecting parallel graph eigensolvers through library composition. In *Performance Optimization for High-Level Languages and Libraries (POHLL)*, April 2006.

[21] Alex Breuer, Peter Gottschling, Douglas Gregor, and Andrew Lumsdaine. Effecting parallel graph eigensolvers through library composition. Presentation at Workshop on Performance Optimization for High-Level Languages and Libraries, Rhodes Island, Greece, April 2006.

[22] Jeffrey M. Squyres Andrew Lumsdaine Brian W. Barrett, Ron Brightwell. Implementation of open mpi on the cray xt3. In *Cray Users Group 2006*, Lugano, Switzerland, May 2006.

[23] R. H. Castain, T. S. Woodall, D. J. Daniel, J. M. Squyres, B. Barrett, and G .E. Fagg. The open run-time environment (openrte): A transparent multi-cluster environment for high-performance computing. In *Proceedings, 12th European PVM/MPI Users' Group Meeting*, Sorrento, Italy, September 2005.

[24] R. H. Castain, T. S. Woodall, D. J. Daniel, J. M. Squyres, B. Barrett, and G. E. Fagg. The open run-time environment (openrte): A transparent multicluster environment for high-performance computing. *Future Generation Computer Systems*, 24(2):153–157, 2008.

[25] Jiangtao Cheng, J. P. Morris, John Tran, Andrew Lumsdaine, N. J. Giordano, and L. J. Pyrak-Nolte. Single phase flow in a fracture: Micro-model experiments & network flow simulation. *International Journal of Rock Mechanics*, 41(4):687–693, 2004.

[26] Joseph A. Cottam. Declarative stream visualization, July 2008. IV'08 PhD colloquium presentation.

[27] Joseph A. Cottam, Joshua Hursey, and Andrew Lumsdaine. Seetest: Unit test visualization. In *IEEE Symposium on Information Visualization (InfoVis 2008)*, November 2008. Poster extending work presented at SoftVis 2008.

[28] Joseph A. Cottam, Joshua Hursey, and Andrew Lumsdaine. Seetest: Unit test visualization, May 2008. Poster at the Indiana University, School of Informatics poster session. Recieved honorable mention award.

[29] Joseph A. Cottam and Andrew Lumsdaine. Thisstar: Declarative visualization prototype. In *IEEE Symposium on Information Visualization*, 2007.

[30] Joseph A. Cottam and Andrew Lumsdaine. Tuple space mapper: Design, challenges and goals. Technical Report TR648, Indiana University, Bloomington, IN, June 2007.

[31] Joseph A. Cottam and Andrew Lumsdaine. Extended assortitivity and the structure in the open source development community. In *International Sunbelt Social Network Conference*. International Network for Social Network Analysis, January 2008.

[32] Joseph A. Cottam and Andrew Lumsdaine. Stencil: A conceptual model for representation and interaction. In *12th International Conference on Information Visualzation (IV'08)*, 2008.

[33] Joseph A. Cottam and Andrew Lumsdaine. Stencil: A declarative, generative system for visualizing dynamic data. In *IEEE Symposium on Information Visualization (InfoVis 2008)*, November 2008. Poster accompanying the presentation of Stencil in the PhD Colloquium.

[34] Joseph A. Cottam and Andrew Lumsdaine. Algebraic guide generation. In *13th International Conference on Information Visualzation (IV'09)*, 2009.

[35] Joseph A. Cottam and Andrew Lumsdaine. Algebraic guide generation. In *13th International Conference on Information Visualzation (IV'09)*, July 2009.

[36] Joseph A. Cottam and Andrew Lumsdaine. Bit by bit: Incremental data visualization. In *IEEE Symposium on Information Visualization*, 2009.

[37] Joseph A. Cottam and Andrew Lumsdaine. Guideme: Automatic generation of legends and guides, May 2009.

[38] Joseph A. Cottam, Andrew Lumsdaine, and Joshua Hursey. Representing unit test data for large scale software development. In *ACM Sypmposium on Software Visualization (SoftVis 2008)*, September 2008.

[39] Joseph A. Cottam, Ben Martin, Chris Mueller, and Andrew Lumsdaine. Reading the envelope: Understanding visual similarity matrices. In *IEEE Symposium on Visualization*, 2007.

[40] Debasis Dan, Chris Mueller, Kun Chen, and James A. Glazier. Solving the advection-diffusion equations in biological contexts using the cellular potts model. *Phys. Rev. E*, 72(041909), October 2005.

[41] Beman Dawes, Douglas Gregor, Jeremiah Willcock, and Andrew Lumsdaine. Concepts for the C++0x Standard Library: Chapter 17 – Introduction (revision 2). Technical Report N2755=08-0265, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, October 2008.

[42] Beman Dawes, Douglas Gregor, Jeremiah Willcock, and Andrew Lumsdaine. Concepts for the C++0x Standard Library: Introduction (revision 1). Technical Report N2618=08-0128, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, May 2008.

[43] Beman Dawes, Daniel Krügler, and Douglas Gregor. Concepts for the C++0x Standard Library: Diagnostics library. Technical Report N2620=08-0130, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, May 2008.

[44] Peter Dimov, Douglas Gregor, J. Järvi, and Gary Powell. A proposal to add an enhanced binder to the library technical report. Technical Report N1455=03-0038, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, : Programming Language C++, 2003.

[45] J. Dongarra, Andrew Lumsdaine, Xinhui Niu, Roldan Pozo, and Karin Remington. A sparse matrix library in C++ for high performance architectures. In *Proceedings Object Oriented Numerics Conference*, Sun River, OR, 1994.

[46] Steve Huss-Lederman (Editor). MPI-2: Extensions to the message passing interface. Technical report, MPI Forum, July 1997.

[47] Nick Edmonds, Alex Breuer, Douglas Gregor, and Andrew Lumsdaine. Single-source shortest paths with the parallel boost graph library. In *The Ninth DIMACS Implementation Challenge: The Shortest Path Problem*, Piscataway, NJ, November 2006.

[48] A. Friedley, T. Hoefler, M. Leininger, and A. Lumsdaine. Scalable High Performance Message Passing over InfiniBand for Open MPI. In *Proceedings of 3rd KiCC Workshop 2007*. RWTH Aachen, 12 2007.

[49] Andrew Friedley. Improving performance critical code development via synthetic programming. Presentation at Sandia National Laboratories, March 2009.

[50] Andrew Friedley, Christopher Mueller, and Andrew Lumsdaine. And now for something completely different... Presentation at SC 2008, November 2008.

[51] Andrew Friedley, Christopher Mueller, and Andrew Lumsdaine. Presenting corepy: A synthetic programming environment for python. Poster Presentation at CGO 2009, March 2009.

[52] Andrew Friedley, Christopher Mueller, Ben Martin, and Andrew Lumsdaine. High-performance code generation using corepy. August 2009. SciPy 2009.

[53] Daniel P. Friedman, Abdulaziz Ghuloum, Jeremy G. Siek, and Lynn Winebarger. Improving the lazy krivine machine. *Higher-Order and Symbolic Computation*, 2003. accepted for publication.

[54] Daniel P. Friedman, Abdulaziz Ghuloum, Jeremy G. Siek, and Lynn Winebarger. Improving the lazy krivine machine. Technical Report 581, Indiana University, November 2003. To appear in the journal, Higher Order and Symbolic Computation.

[55] Edgar Gabriel, Graham E. Fagg, George Bosilca, Thara Angskun, Jack J. Dongarra, Jeffrey M. Squyres, Vishal Sahay, Prabhanjan Kambadur, Brian Barrett, Andrew Lumsdaine, Ralph H. Castain, David J. Daniel, Richard L. Graham, and Timothy S. Woodall. Open MPI: Goals, concept, and design of a next generation mpi implementation. In *Proceedings, 11th European PVM/MPI Users' Group Meeting*, pages 97–104, Budapest, Hungary, September 2004.

[56] Ronald Garcia. Compile-time metaprogramming. Presentation at Connecticut College, December 2007.

[57] Ronald Garcia. Computing while compiling: Reasons and methods for compile-time metaprogramming. Presentation at Wesleyan University, December 2007.

[58] Ronald Garcia. A principled approach to compile-time metaprogramming. Poster presented during the CRA/CDC Programming Languages Summer School, May 2007.

[59] Ronald Garcia. Static computation and reflection: Practice and theory. Presentation at the 2007 LogicBlox Inc. Researchers Symposium, September 2007.

[60] Ronald Garcia. Static computation and reflection: Practice and theory. Presentation at the 2007 Summer School on Generative and Transformational Techniques in Software Engineering, Participants Workshop, July 2007.

[61] Ronald Garcia. Computing while compiling: Reasons and methods for compile-time metaprogramming. Presentation at University of Oregon, May 2008.

[62] Ronald Garcia. Computing while compiling: Reasons and methods for compile-time metaprogramming. Presentation at Rice University, May 2008.

[63] Ronald Garcia. Computing while compiling: Reasons and methods for compile-time metaprogramming. Presentation at Argonne National Laboratory, January 2008.

[64] Ronald Garcia. *Static Computation and Reflection*. PhD thesis, Indiana University, September 2008.

[65] Ronald Garcia, Jaakko Järvi, Andrew Lumsdaine, Jeremy Siek, and Jeremiah Willcock. An extended comparative study of language support for generic programming. *Journal of Functional Programming*, 17(2):145–205, March 2007.

[66] Ronald Garcia, Jaakko Järvi, Andrew Lumsdaine, Jeremy G. Siek, and Jeremiah Willcock. A comparative study of language support for generic programming. In *Proceedings of the 2003 ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications (OOPSLA'03)*, October 2003.

[67] Ronald Garcia and Andrew Lumsdaine. Multiarray: a c++ library for generic programming with arrays. *Softw., Pract. Exper.*, 35(2):159–188, 2005.

[68] Ronald Garcia and Andrew Lumsdaine. Type classes without types. In *2005 Workshop on Scheme and Functional Programming*, September 2005.

[69] Ronald Garcia and Andrew Lumsdaine. Toward foundations for type-reflective metaprogramming. In *GPCE '09: Proceedings of the eighth international conference on Generative programming and component engineering*, pages 25–34, New York, NY, USA, 2009. ACM.

[70] Ronald Garcia, Andrew Lumsdaine, and Amr Sabry. Lazy evaluation and delimited control. In *POPL '09: Proceedings of the 36th annual ACM SIGPLAN-SIGACT symposium on Principles of programming languages*, pages 153–164, New York, NY, USA, 2009. ACM.

[71] Al Geist, William Gropp, Steve Huss-Lederman, Andrew Lumsdaine, Ewing Lusk, William Saphir, Tony Skjellum, and Marc Snir. MPI-2: Extending the message-passing interface. In Luc Bouge, Pierre Fraigniaud, Anne Mignotte, and Yves Robert, editors, *Euro-Par '96 Parallel Processing*, number 1123 in Lecture Notes in Computer Science, pages 128–135. Springer Verlag, 1996.

[72] P. Geoffray and T. Hoefler. Adaptive Routing Strategies for Modern High Performance Networks. In *16th Annual IEEE Symposium on High Performance Interconnects (HOTI 2008)*, pages 165–172. IEEE Computer Society, Aug. 2008.

[73] Todor Georgiev, Chintan Intwala, Derin Babacan, and Andrew Lumsdaine. A unified frequency domain analysis of lightfield cameras. In *Proceedings European Converence on Computer Vision*, 2008. To appear.

[74] Todor G. Georgiev and Andrew Lumsdaine. Resolution in plenoptic cameras. In *Computational Optical Sensing and Imaging*, page CTuB3. Optical Society of America, 2009.

[75] Jens Gerlach, Peter Gottschling, and Uwe Der. A generic c++ framework for parallel mesh based scientific applications. In *6th International Workshop on High-Level Parallel Programming Models and Supportive Environments, San Francisco*, 2001.

[76] Jens Gerlach, Peter Gottschling, and Hans Werner Pohl. Core components of janus. Technical report, Technical Report D-00-028 of RWCP, 2000.

[77] Christopher Gottbrath, Brian Barrett, Bill Gropp, Ewing "Rusty" Lusk, and Jeff Squyres. An interface to support the identification of dynamic mpi 2 processes for scalable parallel debugging. In *Proceedings, 13th European PVM/MPI Users' Group Meeting*, Bonn, Germany, September 2006.

[78] Peter Gottschling. Fast sorting algorithms on parallel and sequential computers in linear time. *Wissenschaftliche Beiträge zur Informatik*, January 1995.

[79] Peter Gottschling. ANGEL – an extensible library for jacobian accumulation. In *4th International Conference on Automatic Differentiation, Chicago*, 2004.

[80] Peter Gottschling. Generic libraries to support FEM applications. Presentation at Extended Finite Element Tutorial, Lausanne, Switzerland, December 2005.

[81] Peter Gottschling. Fundamental algebraic concepts in concept-enabled C++. Technical Report 638, Indiana University, 2006.

[82] Peter Gottschling. Generic linear algebra. Presentation at ANL, Argonne, August 2006.

[83] Peter Gottschling. MTL and GLAS. Presentation at Summer School of CEA – EDF – INRIA on High-Performance Applications, Paris, France, June 2006.

[84] Peter Gottschling. MTL et GLAS. Presentation at INSA (presented in French), Lausanne, July 2006.

[85] Peter Gottschling. Generic linear algebra and MTL. Presentation at Katholieke Universiteit Leuven, Belgium, March 2007.

[86] Peter Gottschling. Algebraic concepts and generic high-performance computing. Presentation at University of Chicago, USA, November 2008.

[87] Peter Gottschling. Algebraic concepts and generic high-performance numerics. Presentation at University of Bergen, Norway, July 2008.

[88] Peter Gottschling. Concept implication and requirement propagation. Technical Report N2645=08-0155, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, June 2008.

[89] Peter Gottschling. Generic high-performance numerics. Presentation at INRIA, Sophia-Antipolis, France, July 2008.

[90] Peter Gottschling. Scientific computing in silicon (in german). Presentation at Silicon Saxony Day, Dresden, May 2009.

[91] Peter Gottschling and Walter E. Brown. Fundamental mathematical concepts for the stl in C++0x. Technical Report N2645=08-0155, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, June 2008.

[92] Peter Gottschling and Walter E. Brown. Toward a more complete taxonomy of algebraic properties for numeric libraries in TR2. Technical Report N2650=08-0160, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, June 2008.

[93] Peter Gottschling, Martin Gaber, and Ralf Müller. Numerical simulation of degassing processes from glass powders (in german). In *77th Conference of German Society of Glass Technology, Leipzig*, May 2003.

[94] Peter Gottschling, Martin Gaber, and Ralf Müller. Finite difference calculation of degassing from glass powders (in german). In *78th Conference of German Society of Glass Technology, Nürnberg*, 2004.

[95] Peter Gottschling and Peter-Wolfgang Gräber. Simulation of stationary and instationary ground-water flow with a parallel hierarchical cg-algorithm. *SAMS*, 25:127–144, 1996.

[96] Peter Gottschling and Dag Lindbo. Generic compressed sparse matrix insertion: Algorithms and implementations in MTL4 and FEniCS. In *POOSC 2009 workshop at ECOOP09, Genova, Italy*, ACM Digital Library (doi pending), 2009.

[97] Peter Gottschling and Andrew Lumsdaine. Integrating semantics and compilation: using c++ concepts to develop robust and efficient reusable libraries. In *GPCE '08: Proceedings of the 7th international conference on Generative programming and component engineering*, pages 67–76, New York, NY, USA, 2008. ACM.

[98] Peter Gottschling and Wolfgang E. Nagel. An efficient parallel linear solver with a cascadic conjugate gradient method: Experience with reality. In *European Parallel Computing Conference, Munich*, number 1900 in LNCS, 2000.

[99] Peter Gottschling, Wolfgang E. Nagel, and Matthias Tief. Domain decomposition of large problems on cray T3. In *6th European Cray/SGI Workshop, Manchester*, 2000.

[100] Peter Gottschling, David S. Wise, and Michael D. Adams. Representation-transparent matrix algorithms with scalable performance. In *ICS '07: Proceedings of the 21st annual international conference on Supercomputing*, pages 116–125, New York, NY, USA, 2007. ACM Press.

[101] Peter Gottschling, David S. Wise, and Adwait Joshi. Generic support of algorithmic and structural recursion for scientific computing. In *POOSC 2008 workshop at ECOOP08, Paphros, Cyprus*, 2008.

[102] Peter Gottschling, David S. Wise, and Adwait Joshi. Generic support of algorithmic and structural recursion for scientific computing. *The International Journal of Parallel, Emergent and Distributed Systems (IJPEDS)*, 2009. Accepted.

[103] Peter Gottschling, Thomas Witkowski, and Axel Voigt. Integrating object-oriented and generic programming paradigms in real-world software environments: Experiences with AMDiS and MTL4. In *POOSC 2008 workshop at ECOOP08, Paphros, Cyprus*, 2008.

[104] Richard L. Graham, Brian W. Barrett, Galen M. Shipman, and Timothy S. Woodall. Open mpi: A high performance, flexible implementation of mpi point-to-point communications. In *Proceedings, Clusters and Computational Grids for cientific Computing*, Flat Rock, North Carolina, September 2006.

[105] Richard L. Graham, Ron Brightwell, Brian Barrett, George Bosilca, and Pješivac-Grbović. An evaluation of open mpi's matching transport layer on the cray xt. Oct 2007.

[106] Richard L. Graham, Galen M. Shipman, Brian W. Barrett, Ralph H. Castain, and George Bosilca. Open mpi: A high performance, heterogeneous mpi. In *Proceedings, Fifth International Workshop on Algorithms, Models and Tools for Parallel Computing on Heterogeneous Networks*, Barcelona, Spain, September 2006.

[107] Richard L. Graham, Timothy S. Woodall, and Jeffrey M. Squyres. Open MPI: A flexible high performance MPI. In *Proceedings, 6th Annual International Conference on Parallel Processing and Applied Mathematics*, Poznan, Poland, September 2005.

[108] D. Gregor, T. Hoefler, B. Barrett, and A. Lumsdaine. Fixing Probe for Multi-Threaded MPI Applications. Technical Report 674, Indiana University, Jan. 2009.

[109] Douglas Gregor. *High-level Static Analysis for Generic Libraries*. PhD thesis, Rensselaer Polytechnic Institute, May 2004.

[110] Douglas Gregor. The boost graph library. Presentation at SciPy 2006, August 2006.

[111] Douglas Gregor. A brief introduction to variadic templates. Technical Report N2087=06-0157, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, October 2006.

[112] Douglas Gregor. Concepts. Presentation to ANSI/ISO C++ Standard Committee, Evolution Working Group, Berlin, Germany, April 2006.

[113] Douglas Gregor. Concepts and the standard library. Presentation to ANSI/ISO C++ Standard Committee, Library Working Group, Berlin, Germany, April 2006.

[114] Douglas Gregor. Concepts for the C++0x Standard Library: Containers. Technical Report N2085=06-0155, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, October 2006.

[115] Douglas Gregor. Conceptualizing the range-based for loop. Technical Report N2049=06-0119, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, June 2006.

[116] Douglas Gregor. Large-scale network analysis with the boost graph libraries. Presentation at Networks and Complex Systems talk series, Indiana University, Bloomington, Indiana, February 2006.

[117] Douglas Gregor. Signals and slots for library tr2. Technical Report N2086=06-0156, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, October 2006.

[118] Douglas Gregor. Boost.mpi: Modernizing the C++ interface to mpi. Presentation to Microsoft High-Performance Computing Group, Redmond, Washington, February 2007.

[119] Douglas Gregor. Concepts: Extending C++ templates for generic programming. Presentation to Microsoft Visual C++ Development Team, Redmond, Washington, February 2007.

[120] Douglas Gregor. Concepts: Extending C++ templates for generic programming. Presentation to Apple Developer Tools Group, Cupertino, California, February 2007.

[121] Douglas Gregor. Concepts: Extending C++ templates for generic programming. Presentation as Google Tech Talk, Mountain View, California, February 2007.

[122] Douglas Gregor. Concepts: Extending C++ templates for generic programming. Presentation at ACCU 2007, Oxford, U.K., April 2007.

[123] Douglas Gregor. Evolving a C++ library to C++0x concepts. Presentation at BoostCon 2007, Aspen, Colorado, May 2007.

[124] Douglas Gregor. Generic programming in the Parallel Boost Graph Library. Presentation to Microsoft Live Labs, Bellevue, Washington, November 2007.

[125] Douglas Gregor. An introduction to concepts in C++0x. Presentation at BoostCon 2007, Aspen, Colorado, May 2007.

[126] Douglas Gregor. Large-scale network analysis with the parallel boost graph library. Center for Data and Search Informatics Seminar, Bloomington, Indiana, September 2007.

[127] Douglas Gregor. MPI.NET: High-performance message passing in C# and .NET. Presentation to Microsoft High-Performance Computing Group, Redmond, Washington, November 2007.

[128] Douglas Gregor. A tour of the concepts wording. Technical Report N2399=07-0259, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, September 2007.

[129] Douglas Gregor. Author's corner: Boost.Function. Presentation at BoostCon 2008, May 2008.

[130] Douglas Gregor. C++0x Today: Features for building better libraries. Presentation at BoostCon 2008, May 2008.

[131] Douglas Gregor. Easier C++: An introduction to concepts. *DevX.com Special Report*, August 2008. `http://www.devx.com/SpecialReports/Article/38864`.

[132] Douglas Gregor. Introduction to parallel programming with the Message Passing Interface. Presentation at BoostCon 2008, May 2008.

[133] Douglas Gregor. MPI.NET: High-performance message passing in C# and .NET. Presentation at Principles and Practice of Parallel Programming, February 2008.

[134] Douglas Gregor. Simplifying unique copy. Technical Report N2742=08-0252, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, July 2008.

[135] Douglas Gregor. Simplifying unique copy (Revision 1). Technical Report N2786=08-0298, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, October 2008.

[136] Douglas Gregor. Type-soundness and optimization in the concepts proposal. Technical Report N2576=08-0086, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, March 2008.

[137] Douglas Gregor. Unifying operator and function-object variants of standard library algorithms. Technical Report N2743=08-0253, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, July 2008.

[138] Douglas Gregor and Pablo Halpern. Concepts for the C++0x Standard Library: Containers (revision 1). Technical Report N2623=08-0133, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, May 2008.

[139] Douglas Gregor and Jaakko Järvi. decltype for the C++0x standard library. Technical Report N2194=07-0054, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, March 2007.

[140] Douglas Gregor and Jaakko Järvi. Variadic templates for C++. In *Object-Oriented Programming Languages and Systems*. ACM Press, March 2007.

[141] Douglas Gregor and Jaakko Järvi. Variadic templates for the C++0x standard library. Technical Report N2151=07-0011, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, January 2007.

[142] Douglas Gregor and Jaakko Järvi. Variadic templates for the C++0x standard library (revision 1). Technical Report N2192=07-0052, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, March 2007.

[143] Douglas Gregor and Jaakko Järvi. Variadic templates for C++0x. *Journal of Object Technology*, 7(2):31–51, February 2008.

[144] Douglas Gregor, Jaakko Järvi, Mayuresh Kulkarni, Andrew Lumsdaine, David Musser, and Sibylle Schupp. Generic programming and high-performance libraries. *International Journal of Parallel Programming*, 33(2), June 2005.

[145] Douglas Gregor, Jaakko Järvi, Jens Maurer, and Jason Merrill. Proposed wording for variadic templates. Technical Report N2152=07-0012, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, January 2007.

[146] Douglas Gregor, Jaakko Järvi, Jens Maurer, and Jason Merrill. Proposed wording for variadic templates (revision 1). Technical Report N2191=07-0051, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, March 2007.

[147] Douglas Gregor, Jaakko Järvi, Jens Maurer, and Jason Merrill. Proposed wording for variadic templates (revision 2). Technical Report N2242=07-0102, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, May 2007.

[148] Douglas Gregor, Jaakko Järvi, and Gary Powell. Variadic templates (revision 3). Technical Report N2080=06-0150, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, October 2006.

[149] Douglas Gregor, Jaakko Järvi, Jeremy Siek, Bjarne Stroustrup, Gabriel Dos Reis, and Andrew Lumsdaine. Concepts: Linguistic support for generic programming in C++. In *Proceedings of the 2006 ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications (OOPSLA '06)*, pages 291–310. ACM Press, October 2006. This is the author's version of the work. It is posted here by permission of ACM for your personal use. Not for redistribution.

[150] Douglas Gregor and Andrew Lumsdaine. The execution instance overloading pattern. In *Workshop on Patterns in High-Performance Computing*, May 2005.

[151] Douglas Gregor and Andrew Lumsdaine. Lifting sequential graph algorithms for distributed-memory parallel computation. In *Proceedings of the 2005 ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications (OOPSLA '05)*, pages 423–437, October 2005.

[152] Douglas Gregor and Andrew Lumsdaine. The Parallel BGL: A generic library for distributed graph computations. In *Parallel Object-Oriented Scientific Computing (POOSC)*, July 2005. Accepted.

[153] Douglas Gregor and Andrew Lumsdaine. The parallel boost graph library. Presentation at the SIAM Conference on Parallel Processing, San Francisco, California, February 2006.

[154] Douglas Gregor and Andrew Lumsdaine. Concepts for the C++0x Standard Library: Utilities (revision 2). Technical Report N2322=07-0182, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, July 2007.

[155] Douglas Gregor and Andrew Lumsdaine. Concepts for the C++0x Standard Library: Algorithms (revision 2). Technical Report N2573=08-0083, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, March 2008.

[156] Douglas Gregor and Andrew Lumsdaine. Concepts for the C++0x Standard Library: Iterators (revision 3). Technical Report N2734=08-0244, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, July 2008.

[157] Douglas Gregor and Andrew Lumsdaine. Concepts for the C++0x Standard Library: Iterators (revision 4). Technical Report N2777=08-0287, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, October 2008.

[158] Douglas Gregor and Andrew Lumsdaine. Concepts for the C++0x Standard Library: Numerics (revision 1). Technical Report N2574=08-0084, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, March 2008.

[159] Douglas Gregor and Andrew Lumsdaine. Concepts for the C++0x Standard Library: Numerics (revision 2). Technical Report N2626=08-0136, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, May 2008.

[160] Douglas Gregor and Andrew Lumsdaine. Concepts for the C++0x Standard Library: Numerics (revision 3). Technical Report N2736=08-0246, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, July 2008.

[161] Douglas Gregor and Andrew Lumsdaine. Concepts for the C++0x Standard Library: Utilities (revision 3). Technical Report N2622=08-0132, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, May 2008.

[162] Douglas Gregor and Andrew Lumsdaine. Concepts for the C++0x Standard Library: Utilities (revision 4). Technical Report N2735=08-0245, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, July 2008.

[163] Douglas Gregor and Andrew Lumsdaine. Concepts for the C++0x Standard Library: Utilities (revision 5). Technical Report N2770=08-0280, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, October 2008.

[164] Douglas Gregor and Andrew Lumsdaine. Core concepts for the c++0x standard library. Technical Report N2502=08-0012, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, February 2008.

[165] Douglas Gregor and Andrew Lumsdaine. Core concepts for the c++0x standard library (revision 1). Technical Report N2572=08-0082, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, March 2008.

[166] Douglas Gregor and Andrew Lumsdaine. Core concepts for the c++0x standard library (revision 2). Technical Report N2621=08-0131, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, May 2008.

[167] Douglas Gregor and Andrew Lumsdaine. Design and implementation of a high-performance MPI for C# and the common language infrastructure. In *PPoPP '08: Proceedings of the 13th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming*, pages 133–142, New York, NY, USA, February 2008. ACM.

[168] Douglas Gregor, Mat Marcus, and Pablo Halpern. Concepts for the C++0x Standard Library: Containers (revision 2). Technical Report N2694=08-0204, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, June 2008.

[169] Douglas Gregor, Mat Marcus, and Pablo Halpern. Concepts for the C++0x Standard Library: Containers (revision 3). Technical Report N2738=08-0248, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, July 2008.

[170] Douglas Gregor, Mat Marcus, and Pablo Halpern. Concepts for the C++0x Standard Library: Containers (revision 4). Technical Report N2776=08-0286, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, October 2008.

[171] Douglas Gregor, Mat Marcus, and Thomas Witt Andrew Lumsdaine. Foundational concepts for the C++0x Standard Library (revision 3). Technical Report N2737=08-0247, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, July 2008.

[172] Douglas Gregor, Mat Marcus, and Thomas Witt Andrew Lumsdaine. Foundational concepts for the C++0x Standard Library (revision 5). Technical Report N2774=08-02484, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, October 2008.

[173] Douglas Gregor, Mat Marcus, Thomas Witt, and Andrew Lumsdaine. Concepts for the C++0x Standard Library: Algorithms (revision 2). Technical Report N2696=08-0206, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, June 2008.

[174] Douglas Gregor, Mat Marcus, Thomas Witt, and Andrew Lumsdaine. Concepts for the C++0x Standard Library: Algorithms (revision 4). Technical Report N2740=08-0250, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, July 2008.

[175] Douglas Gregor, Mat Marcus, Thomas Witt, and Andrew Lumsdaine. Concepts for the C++0x Standard Library: Algorithms (revision 5). Technical Report N2759=08-0269, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, October 2008.

[176] Douglas Gregor, Mat Marcus, Thomas Witt, and Andrew Lumsdaine. Foundational concepts for the c++0x standard library (revision 3). Technical Report N2677=08-0187, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, June 2008.

[177] Douglas Gregor and Eric Niebler. Extending variadic template template parameters. Technical Report N2488=07-0358, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, December 2007.

[178] Douglas Gregor and Eric Niebler. Extending variadic template template parameters (revision 1). Technical Report N2555=08-0065, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, March 2008.

[179] Douglas Gregor, Gary Powell, and J. Järvi. Typesafe variable-length function and template argument lists. Technical Report N1483=03-0066, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, : Programming Language C++, 2003.

[180] Douglas Gregor and Sibylle Schupp. Making the usage of STL safe. In Jeremy Gibbons and Johan Jeuring, editors, *Generic Programming, IFIP TC2/WG2.1 Working Conference on Generic Programming*, volume 243 of *IFIP Conference Proceedings*, pages 127–140. Kluwer, July 2002.

[181] Douglas Gregor and Sibylle Schupp. Retaining path-sensitive relations across control-flow merges. Technical Report 03-15, Rensselaer Polytechnic Institute, November 2003.

[182] Douglas Gregor and Sibylle Schupp. STLlint: Lifting static checking from languages to libraries. *Software: Practice & Experience*, 36:225–254, 2006.

[183] Douglas Gregor, Sibylle Schupp, and David R. Musser. Base class injection. In Jan Bosch, editor, *Proceedings of the Third International Conference on Generative and Component-Based Software Engineering*, number 2186 in LNCS, pages 106–117. Springer-Verlag, Berlin Heidelberg, September 2001.

[184] Douglas Gregor, Sibylle Schupp, and David R. Musser. Design patterns for library optimizations. In J.Strieglitz K. Davis, editor, *Proceedings International Conference on Parallel/High-Performance Object-Oriented Scientific Computing (POOSC'01) Tampa, FL, 2001*, 2001.

[185] Douglas Gregor, Sibylle Schupp, and David R. Musser. Design patterns for library optimization. *Scientific Programming*, 11(4):309–320, 2003.

[186] Douglas Gregor and Jeremy Siek. Explicit model definitions are necessary. Technical Report N1798=05-0058, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, May 2005.

[187] Douglas Gregor and Jeremy Siek. Implementing concepts. Technical Report N1848=05-0108, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, August 2005.

[188] Douglas Gregor, Jeremy Siek, and Andrew Lumsdaine. Concepts for the C++0x Standard Library: Iterators (revision 2). Technical Report N2323=07-0183, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, July 2007.

[189] Douglas Gregor, Jeremy Siek, and Andrew Lumsdaine. Iterator concepts for the c++0x standard library. Technical Report N2500=08-0010, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, February 2008.

[190] Douglas Gregor, Jeremy Siek, and Andrew Lumsdaine. Iterator concepts for the c++0x standard library (revision 1). Technical Report N2570=08-0080, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, March 2008.

[191] Douglas Gregor, Jeremy Siek, and Andrew Lumsdaine. Iterator concepts for the c++0x standard library (revision 2). Technical Report N2624=08-0134, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, May 2008.

[192] Douglas Gregor, Jeremy Siek, and Andrew Lumsdaine. Iterator concepts for the c++0x standard library (revision 3). Technical Report N2695=08-0205, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, June 2008.

[193] Douglas Gregor, Jeremy Siek, and Andrew Lumsdaine. Iterator concepts for the C++0x Standard Library (revision 4). Technical Report

N2739=08-0249, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, July 2008.

[194] Douglas Gregor, Jeremy Siek, and Andrew Lumsdaine. Iterator concepts for the C++0x Standard Library (revision 5). Technical Report N2758=08-0268, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, October 2008.

[195] Douglas Gregor, Jeremy Siek, Jeremiah Willcock, Jaakko Järvi, Ronald Garcia, and Andrew Lumsdaine. Concepts for C++0x (revision 1). Technical Report N1849=05-0109, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, August 2005.

[196] Douglas Gregor and Bjarne Stroustrup. Concepts. Technical Report N2042=06-0112, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, June 2006.

[197] Douglas Gregor and Bjarne Stroustrup. Concepts (revision 1). Technical Report N2081=06-0151, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, October 2006.

[198] Douglas Gregor and Bjarne Stroustrup. Proposed wording for concepts. Technical Report N2193=07-0053, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, March 2007.

[199] Douglas Gregor and Bjarne Stroustrup. Proposed wording for concepts (revision 1). Technical Report N2307=07-0167, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, July 2007.

[200] Douglas Gregor and Bjarne Stroustrup. Proposed wording for concepts (revision 2). Technical Report N2398=07-0258, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, September 2007.

[201] Douglas Gregor and Bjarne Stroustrup. Proposed wording for concepts (revision 3). Technical Report N2421=07-0281, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, October 2007.

[202] Douglas Gregor, Bjarne Stroustrup, Jeremy Siek, and James Widman. Proposed wording for concepts (revision 4). Technical Report N2501=08-0011, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, February 2008.

[203] Douglas Gregor, Bjarne Stroustrup, James Widman, and Jeremy Siek. Proposed wording for concepts (revision 5). Technical Report N2617=08-0127, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, May 2008.

[204] Douglas Gregor, Bjarne Stroustrup, James Widman, and Jeremy Siek. Proposed wording for concepts (revision 6). Technical Report N2676=08-0186, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, June 2008.

[205] Douglas Gregor, Bjarne Stroustrup, James Widman, and Jeremy Siek. Proposed wording for concepts (revision 7). Technical Report N2710=08-0220, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, July 2008.

[206] Douglas Gregor, Bjarne Stroustrup, James Widman, and Jeremy Siek. Proposed wording for concepts (revision 8). Technical Report N2741=08-0251, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, July 2008.

[207] Douglas Gregor, Bjarne Stroustrup, James Widman, and Jeremy Siek. Proposed wording for concepts (revision 9). Technical Report N2773=08-0283, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, October 2008.

[208] Douglas Gregor, Jeremiah Willcock, and Andrew Lumsdaine. Concepts for the C++0x Standard Library: Algorithms. Technical Report N2040=06-0110, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, June 2006.

[209] Douglas Gregor, Jeremiah Willcock, and Andrew Lumsdaine. Concepts for the C++0x Standard Library: Algorithms (revision 1). Technical Report N2084=06-0154, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, October 2006.

[210] Douglas Gregor, Jeremiah Willcock, and Andrew Lumsdaine. Concepts for the C++0x Standard Library: Approach. Technical Report N2036=06-0106, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, June 2006.

[211] Douglas Gregor, Jeremiah Willcock, and Andrew Lumsdaine. Concepts for the C++0x Standard Library: Introduction. Technical Report N2037=06-0107, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, June 2006.

[212] Douglas Gregor, Jeremiah Willcock, and Andrew Lumsdaine. Concepts for the C++0x Standard Library: Iterators. Technical Report N2039=06-0109, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, June 2006.

[213] Douglas Gregor, Jeremiah Willcock, and Andrew Lumsdaine. Concepts for the C++0x Standard Library: Iterators (revision 1). Technical Report N2083=06-0153, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, October 2006.

[214] Douglas Gregor, Jeremiah Willcock, and Andrew Lumsdaine. Concepts for the C++0x Standard Library: Numerics. Technical Report N2041=06-0111, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, June 2006.

[215] Douglas Gregor, Jeremiah Willcock, and Andrew Lumsdaine. Concepts for the C++0x Standard Library: Utilities. Technical Report N2038=06-0108, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, June 2006.

[216] Douglas Gregor, Jeremiah Willcock, and Andrew Lumsdaine. Concepts for the C++0x Standard Library: Utilities (revision 1). Technical Report N2082=06-0152, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, October 2006.

[217] William Gropp, Steven Huss-Lederman, Andrew Lumsdaine, Ewing Lusk, Bill Nitzberg, William Saphir, , and Marc Snir. *MPI — The Complete Reference: Volume 2, the MPI-2 Extensions*. MIT Press, 1998.

[218] Randy Heiland. CompuCell3D: an open source cellular modeling framework. Presentation at the Supercomputing 2008 Conference, Austin, TX, November 2008.

[219] T. Hoefler. Remote Network Analysis. In *21C3 Proceedings*, pages 33–37, 12 2004.

[220] T. Hoefler. Evaluation of publicly available Barrier-Algorithms and Improvement of the Barrier-Operation for large-scale Cluster-Systems with special Attention on InfiniBand Networks, Apr. 2005.

[221] T. Hoefler. The Cell Processor. In *22C3 Proceedings*, pages 286–292, 12 2005.

[222] T. Hoefler. Application Optimization with non-blocking Collectives. Presentation at parallel applications group at the Commissariat a l'Energie Atomique - Direction des applications militaires (CEA-DAM), Bruyeres-le-chatel, France, 01 2007.

[223] T. Hoefler. Non-Blocking Collectives for MPI-2. Presentation at parallel systems group at the Commissariat a l'Energie Atomique - Direction des applications militaires (CEA-DAM), Bruyeres-le-chatel, France, 01 2007.

[224] T. Hoefler. *Principles for Coordinated Optimization of Computation and Communication in Large-Scale Parallel Systems*. PhD thesis, Sep. 2008.

[225] T. Hoefler, L. Cerquetti, T. Mehlan, F. Mietke, and W. Rehm. A practical approach to the rating of barrier algorithms using the LogP model and Open-MPI. In *Proceedings of the 2005 International Conference on Parallel Processing Workshops*, pages 562–569, 06 2005.

[226] T. Hoefler, P. Gottschling, A. Lumsdaine, and W. Rehm. Optimizing a Conjugate Gradient Solver with Non-Blocking Collective Operations. *Elsevier Journal of Parallel Computing (PARCO)*, 33(9):624–633, 9 2007.

[227] T. Hoefler, P. Gottschling, W. Rehm, and A. Lumsdaine. Optimizing a Conjugate Gradient Solver with Non-Blocking Collective Operations. In *Recent Advantages in Parallel Virtual Machine and Message Passing Interface. 13th European PVM/MPI User's Group Meeting, Proceedings, LNCS 4192*, pages 374–382. Springer, 9 2006.

[228] T. Hoefler, R. Janisch, and W. Rehm. *A Performance Analysis of ABINIT on a Cluster System*, pages 37–51. Springer, Lecture Notes in Computational Science and Engineering, 12 2005.

[229] T. Hoefler, R. Janisch, and W. Rehm. *Improving the parallel scaling of ABINIT*, pages 551–559. CINECA Conzorzio Interuniversitario, 12 2005.

[230] T. Hoefler, R. Janisch, and W. Rehm. Parallel scaling of Teter's minimization for Ab Initio calculations. 11 2006. Presented at the workshop HPC Nano in conjunction with SC'06.

[231] T. Hoefler, A. Lichei, and W. Rehm. Low-Overhead LogGP Parameter Assessment for Modern Interconnection Networks. In *Proceedings of the 21st IEEE International Parallel & Distributed Processing Symposium*. IEEE Computer Society, 03 2007.

[232] T. Hoefler, F. Lorenzen, and A. Lumsdaine. Sparse Non-Blocking Collectives in Quantum Mechanical Calculations. In *Recent Advances in Parallel Virtual Machine and Message Passing Interface, 15th European PVM/MPI Users' Group Meeting*, volume LNCS 5205, pages 55–63. Springer, Sep. 2008.

[233] T. Hoefler and A. Lumsdaine. Design, Implementation, and Usage of LibNBC. Technical report, Open Systems Lab, Indiana University, 08 2006.

[234] T. Hoefler and A. Lumsdaine. Message Progression in Parallel Computing - To Thread or not to Thread? In *Proceedings of the 2008 IEEE International Conference on Cluster Computing*. IEEE Computer Society, Oct. 2008.

[235] T. Hoefler and A. Lumsdaine. Optimizing non-blocking Collective Operations for InfiniBand. In *Proceedings of the 22nd IEEE International Parallel & Distributed Processing Symposium (IPDPS)*, 04 2008.

[236] T. Hoefler and A. Lumsdaine. Overlapping Communication and Computation with High Level Communication Routines. In *Proceedings of the 8th IEEE Symposium on Cluster Computing and the Grid (CCGrid 2008)*, 05 2008.

[237] T. Hoefler, A. Lumsdaine, and J. Dongarra. Towards Efficient MapReduce Using MPI. In *Recent Advances in Parallel Virtual Machine and Message Passing Interface, 16th European PVM/MPI Users' Group Meeting*. Springer, Sep. 2009.

[238] T. Hoefler, T. Mehlan, F. Mietke, and W. Rehm. Adding Low-Cost Hardware Barrier Support to Small Commodity Clusters. In *Proceedings of 19th International Conference on Architecture and Computing Systems - ARCS'06*, pages 343–250, 3 2006.

[239] T. Hoefler, T. Mehlan, F. Mietke, and W. Rehm. Fast Barrier Synchronization for InfiniBand. In *Proceedings of the 20th IEEE International Parallel & Distributed Processing Symposium (IPDPS)*, 4 2006.

[240] T. Hoefler, T. Mehlan, F. Mietke, and W. Rehm. LogfP - A Model for small Messages in InfiniBand. In *Proceedings of the 20th IEEE International Parallel & Distributed Processing Symposium (IPDPS)*, 4 2006.

[241] T. Hoefler and W. Rehm. A Communication Model for Small Messages with InfiniBand. In *PARS Mitteilungen*, pages 32–41. PARS, 06 2005. (Awarded with the PARS Junior Researcher Prize).

[242] T. Hoefler, M. Reinhardt, F. Mietke, T. Mehlan, and W. Rehm. Low Overhead Ethernet Communication for Open MPI on Linux Clusters. Technical Report 06, TU Chemnitz, 7 2006.

[243] T. Hoefler, M. Schellmann, S. Gorlatch, and A. Lumsdaine. Communication Optimization for Medical Image Reconstruction Algorithms. In *Recent Advances in Parallel Virtual Machine and Message Passing Interface, 15th European PVM/MPI Users' Group Meeting*, volume LNCS 5205, pages 75–83. Springer, Sep. 2008.

[244] T. Hoefler, T. Schneider, and A. Lumsdaine. Accurately Measuring Collective Operations at Massive Scale. In *Proceedings of the 22nd IEEE International Parallel & Distributed Processing Symposium (IPDPS)*, 04 2008.

[245] T. Hoefler, T. Schneider, and A. Lumsdaine. Multistage Switches are not Crossbars: Effects of Static Routing in High-Performance Networks. In *Proceedings of the 2008 IEEE International Conference on Cluster Computing*. IEEE Computer Society, Oct. 2008.

[246] T. Hoefler, T. Schneider, and A. Lumsdaine. A Power-Aware, Application-Based, Performance Study Of Modern Commodity Cluster Interconnection Networks. In *Proceedings of the 23rd IEEE International Parallel & Distributed Processing Symposium (IPDPS), CAC Workshop*, May 2009.

[247] T. Hoefler, T. Schneider, and A. Lumsdaine. LogGP in Theory and Practice - An In-depth Analysis of Modern Interconnection Networks and

Benchmarking Methods for Collective Operations. *Elsevier Journal of Simulation Modelling Practice and Theory*, Jun. 2009.

[248] T. Hoefler, T. Schneider, and A. Lumsdaine. Optimized Routing for Large-Scale InfiniBand Networks. In *17th Annual IEEE Symposium on High Performance Interconnects (HOTI 2009)*, Aug. 2009.

[249] T. Hoefler, T. Schneider, and A. Lumsdaine. The Effect of Network Noise on Large-Scale Collective Communications. *Parallel Processing Letters (PPL)*, Dec. 2009. Accepted for publication (in print).

[250] T. Hoefler, T. Schneider, and A. Lumsdaine. The Impact of Network Noise at Large-Scale Communication Performance. In *Proceedings of the 23rd IEEE International Parallel & Distributed Processing Symposium (IPDPS), LSPP Workshop*, May 2009.

[251] T. Hoefler, C. Siebert, and A. Lumsdaine. Group Operation Assembly Language - A Flexible Way to Express Collective Communication. In *In proceedings of the 38th International Conference on Parallel Processing (ICPP-2009)*, Sep. 2009.

[252] T. Hoefler, C. Siebert, and A. Lumsdaine. Scalable Communication Protocols for Dynamic Sparse Data Exchange. Jan. 2010. Accepted at the 2010 International Conference on Principles and Practice of Parallel Programming (PPoPP'10).

[253] T. Hoefler, C. Siebert, and W. Rehm. A practically constant-time MPI Broadcast Algorithm for large-scale InfiniBand Clusters with Multicast. In *Proceedings of the 21st IEEE International Parallel & Distributed Processing Symposium*, page 232. IEEE Computer Society, 03 2007.

[254] T. Hoefler, J. Squyres, G. Bosilca, G. Fagg, A. Lumsdaine, and W. Rehm. Non-Blocking Collective Operations for MPI-2. Technical report, Open Systems Lab, Indiana University, 08 2006.

[255] T. Hoefler, J. Squyres, G. Fagg, G. Bosilca, W. Rehm, and A. Lumsdaine. A New Approach to MPI Collective Communication Implementations. 09 2006. Accepted for Publication at the 6th Austrian-Hungarian Workshop on Distributed and Parallel Systems.

[256] T. Hoefler, J. Squyres, W. Rehm, and A. Lumsdaine. A Case for Non-Blocking Collective Operations. In *Frontiers of High Performance Computing and Networking - ISPA 2006 Workshops*, volume 4331/2006, pages 155–164. Springer Berlin / Heidelberg, 12 2006.

[257] T. Hoefler and J. L. Traeff. Sparse Collective Operations for MPI. In *Proceedings of the 23rd IEEE International Parallel & Distributed Processing Symposium (IPDPS), HIPC Workshop*, May 2009.

[258] T. Hoefler, C. Viertel, T. Mehlan, F. Mietke, and W. Rehm. Assessing Single-Message and Multi-Node Communication Performance of Infini-Band. In *Proceedings of IEEE Inernational Conference on Parallel Computing in Electrical Engineering, PARELEC 2006*, pages 227–232. IEEE Computer Society, 9 2006.

[259] T. Hoefler and G. Zerah. Optimization of a parallel 3d-FFT with non-blocking collective operations. Invited presentation at the 3rd International ABINIT Developer Workshop, Liege, Belgium, 01 2007.

[260] Torsten Hoefler, Prabhanjan Kambadur, Richard L Graham, Galen Shipman, and Andrew Lumsdaine. A case for standard non-blocking collective operations. In *Proceedings of the 14th European PVM/MPI Users' Group Meeting*, LNCS, pages 125–134, Paris, France, 2007.

[261] Torsten Hoefler, Jeffrey M. Squyres, Torsten Mehlan, Frank Mietke, and Wolfgang Rehm. Implementing a hardware-based barrier in Open MPI. In *Proceedings of KiCC'05, Chemnitzer Informatik Berichte*, November 2005.

[262] Joshua Hursey, Ethan Mallove, Jeffrey M. Squyres, and Andrew Lumsdaine. An extensible framework for distributed testing of mpi implementations. In *Proceedings, Euro PVM/MPI*, Paris, France, October 2007.

[263] Joshua Hursey, Timothy I. Mattox, and Andrew Lumsdaine. Interconnect agnostic checkpoint/restart in Open MPI. In *Proceedings of the Eighteenth International Symposium on High Performance Distributed Computing (HPDC 2009)*. ACM, June 2009.

[264] Joshua Hursey, Jeffrey M. Squyres, and Andrew Lumsdaine. A checkpoint and restart service specification for Open MPI. Technical Report TR635, Indiana University, Bloomington, Indiana, USA, July 2006.

[265] Joshua Hursey, Jeffrey M. Squyres, Timothy I. Mattox, and Andrew Lumsdaine. The design and implementation of checkpoint/restart process fault tolerance for Open MPI. In *Proceedings of the 21st IEEE International Parallel and Distributed Processing Symposium (IPDPS)*. IEEE Computer Society, March 2007.

[266] J. A. Izaguirre, Q. Ma, T. Matthey, J. Willcock, T. Slabach, B. Moore, and G. Viamontes. Overcoming instabilities in Verlet-I/r-RESPA with the mollified impulse method. In T. Schlick and H. H. Gan, editors, *Proceedings of 3rd International Workshop on Methods for Macromolecular Modeling*, volume 24 of *Lecture Notes in Computational Science and Engineering*, pages 146–174. Springer-Verlag, Berlin, New York, 2002.

[267] Assad Jarrahian, Gary Moore, Murugan Pandian, and Marcin Zalewski. Fast concept analysis algorithm: STL/C++ implementation, 2001. Undergraduate project report.

[268] J. Järvi. Processing sparse vectors during compile time in C++. In *Scientific Computing in Object-Oriented Parallel Environments*, volume 1343, pages 41–48, 1997.

[269] J. Järvi. Compile time recursive objects in C++. In *Technology of Object-Oriented Languages and Systems*, pages 66–77. IEEE Computer Society Press, 1998.

[270] J. Järvi. Object-oriented model for partially separable functions in parameter estimation. *Acta Cybernetica*, 14(2):285–302, 1999.

[271] J. Järvi. C++ function object binders made easy. In *Proceedings of the Generative and Component-Based Software Engineering'99*, volume 1799, pages 165–177, Berlin, Germany, August 2000. Springer.

[272] J. Järvi. Tuple types and multiple return values. *C/C++ Users Journal*, 19:24–35, August 2001.

[273] J. Järvi. Proposal for adding tuple types into the standard library. Technical Report N1403=02-0061, ISO/IEC JTC 1, Information technology, Subcommittee SC 22, Programming Language C++, 2002.

[274] J. Järvi, A. Lumsdaine, and D. S. Wise, editors. *MSPLS 2002: Proceedings of the Workshop of the Midwest Society for Programming Languages and Systems*, April 2002. Technical Report 560. in the Indiana University Computer Science TR Series.

[275] J. Järvi, S. Nyman, M. Komu, and J. J. Forsström. A PC-program for automatic analysis of NMR spectrum series. *Computer Methods and Programs in Biomedicine*, 52:213–222, 1997.

[276] J. Järvi and G. Powell. The lambda library : Lambda abstraction in C++. In *Proceedings of the Second Workshop on C++ Template Programming (TMPW'01) at OOPSLA 2001*, October 2001.

[277] J. Järvi and G. Powell. Side effects and partial function application in C++. In *Proceedings of the Multiparadigm Programming with OO Languages Workshop (MPOOL'01) at ECOOP 2001*, John von Neumann Institute of Computing series, pages 43–60, 2001.

[278] J. Järvi, G. Powell, and A. Lumsdaine. The Lambda Library: unnamed functions in C++. *Software—Practice and Experience*, 33:259–291, 2003.

[279] J. Järvi and B. Stroustrup. Mechanisms for querying types of expressions: Decltype and auto revisited. Technical Report N1527=03-0110, ISO/IEC JTC 1, Information technology, Subcommittee SC 22, Programming Language C++, September 2003.

[280] J. Järvi, Bjarne Stroustrup, Douglas Gregor, and Jeremy Siek. Decltype and auto. Technical Report N1478=03-0061, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, : Programming Language C++, 2003.

[281] J. Järvi, J. Willcock, H. Hinnant, and A. Lumsdaine. Function overloading based on arbitrary properties of types. *C/C++ Users Journal*, 21(6):25–32, June 2003.

[282] Jaakko Järvi, Douglas Gregor, Jeremiah Willcock, Andrew Lumsdaine, and Jeremy Siek. Algorithm specialization in generic programming: Challenges of constrained generics in C++. In *PLDI '06: Proceedings of the 2006 ACM SIGPLAN conference on Programming language design and implementation*, pages 272–282, New York, NY, USA, 2006. ACM Press.

[283] Jaakko Järvi, Andrew Lumsdaine, Douglas Gregor, M. Kulkarni, David Musser, and Sibylle Schupp. Generic programming and high-performance libraries. In *Workshop on Next Generation Software at IPDPS 2004, Santa Fe, New Mexico*, 2004.

[284] Jaakko Järvi, Andrew Lumsdaine, Jeremy Siek, and Jeremiah Willcock. An analysis of constrained polymorphism for generic programming. In Kei Davis and Jörg Striegnitz, editors, *Multiparadigm Programming 2003: Proceedings of the MPOOL Workshop at OOPSLA'03*, John von Neumann Institute of Computing series, pages 87–107, Anaheim, CA, October 2003.

[285] Jaakko Järvi, Jeremiah Willcock, and Andrew Lumsdaine. Concept-controlled polymorphism. In Frank Pfennig and Yannis Smaragdakis, editors, *Generative Programming and Component Engineering*, volume 2830 of *LNCS*, pages 228–244. Springer Verlag, September 2003.

[286] Jaakko Järvi, Jeremiah Willcock, and Andrew Lumsdaine. Algorithm specialization and concept constrained genericity. In *Concepts: a Linguistic Foundation of Generic Programming at Adobe Tech Summit, San Jose, CA*. Adobe Systems, April 2004.

[287] Jaakko Järvi, Jeremiah Willcock, and Andrew Lumsdaine. Associated types and constraint propagation for mainstream object-oriented generics. Technical Report 2004-12-3, Texas A&M University, Computer Science, December 2004. http://www.cs.tamu.edu/research/tr.

[288] Jaakko Järvi, Jeremiah Willcock, and Andrew Lumsdaine. Associated types and constraint propagation for mainstream object-oriented generics. In *OOPSLA '05: Proceedings of the 20th annual ACM SIGPLAN conference on Object-oriented programming systems languages and applications*, pages 1–19, New York, NY, USA, 2005. ACM Press.

[289] C. Kaiser, T. Hoefler, B. Bierbaum, and T. Bemmerl. Implementation and Analysis of Nonblocking Collective Operations on SCI Networks. In *Proceedings of the 23rd IEEE International Parallel & Distributed Processing Symposium (IPDPS), CAC Workshop*, May 2009.

[290] Prabhanjan Kambadur. Parallelization of generic libraries based on type properties. Presentation at the 7th International Conference on Computational Science, Beijing, China, May 2007.

[291] Prabhanjan Kambadur. Task paralleism in modern languages. Presentation at the IBM T J Watson Research Center, Yorktown Heights, NY, May 2008.

[292] Prabhanjan Kambadur. Extending task parallelism for frequent pattern mining. IBM T J Watson Research Center, August 2009.

[293] Prabhanjan Kambadur. Extending task parallelism for frequent pattern mining. International Conference on Parallel Computing, September 2009.

[294] Prabhanjan Kambadur. An opencl tutorial. Invited talk at IBM T J Watson Research Center, August 2009.

[295] Prabhanjan Kambadur. PFunc: Modern Task Parallelism For Modern High Performance Computing. IBM T J Watson Research Center, July 2009.

[296] Prabhanjan Kambadur, Amol Ghoting, Anshul Gupta, and Andrew Lumsdaine. Extending task parallelism for frequent pattern mining. In *Proceedings of the International Conference on Parallel Computing (ParCO)*, Lyon, France, September 2009.

[297] Prabhanjan Kambadur, Douglas Gregor, and Andrew Lumsdaine. Parallelization of generic libraries based on type properties. In *Proceedings of the 7th International Conference on Computational Science*, LNCS, Beijing, China, May 2007. Springer.

[298] Prabhanjan Kambadur, Douglas Gregor, and Andrew Lumsdaine. Openmp extensions for generic libraries. In *OpenMP in a New Era of Parallelism*, volume 5004 of *Lecture Notes in Computer Science*, pages 123–133. Springer, May 2008.

[299] Prabhanjan Kambadur, Douglas Gregor, Andrew Lumsdaine, and Amey Dharurkar. Modernizing the c++ interface to mpi. In *Proceedings of the 13th European PVM/MPI Users' Group Meeting*, LNCS, pages 266–274, Bonn, Germany, September 2006. Springer.

[300] Prabhanjan Kambadur, Douglas Gregor, Andrew Lumsdaine, and Amey Dharurkar. Modernizing the c++ interface to mpi. Presentation at the 13th European PVM/MPI Users' Group Meeting, Bonn, Germany, September 2006.

[301] Prabhanjan Kambadur, Anshul Gupta, Amol Ghoting, Haim Avron, and Andrew Lumsdaine. Pfunc: Modern task parallelism for modern high performance computing. In *Proceedings of the 2008 ACM/IEEE conference on Supercomputing (SC)*, Portland, Oregon, November 2009.

[302] Prabhanjan Kambadur, Torsten Hoefler, Anshul Gupta, and Andrew Lumsdaine. Demand-driven execution of static direct acyclic graphs using task parallelism. In *International Conference on High Performance Computing (HiPC)*, Kochi, India, December 2009.

[303] Prabhanjan Kambadur, Raj Mutharasan, and Sangli Pranesh Srinivas. Adaptation of a spot fluorometer for lifetime applications. In *Proceedings of Association For Research in Vision and Opthalmology (ARVO)*, Fort Lauderdale, USA, May 2003.

[304] DongInn Kim, Jeffrey M. Squyres, and Andrew Lumsdaine. Revamping the OSCAR database: A flexible approach to cluster configuration data management. In Ilias Kotsireas and Deborah Stacey, editors, *19th International Symposium on High Performance Computing Systems and Applications*, pages 326–332, Guelph, Ontario, Canada, May 2005. IEEE Computer Society.

[305] DongInn Kim, Jeffrey M. Squyres, and Andrew Lumsdaine. The Introduction of the OSCAR Database API (ODA). In *Proceedings of the 20th International Symposium on High-Performance Computing in an Advanced Collaborative Environment (HPCS'06)*, page 39. IEEE Computer Society, May 14-17 2006. Session track: 4th Annual OSCAR Symposium (OSCAR'06).

[306] Lie-Quan Lee. The high performance generic graph component library. Master's thesis, University of Notre Dame, 1999.

[307] Lie-Quan Lee. *Generic Programming for High-Performance Scientific Computing*. PhD thesis, University of Notre Dame, 2002.

[308] Lie-Quan Lee and Andrew Lumsdaine. Generic programming for high performance scientific applications. In *Proceedings of the 2002 Joint ACM Java Grande – ISCOPE Conference*, pages 112–121. ACM Press, 2002.

[309] Lie-Quan Lee and Andrew Lumsdaine. The generic message passing framework. In *Proceedings of the International Parallel and Distributed Processing Symposium (IPDPS)*, page 53, April 2003.

[310] Lie-Quan Lee and Andrew Lumsdaine. Generic programming for high performance scientific applications. *Concurrency and Computation: Practice & Experience*, 17(7-8):941–965, June/July 2003.

[311] Lie-Quan Lee and Andrew Lumsdaine. The iterative template library. *TOMS*, 2003. In preparation.

[312] Lie-Quan Lee, Jeremy G. Siek, and Andrew Lumsdaine. Generic graph algorithms for sparse matrix ordering. In *ISCOPE'99*, Lecture Notes in Computer Science. Springer-Verlag, 1999.

[313] Daniel Lincke, Patrik Jansson, Marcin Zalewski, and Cezar Ionescu. Generic libraries in C++ with concepts from high-level domain descriptions in Haskell. In *IFIP TC 2 Working Conference on Domain-Specific Languages*, pages 236–261, July 2009.

[314] A. Lumsdaine, M. Silveira, and J. White. CMVSIM programmer's guide. Research Laboratory of Electronics, Massachusetts Institute of Technology. Unpublished, 1990.

[315] A. Lumsdaine, M. Silveira, and J. White. CMVSIM users' guide. Research Laboratory of Electronics, Massachusetts Institute of Technology, 1990.

[316] A. Lumsdaine, M. Silveira, and J. White. SIMLAB programmer's guide. Research Laboratory of Electronics, Massachusetts Institute of Technology. Unpublished, 1990.

[317] A. Lumsdaine, J. M. Squyres, and R. L. Stevenson. A cluster-based parallel image processing toolkit. In *SPIE/IS&T Conference on Image and Video Processing, III*, San Jose, CA, USA, February 1995.

[318] A. Lumsdaine, J. White, D. Webber, and A. Sangiovanni-Vincentelli. A band relaxation algorithm for reliable and parallelizable circuit simulation. In *International Conference on Computer Aided-Design*, Santa Clara, California, November 1988.

[319] Andrew Lumsdaine. Krylov-subspace acceleration of time periodic waveform relaxation. In *Colorado Conference on Iterative Methods*, Breckenridge, CO, April 1994.

[320] Andrew Lumsdaine. Generic programming for scientific computing. Presentation at Army Research Lab, Adelphi, Maryland, January 2006.

[321] Andrew Lumsdaine. Lifting sequential graph algorithms for distributed-memory parallel computation. Presentation at Google, Mountain View, California, February 2006.

[322] Andrew Lumsdaine. Openmp extensions for generic libraries. Presentation at IWOMP '08, May 2008.

[323] Andrew Lumsdaine. Radiance photography. Tutorial at 2008 Computer Society Conference on Computer Vision and Pattern Recognition, June 2008.

[324] Andrew Lumsdaine. Radiance photography. Tutorial at 29th annual conference of the European Association for Computer Graphics, April 2008.

[325] Andrew Lumsdaine and Douglas Gregor. The parallel boost graph library. Presentation at Fort Meade, Maryland, January 2006.

[326] Andrew Lumsdaine, Douglas Gregor, Bruce Hendrickson, and Jonathan Berry. Challenges in parallel graph processing. *Parallel Processing Letters*, 17(1):5–20, 2007 2007.

[327] Andrew Lumsdaine and Brian McCandless. Parallel extensions to the matrix template library. In *Proc. 8th SIAM Conference on Parallel Processing for Scientific Computing*. SIAM, 1997.

[328] Andrew Lumsdaine and Brian C. McCandless. The role of abstraction in high performance computing. In *Proceedings, 1997 Internantional Conference on Scientific Computing in Object-Oriented Parallel Computing*, Lecture Notes in Computer Science. Springer-Verlag, 1997.

[329] Andrew Lumsdaine, Mark Reichelt, and Jacob White. Conjugate direction waveform methods for transient two-dimensional simulation of MOS devices. In *International Conference on Computer Aided-Design*, pages 116–119, Santa Clara, California, November 1991.

[330] Andrew Lumsdaine, Mark Reichelt, and Jacob White. Accelerated waveform methods for parallel transient simulation of semiconductor devices. In *International Conference on Computer Aided-Design*, pages 270–274, Santa Clara, California, November 1993.

[331] Andrew Lumsdaine and Mark W. Reichelt. Waveform iterative techniques for device transient simulation on parallel machines. In *Sixth SIAM Conference on Parallel Processing for Scientific Computing*, Norfolk, VA, 1993.

[332] Andrew Lumsdaine and Mark W. Reichelt. Decomposition of space-time domains: Accelerated waveform methods, with application to semiconductor device simulation. In D. E. Keyes, Y. Saad, and D. G. Truhlar, editors, *Domain-Based Parallelism and Problem Decomposition Methods in Computational Science and Engineering*. Society for Industrial and Applied Mathematics, Philadelphia, 1994.

[333] Andrew Lumsdaine, Mark W. Reichelt, Jeffrey M. Squyres, and Jacob K. White. Accelerated waveform methods for parallel transient simulation of semiconductor devices. *IEEE Trans. CAD*, 15(7):716–726, July 1996.

[334] Andrew Lumsdaine, L. Miguel Silveira, and Jacob White. Massively parallel simulation algorithms for grid-based analog signal processors. *IEEE Trans. CAD*, 12(11):1665–1678, November 1993.

[335] Andrew Lumsdaine and Anthony Skjellum, editors. *Proceedings MPI Developers Conference*. IEEE Computer Society Press, 1996.

[336] Andrew Lumsdaine, Jeffrey M. Squyres, and Mark W. Reichelt. Waveform iterative methods for parallel solution of initial value problems. In Andrew Lumsdaine, Ewing Lusk, Donna Reese, and Anthony Skjellum, editors, *Proceedings of the Second Scalable Parallel Libraries Conference*, pages 88–97, Mississippi State, MS, October 1994. IEEE Computer Society Press.

[337] Andrew Lumsdaine and Jacob K. White. Accelerating dynamic iteration methods with application to semiconductor device simulation. In *Proceedings Copper Mountain Conference on Iterative Methods*, Copper Mountain, CO, April 1992. Awarded third place in student paper contest.

[338] Andrew Lumsdaine and Jacob K. White. Accelerating dynamic iteration methods with application to semiconductor device simulation. In W. M. Coughran et al., editors, *Semiconductors*, volume 58-59 of *IMA Volumes in Mathematics and its Applications*. Springer-Verlag, New York, 1994.

[339] Andrew Lumsdaine and Jacob K. White. Accelerating dynamic iteration methods with application to parallel semiconductor device simulation. *Numerical Functional Anal. Optimization*, 16:395–414, 1995.

[340] Andrew Lumsdaine and Deyun Wu. Krylov-subspace acceleration of waveform relaxation. In *Copper Mountain Conference on Iterative Methods*, Copper Mountain, CO, 1996.

[341] Andrew Lumsdaine and Deyun Wu. Spectra and pseudospectra of block Toeplitz matrices. *Linear Algebra and its Applications*, (272/01–03):103–130, 1997.

[342] Andrew Lumsdaine and Deyun Wu. Spectra and pseudospectra of waveform relaxation operators. *SIAM J. Sci. Comput.*, 18(1):286–304, 1997.

[343] Andrew Lumsdaine and Deyun Wu. Krylov-subspace acceleration of waveform relaxation. *SIAM Journal on Numerical Analysis*, 41(1):90–111, 2003.

[344] Benjamin Martin. A comparison of vertex ordering algorithms for large graph visualization. Presentation at Asia-Pacific Symposium on Visualization, February 2007.

[345] Benjamin Martin. Interpreting large visual similarity matrices. Presentation at Asia-Pacific Symposium on Visualization, February 2007.

[346] Timothy I. Mattox. MPI Is Dead? Long Live MPI! Evolving MPI for the Next Generation of Supercomputing. Presentation in the Cisco booth at the ACM/IEEE SC07 Conference, Reno, Nevada, November 2007.

[347] Timothy I. Mattox. Open MPI on Mac OS X: Enabling big science on the Mac. Scientific Development Poster presented during the Apple World Wide Developers Conference (WWDC07), San Francisco, California, June 2007.

[348] Timothy I. Mattox. Research at Indiana University for Reliable Petascale Performance. Presentation at Fault Tolerance workshop at Los Alamos National Lab, Los Alamos, New Mexico, August 2008.

[349] Timothy I. Mattox. Research at Indiana University for Reliable Petascale Performance. Presentation in the Indiana University booth at the IEEE/ACM SC08 Conference, Austin, Texas, November 2008.

[350] Timothy I. Mattox. The Multi-Core Problem: How Message Passing Can Help. Presentation at Apple, Inc., Cupertino, California, July 2008.

[351] B. C. McCandless, J. M. Squyres, and A. Lumsdaine. Object oriented MPI (OOMPI): a class library for the Message Passing Interface. In IEEE, editor, *Proceedings. Second MPI Developer's Conference: Notre Dame, IN, USA, 1–2 July 1996*, pages 87–94, 1109 Spring Street, Suite 300, Silver Spring, MD 20910, USA, 1996. IEEE Computer Society Press.

[352] T. Mehlan, J. Strunk, T. Hoefler, F. Mietke, and W. Rehm. IRS - A portable Interface for Reconfigurable Systems. In *Proceedings of IEEE Inernational Conference on Parallel Computing in Electrical Engineering, PARELEC 2006*, pages 187–191. IEEE Computer Society, 9 2006.

[353] F. Mietke, R. Baumgartl, R. Rex, T. Mehlan, T. Hoefler, and W. Rehm. Analysis of the Memory Registration Process in the Mellanox InfiniBand Software Stack. In *Euro-Par 2006 Parallel Processing*, pages 124–133. Springer-Verlag Berlin, 8 2006.

[354] F. Mietke, M. Steiger, T. Mehlan, and T. Hoefler und W. Rehm. SHIBA Shared Memory Support for InfiniBand MPICH2 Device. In *PARS Mitteilungen 2005*, pages 14–23, 6 2005.

[355] J. P. Morris, N. J. Giordano, J.-T. Cheng, J. Tran, A. Lumsdaine, and L. J. Pyrak-Nolte. Fracture geometry and relative permeabilities: Application to multiphase flow through coal. In *International Coalbed Methane Symposium*, pages 377–388, 1999.

[356] Christopher Mueller. Agile high-performance software development. Presentation at RIDMS-2, January 2007.

[357] Christopher Mueller. Corepy: High-productivity Cell/B.E. programming. Presentation at Georgia Tech STI Cell/B.E. Workshop, June 2007.

[358] Christopher Mueller. *Synthetic Programming: User-Directed Run-Time Code Synthesis for High Performance Computing*. PhD thesis, Indiana University, 2007.

[359] Christopher Mueller, Mehmet Dalkilic, and Andrew Lumsdaine. High-performance direct pairwise comparison of large genomic sequences. In *Proceedings of the Fourth IEEE International Workshop on High Performance Computational Biology (HiCOMB 2005)*, April 2005.

[360] Christopher Mueller, Mehmet Dalkilic, and Andrew Lumsdaine. Implementing data parallel algorithms for bioinformatics. Presentation at SIAM Conference on Computational Science and Engineering, Orlando, FL, February 14 2005.

[361] Christopher Mueller, Mehmet Dalkilic, and Andrew Lumsdaine. High-performance direct pairwise comparison of large genomic sequences. *IEEE Transactions on Parallel and Distributed Systems, Special Issue - High-Performance Computational Biology*, 17(8):764–772, 2006.

[362] Christopher Mueller, Douglas Gregor, and Andrew Lumsdaine. Distributed force-directed graph layout and visualization. In *Eurographics Symposium on Parallel Graphics and Visualization*, May 2006.

[363] Christopher Mueller and Andrew Lumsdaine. Developing reliable complex software systems in a research environment. Presentation at iLight 2005, September 2005.

[364] Christopher Mueller and Andrew Lumsdaine. Exression and loop libraries for high-performance code synthesis. In *Languages and Compilers for Parallel Computing*, November 2006.

[365] Christopher Mueller and Andrew Lumsdaine. Psweep: A lightweight pattern for distributed computational experiments. Presentation at SIAM Conference on Parallel Processing, San Francisco, California, February 2006.

[366] Christopher Mueller and Andrew Lumsdaine. Runtime synthesis of high-performance code from scripting languages. In *Dynamic Language Symposion at OOPSLA 2006*, October 2006.

[367] Christopher Mueller and Andrew Lumsdaine. Synthetic programming. Invited seminar at Intel, August 2006.

[368] Christopher Mueller and Andrew Lumsdaine. Synthetic programming with python. Presentation at SciPy 2006, August 2006.

[369] Christopher Mueller, Benjamin Martin, and Andrew Lumsdaine. Blast on ibm's cell broadband engine. Presentation at SuperComputing 2006, November 2006.

[370] Christopher Mueller, Benjamin Martin, and Andrew Lumsdaine. A comparison of vertex ordering algorithms for large graph visualization. In *Asia-Pacific Symposium on Visualization*, February 2007.

[371] Christopher Mueller, Benjamin Martin, and Andrew Lumsdaine. Interpreting large visual similarity matrices. In *Asia-Pacific Symposium on Visualization*, February 2007.

[372] J. Mueller, T. Schneider, J. Domke, R. Geyer, M. Haesing, T. Hoefler, S. Hoehlig, G. Juckeland, A. Lumsdaine, M. Mueller, , and W. Nagel. Cluster Challenge 2008: Optimizing Cluster Configuration and Applications to Maximize Power Efficiency. Mar. 2009. 2nd best paper at the 10th LCI International Conference on High-Performance Clustered Computing.

[373] John Mugler, Thomas Naughton, Stephen L. Scott, Brian Barrett, Andrew Lumsdaine, Jeffrey M. Squyres, Benot des Ligneris, Francis Giraldeau, and Chokchai Leangsuksun. OSCAR Clusters. In *Proceedings of the Ottawa Linux Symposium (OLS'03)*, Ottawa, Canada, 23-26, 2003.

[374] Ralf Müller, Martin Gaber, and Peter Gottschling. Water release from silicate glass powders. *Glass Science and Technology*, 75:92–97, 2002.

[375] Ralf Müller, Martin Gaber, and Peter Gottschling. Water release and crystallization of glass powders. In *7th Intern. Symposium on Crystallisation in Glasses and Liquids, Sheffield, UK*, July 2003.

[376] Ralf Müller, Martin Gaber, and Peter Gottschling. Volatile concentration and diffusivity determined by vacuum hot extraction. In *20th International Congress on Glass, Kyoto*, 2004.

[377] Ralf Müller, Peter Gottschling, and Martin Gaber. Water concentration and diffusivity in silicates obtained by vacuum extraction. *Glass Science and Technology*, 78:76–89, 2005.

[378] Gustav Munkby, Andreas Priesnitz, Sibylle Schupp, and Marcin Zalewski. Scrap++: Scrap your boilerplate in C++. In *Proc. ACM SIGPLAN Workshop on Generic Programming*, pages 66–75. ACM, 2006.

[379] Richard C. Murphy, Jonathan Berry, William McLendon, Bruce Hendrickson, Douglas Gregor, and Andrew Lumsdaine. DFS: A simple to write yet difficult to execute benchmark. In *Proceedings of the IEEE International Symposium on Workload Characterizations 2006 (IISWC06)*, pages 175–177, October 2006.

[380] Raj Mutharasan, Prabhanjan Kambadur, and Sangli Pranesh Srinivas. Calculation of bioavailability using distributed parameter models of transcorneal transport. In *Proceedings of Association For Research in Vision and Opthalmology (ARVO)*, Fort Lauderdale, USA, May 2003.

[381] Thomas Naughton, Stephen L. Scott, Brian Barret, Jeffrey M. Squyres, Andrew Lumsdaine, and Yung-Chin Fant. The penguin in the pail – OSCAR cluster installation tool. In *The 6th World MultiConference on Systemic, Cybernetics and Informatics (SCI 2002)*, Invited Session of SCI'02, Commodity, High Performance Cluster Computing Technologies and Applications, Orlando, FL, USA, 2002.

[382] Thomas Naughton, Stephen L. Scott, Brian Barrett, Jeffrey M. Squyres, Andrew Lumsdaine, Yung-Chin Gang, and Victor Mashayekhi. Looking inside the OSCAR cluster toolkit. Technical Report In *PowerSolutions Magazine*, chapter HPC Cluster Environment, Dell Computer Corporation, November 2002.

[383] Uwe Naumann and Peter Gottschling. Simulated annealing for optimal pivot selection in jacobian accumulation. In *2nd Symp. on Stochastic Algorithms: Foundations and Applications SAGA03, Hatfield, UK*, September 2003.

[384] D. Nolte, L. J. Pyrak-Nolte, A. Lumsdaine, and J. Tran. Modeling scale dependence of unsaturated flow through single fractures containing correlated aperture distributions. In *EOS - Transactions, American Geophysical Union*, volume 76, page 218, 1995.

[385] T. Hoefler on behalf of the MPI Forum. MPI: A Message-Passing Interface Standard – Working-Draft for Nonblocking Collective Operations. Technical report, MPI Forum, Apr. 2009.

[386] Rolf Rabenseifner, Peter Gottschling, Wolfgang E. Nagel, and Stephan Seidl. Effective performance problem detection of mpi programs on mpp systems: From the global view to the detail. In *Parallel Computing, Delft*, 1999.

[387] Craig E. Rasmussen and Jeffrey M. Squyres. A case for new mpi fortran bindings. In *Proceedings, 12th European PVM/MPI Users' Group Meeting*, Sorrento, Italy, September 2005.

[388] Sriram Sankaran, Jeffrey M. Squyres, Brian Barrett, and Andrew Lumsdaine. Checkpoint-restart support system services interface (SSI) modules for LAM/MPI. Technical Report TR578, Indiana University, Computer Science Department, 2003.

[389] Sriram Sankaran, Jeffrey M. Squyres, Brian Barrett, Andrew Lumsdaine, Jason Duell, Paul Hargrove, and Eric Roman. The LAM/MPI checkpoint/restart framework: System-initiated checkpointing. *International Journal of High Performance Computing Applications*, 19(4):479–493, Winter 2005.

[390] T. Schneider, T. Hoefler, and A. Lumsdaine. ORCS: An Oblivious Routing Congestion Simulator. Technical Report 675, Indiana University, Feb. 2009.

[391] T. Schneider, T. Hoefler, S. Wunderlich, T. Mehlan, and W. Rehm. An optimized ZGEMM implementation for the Cell BE. In *Proceedings of the 9th Workshop on Parallel Systems and Algorithms (PASA)*, 02 2008.

[392] R. R. Schultz, R. L. Stevenson, and A. Lumsdaine. Maximum likelihood parameter estimation for non-Gaussian prior signal models. Submitted to the First IEEE International Conference on Image Processing, Austin, TX, November 13-16, 1994, 1994.

[393] Sibylle Schupp, D. P. Gregor, B. Osman, David R. Musser, Jeremy G. Siek, Lie-Quan Lee, and Andrew Lumsdaine. Concept-based component libraries and optimizing compilers. In *Proceedings IPDPS'02*, 2002.

[394] Sibylle Schupp, Douglas Gregor, and David R. Musser. Algebraic concepts represented in C++. Technical Report TR-00-8, Rensselaer Polytechnic Institute, 2000.

[395] Sibylle Schupp, Douglas Gregor, David R. Musser, and Shin-Ming Liu. Library transformations. In *First IEEE International Workshop on Source Code Analysis and Manipulation (SCAM 2001), Florence, Italy*, pages 109–121. IEEE, November 2001.

[396] Sibylle Schupp, Douglas Gregor, David R. Musser, and Shin-Ming Liu. User-extensible simplification–Type-based optimizer generators. In Reinhard Wilhelm, editor, *Proceedings of the 10th International Conference on Compiler Construction*, number 2027 in LNCS, pages 86–101. Springer-Verlag, Berlin Heidelberg, April 2001.

[397] Sibylle Schupp, Douglas Gregor, David R. Musser, and Shin-Ming Liu. Semantic and behavioral library transformations. *Information and Software Technology*, 44(13):797–810, 2002.

[398] Sibylle Schupp, Douglas Gregor, Brian Osman, David R. Musser, Jeremy Siek, Lie-Quan Lee, and Andrew Lumsdaine. Concept-based component libraries and optimizing compilers. Technical report, RPI Computer Science Department Technical Report 02-02, 2002.

[399] Sibylle Schupp, Mukkai Krishnamoorthy, Marcin Zalewski, and James Kilbride. The "right" level of abstraction — assessing reusable software with formal concept analysis. In G. Angelova, D. Corbett, and U. Priss, editors, *Foundations and Applications of Conceptual Structures — Contributions to ICCS 2002*, pages 74–91. Bulgarian Academy of Sciences, 2002.

[400] Sibylle Schupp, Marcin Zalewski, and Kyle Ross. Rapid performance prediction for library components. In *Proc. 4th. ACM Workshop on Software and Performance*, pages 73–69. ACM Press, 2004.

[401] Galen M. Shipman, Ron Brightwell, Brian Barrett, Jeffrey M. Squyres, and Gil Bloch. Investigations on infiniband: Efficient network buffer utilization at scale. In *Proceedings, Euro PVM/MPI*, Paris, France, October 2007.

[402] Jeremy Siek, Douglas Gregor, Ronald Garcia, Jeremiah Willcock, Jaakko Järvi, and Andrew Lumsdaine. Concepts for c++0x. Technical Report N1758=05-0018, ISO/IEC JTC 1, Information Technology, Subcommittee SC 22, Programming Language C++, January 2005.

[403] Jeremy Siek, Lee-Quan Lee, and Andrew Lumsdaine. *The Boost Graph Library: User Guide and Reference Manual*. Addison-Wesley, 2002.

[404] Jeremy Siek and Andrew Lumsdaine. Mayfly: A pattern for lightweight generic interfaces. In *Pattern Languages of Programs*, July 1999.

[405] Jeremy Siek and Andrew Lumsdaine. Concept checking: Binding parametric polymorphism in C++. In *Proceedings of the First Workshop on C++ Template Programming*, Erfurt, Germany, 2000.

[406] Jeremy Siek and Andrew Lumsdaine. Software engineering for peak performance. *C++ Report*, pages 23–27, May 2000.

[407] Jeremy Siek and Andrew Lumsdaine. Essential language support for generic programming: Formalization part 1. Technical Report 605, Indiana University, December 2004.

[408] Jeremy Siek and Andrew Lumsdaine. Modular generics. In *Concepts: a Linguistic Foundation of Generic Programming*. Adobe Systems, April 2004.

[409] Jeremy Siek and Andrew Lumsdaine. Essential language support for generic programming. In *PLDI '05: Proceedings of the ACM SIGPLAN 2005 conference on Programming language design and implementation*, pages 73–84, New York, NY, USA, June 2005. ACM Press.

[410] Jeremy Siek and Andrew Lumsdaine. Language requirements for large-scale generic libraries. In *GPCE '05: Proceedings of the fourth international conference on Generative Programming and Component Engineering*, September 2005. accepted for publication.

[411] Jeremy G. Siek. A modern framework for portable high performance numerical linear algebra. Master's thesis, University of Notre Dame, 1999.

[412] Jeremy G. Siek. *A Language for Generic Programming*. PhD thesis, Indiana University, August 2005.

[413] Jeremy G. Siek, Lie-Quan Lee, and Andrew Lumsdaine. The generic graph component library. In *Proceedings of the 1999 ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications*, pages 399–414. ACM Press, 1999.

[414] Jeremy G. Siek and Andrew Lumsdaine. The basic linear algebra instruction set: Building blocks for portable high performance. In *SciTools*, 1998.

[415] Jeremy G. Siek and Andrew Lumsdaine. Generic programming for high performance numerical linear algebra. In *SciTools*, 1998.

[416] Jeremy G. Siek and Andrew Lumsdaine. The Matrix Template Library: A generic programming approach to high performance numerical linear algebra. In *International Symposium on Computing in Object-Oriented Parallel Environments*, number 1505 in Lecture Notes in Computer Science, pages 59–70, 1998.

[417] Jeremy G. Siek and Andrew Lumsdaine. The matrix template library: A unifying framework for numerical linear algebra. In *Parallel Object Oriented Scientific Computing*. ECOOP, 1998.

[418] Jeremy G. Siek and Andrew Lumsdaine. A rational approach to portable high performance: The basic linear algebra instruction set (blais) and the fixed algorithm size template (fast) library. In *Parallel Object Oriented Scientific Computing*. ECOOP, 1998.

[419] Jeremy G. Siek and Andrew Lumsdaine. The matrix template library: Generic components for high-performance scientific computing. *Computing in Science and Engineering*, 1(6):70–78, Nov/Dec 1999.

[420] Jeremy G. Siek and Andrew Lumsdaine. *Advances in Software Tools for Scientific Computing*, chapter A Modern Framework for Portable High Performance Numerical Linear Algebra. Springer, 2000.

[421] Jeremy G. Siek and Andrew Lumsdaine. The generic graph component library. *Dr. Dobb's Journal*, September 2000.

[422] Jeremy G. Siek and Andrew Lumsdaine. C++ concept checking. *Dr. Dobb's Journal*, June 2001.

[423] Jeremy G. Siek, Andrew Lumsdaine, and Lie-Quan Lee. Generic programming for high performance numerical linear algebra. In *Proceedings of the SIAM Workshop on Object Oriented Methods for Inter-operable Scientific and Engineering Computing (OO'98)*. SIAM Press, 1998.

[424] L. Miguel Silveira, Andrew Lumsdaine, and Jacob K. White. Parallel simulation algorithms for grid-based analog signal processors. In *International Conference on Computer Aided-Design*, pages 442–445, Santa Clara, California, November 1990.

[425] M. Silveira, A. Lumsdaine, and J. White. SIMLAB users' guide. Research Laboratory of Electronics, Massachusetts Institute of Technology, 1990.

[426] Anthony Skjellum, Diane G. Wooley, Ziyang Lu, Michael Wolf, Purushotham V. Bangalore, Andrew Lumsdaine, Jeffrey M. Squyres, and Brian McCandless. Object-oriented analysis and design of the Message Passing Interface. 13(4):245–292, April 2001.

[427] J. M. Squyres, B. C. McCandless, and A. Lumsdaine. C++ bindings and class libraries for MPI. In *Proceedings, MPI Developer's Conference*, Notre Dame, IN, USA, July 1996.

[428] J. M. Squyres, B. C. McCandless, A Lumsdaine, and R. L. Stevenson. Parallel and distributed algorithms for high-speed image processing. In *Sixth Annual Dual-Use Technologies and Applications Conference*, Syracuse, NY, USA, 1996.

[429] Jeffrey M. Squyres. MPI extensions and applications. Master's thesis, Department of Computer Science and Engineering, University of Notre Dame, Notre Dame, IN, USA, May 1996.

[430] Jeffrey M. Squyres. Definitions and fundamentals – the message passing interface (MPI). *ClusterWorld Magazine, MPI Mechanic Column*, 1(1):26–29, December 2003.

[431] Jeffrey M. Squyres. *A Component Architecture for the Message Passing Interface (MPI): The Systems Service Interface (SSI) of LAM/MPI.* PhD thesis, University of Notre Dame, Notre Dame, IN, April 2004.

[432] Jeffrey M. Squyres. Everything you wanted to know about groups and communicators. *ClusterWorld Magazine, MPI Mechanic Column*, 2(4):32–34, 48, April 2004.

[433] Jeffrey M. Squyres. How to succeed in datatypes without really trying. *ClusterWorld Magazine, MPI Mechanic Column*, 2(5):32–34, May 2004.

[434] Jeffrey M. Squyres. In parallel, everyone hears you scream. *ClusterWorld Magazine, MPI Mechanic Column*, 2(7):36–37, July 2004.

[435] Jeffrey M. Squyres. In parallel, everyone hears you scream II. *ClusterWorld Magazine, MPI Mechanic Column*, 2(8):34–36, August 2004.

[436] Jeffrey M. Squyres. The joys of asynchronous computing. *ClusterWorld Magazine, MPI Mechanic Column*, 2(9):36–38, September 2004.

[437] Jeffrey M. Squyres. More joys of asynchronous communication. *ClusterWorld Magazine, MPI Mechanic Column*, 2(10):34–36, October 2004.

[438] Jeffrey M. Squyres. Mpi debugging – can you hear me now? *ClusterWorld Magazine, MPI Mechanic Column*, 2(12):32–35, December 2004.

[439] Jeffrey M. Squyres. Mpi_reduce: Introducing open mpi. *ClusterWorld Magazine*, 2(11):20–28, November 2004.

[440] Jeffrey M. Squyres. Processes, processors, and MPI, oh my! *ClusterWorld Magazine, MPI Mechanic Column*, 2(1), January 2004.

[441] Jeffrey M. Squyres. What really happens during MPI_INIT. *ClusterWorld Magazine, MPI Mechanic Column*, 2(2):26–28, 46, February 2004.

[442] Jeffrey M. Squyres. Zen and the art of MPI collectives. *ClusterWorld Magazine, MPI Mechanic Column*, 2(3):32–34, March 2004.

[443] Jeffrey M. Squyres. Debugging in parallel (in parallel). *ClusterWorld Magazine, MPI Mechanic Column*, 3(1):34–37, January 2005.

[444] Jeffrey M. Squyres. Doing more with less. *ClusterWorld Magazine, MPI Mechanic Column*, 3(6):32–34,50, June 2005.

[445] Jeffrey M. Squyres. Is your application *spawnworthy*? *ClusterWorld Magazine, MPI Mechanic Column*, 3(3):32–33,46–47, March 2005.

[446] Jeffrey M. Squyres. "progress" is the opposite of "congress". *ClusterWorld Magazine, MPI Mechanic Column*, 3(5):32–35, May 2005.

[447] Jeffrey M. Squyres. The spawn of MPI. *ClusterWorld Magazine, MPI Mechanic Column*, 3(2):40–43, February 2005.

[448] Jeffrey M. Squyres. Why are there so many mpi implementations? *ClusterWorld Magazine, MPI Mechanic Column*, 3(4):32–35, April 2005.

[449] Jeffrey M. Squyres. Shared memory collectives and their consequences. Computer Science Department Seminar, University of Houston, February 2006.

[450] Jeffrey M. Squyres, Brian Barrett, and Andrew Lumsdaine. Boot system services interface (SSI) modules for LAM/MPI. Technical Report TR576, Indiana University, Computer Science Department, 2003.

[451] Jeffrey M. Squyres, Brian Barrett, and Andrew Lumsdaine. MPI collective operations system services interface (SSI) modules for LAM/MPI. Technical Report TR577, Indiana University, Computer Science Department, 2003.

[452] Jeffrey M. Squyres, Brian Barrett, and Andrew Lumsdaine. Request progression interface (RPI) system services interface (SSI) modules for LAM/MPI. Technical Report TR579, Indiana University, Computer Science Department, 2003.

[453] Jeffrey M. Squyres, Brian Barrett, and Andrew Lumsdaine. The system services interface (SSI) to LAM/MPI. Technical Report TR575, Indiana University, Computer Science Department, 2003.

[454] Jeffrey M. Squyres, Brian W. Barrett, George Bosilca, Richard L. Graham, Galen M. Shipman, and Timothy S. Woodall. Open mpi developer's workshop, April 2006. San Jose, CA, USA.

[455] Jeffrey M. Squyres and Andrew Lumsdaine. Parallel numerical methods for large scale initial-value problems. In *Argonne Undergraduate Research Symposium*, Argonne, IL, November 1993.

[456] Jeffrey M. Squyres and Andrew Lumsdaine. A component architecture for LAM/MPI. In *Proceedings, Euro PVM/MPI*, October 2003.

[457] Jeffrey M. Squyres and Andrew Lumsdaine. The component architecture of LAM/MPI (poster). In *Proceedings ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming*, San Diego, CA, USA, June 2003.

[458] Jeffrey M. Squyres and Andrew Lumsdaine. The component architecture of open MPI: Enabling third-party collective algorithms. In Vladimir Getov and Thilo Kielmann, editors, *Proceedings, 18th ACM International*

Conference on Supercomputing, Workshop on Component Models and Systems for Grid Applications, pages 167–185, St. Malo, France, July 2004. Springer.

[459] Jeffrey M. Squyres, Andrew Lumsdaine, William L. George, John G. Hagedorn, and Judith E. Devaney. The interoperable message passing interface (IMPI) extensions to LAM/MPI. In *Proceedings, MPI Developer's Conference*, Cornell, NY, USA, 2000.

[460] Jeffrey M. Squyres, Andrew Lumsdaine, and Robert L. Stevenson. A parallel image processing toolkit using mpi. In *Proceedings of MPI Developers Conference*, 1995.

[461] Jeffrey M. Squyres, Andrew Lumsdaine, and Robert L. Stevenson. A toolkit for parallel image processing. In *SPIE Annual Meeting*, San Diego, 1998.

[462] Jeffrey M. Squyres, Brian C. McCandless, and Andrew Lumsdaine. Object oriented mpi: A class library for the message passing interface. In *Parallel Object-Oriented Methods and Applications (POOMA '96)*, Santa Fe, 1996.

[463] Jeffrey M. Squyres, Brian C. McCandless, and Andrew Lumsdaine. Object oriented MPI reference. Technical Report TR 96-12, Department of Computer Science and Engineering, University of Notre Dame, Notre Dame, IN, USA, 1996.

[464] Jeffrey M. Squyres, Bill Saphir, and Andrew Lumsdaine. The design and evolution of the mpi-2 c++ interface. In *Proceedings, 1997 International Conference on Scientific Computing in Object-Oriented Parallel Computing*, Lecture Notes in Computer Science. Springer-Verlag, 1997.

[465] Jeffrey M. Squyres, Timothy Woodall, and Shawn Hansen. Open mpi and openib: Past, present, and future. Presentation at the OpenIB Workshop, Sonoma, CA, February 2006.

[466] M. H. Swat, S. D. Hester, A. I. Balter, R. W. Heiland, B. L. Zaitlen, and J. A. Glazier. Multicell simulations of development and disease using the compucell3d simulation environment. In I. V. Maly, editor, *Systems Biology*, volume 500 of *Methods in Molecular Biology*, pages 361–428. Humana Press, Clifton, N.J., 2009.

[467] Todd L. Veldhuizen. *Active Libraries and Universal Languages*. PhD thesis, Indiana University Computer Science, May 2004.

[468] Todd L. Veldhuizen. Guaranteed optimization for domain-specific programming. In *Domain-Specific Program Generation*, volume 3016 of *Lecture Notes in Computer Science*, pages 306–324. Springer-Verlag, 2004.

[469] Todd L. Veldhuizen. Software libraries and their reuse: Entropy, Kolmogorov complexity, and Zipf's law. 2005. OOPSLA 2005 Workshop on Library-Centric Software Design (LCSD'05), San Diego, October 16.

[470] Todd L. Veldhuizen. Tradeoffs in metaprogramming. In *ACM SIGPLAN Workshop on Partial Evaluation and Semantics-Based Program Manipulation*, Charleston, South Carolina, January 2006.

[471] Todd L. Veldhuizen and Andrew Lumsdaine. Guaranteed optimization: Proving nullspace properties of compilers. In *Proceedings of the 2002 Static Analysis Symposium (SAS'02)*, volume 2477 of *Lecture Notes in Computer Science*, pages 263–277. Springer-Verlag, 2002.

[472] Todd L. Veldhuizen and Andrew Lumsdaine. Guaranteed optimization: Proving nullspace properties of compilers. Technical Report TR564, Indiana University Computer Science, 2002.

[473] Todd L. Veldhuizen and Jeremy Siek. Combining optimizations, combining theories. Technical Report 582, Indiana University, May 2003.

[474] Irina Virtanen, Eeva Ekholm, Tero Aittokallio, Jarno Tähtinen, Jussi Salmi, Jaakko Järvi, Päivi Polo-Kantola, and Olli Polo. State of vigilance, oestrogen replacement therapy, and libid profile as modifiers of nocturnal movement-induced heart rate responces. *Clinical Physiology and Functional Imaging*, 23(5):293–299, 2003.

[475] Jeremiah Willcock. Concept representation for generic programming. Master's thesis, University of Notre Dame du Lac, July 2002.

[476] Jeremiah Willcock, Jaakko Järvi, Andrew Lumsdaine, and David Musser. A formalization of concepts for generic programming. In *Concepts: a Linguistic Foundation of Generic Programming at Adobe Tech Summit*. Adobe Systems, April 2004.

[477] Jeremiah Willcock and Andrew Lumsdaine. Accelerating sparse matrix computations via data compression. In *International Conference on Supercomputing*, pages 307–316, June 2006.

[478] Jeremiah Willcock, Andrew Lumsdaine, and Daniel Quinlan. Tabled execution in Scheme. In *Workshop on Scheme and Functional Programming*, September 2008.

[479] Jeremiah Willcock, Andrew Lumsdaine, and Arch Robison. Using MPI with C# and the Common Language Infrastructure. Technical Report 570, Indiana University Computer Science Department, October 2002. http://www.cs.indiana.edu/cgi-bin/techreports/TRNNN.cgi?trnum=TR570.

[480] Jeremiah Willcock, Andrew Lumsdaine, and Arch Robison. Using MPI with C# and the Common Language Infrastructure. In *Proceedings of the Joint ACM Java Grande - ISCOPE 2002 Conference*, page 238. ACM SIGPLAN, ACM Press, November 2002. Poster abstract.

[481] Jeremiah Willcock, Andrew Lumsdaine, and Arch Robison. Using MPI with C# and the Common Language Infrastructure. *Concurrency and Computation: Practice & Experience*, 17(7-8):895–917, June/July 2005.

[482] Jeremiah Willcock, Jeremy Siek, and Andrew Lumsdaine. Caramel: A concept representation system for generic programming. In *Second Workshop on C++ Template Programming*, Tampa, Florida, October 2001.

[483] Jeremiah J. Willcock. *A Language for Specifying Compiler Optimizations for Generic Software*. PhD thesis, Indiana University, January 2008.

[484] Jeremiah James Willcock, Andrew Lumsdaine, and Daniel J. Quinlan. Reusable, generic program analyses and transformations. In *GPCE*, pages 5–14, New York, NY, USA, 2009. ACM.

[485] T.S. Woodall, R.L. Graham, R.H. Castain, D.J. Daniel, M.W. Sukalski, G.E. Fagg, E. Gabriel, G. Bosilca, T. Angskun, J.J. Dongarra, J.M. Squyres, V. Sahay, P. Kambadur, B. Barrett, and A. Lumsdaine. Open MPI's TEG point-to-point communications methodology: Comparison to existing implementations. In *Proceedings, 11th European PVM/MPI Users' Group Meeting*, pages 105–111, Budapest, Hungary, September 2004.

[486] T.S. Woodall, R.L. Graham, R.H. Castain, D.J. Daniel, M.W. Sukalski, G.E. Fagg, E. Gabriel, G. Bosilca, T. Angskun, J.J. Dongarra, J.M. Squyres, V. Sahay, P. Kambadur, B. Barrett, and A. Lumsdaine. TEG: A high-performance, scalable, multi-network point-to-point communications methodology. In *Proceedings, 11th European PVM/MPI Users' Group Meeting*, pages 303–310, Budapest, Hungary, September 2004.

[487] Ben Zaitlen and Randy Heiland. Spatial geometry in compucell3d. Presentation at the CellML-SBGN-SBO-BioPAX-MIASE Workshop, Auckland, New Zealand, April 2009.

[488] M. Zalewski and S. Schupp. A polymorphic radix-n framework for fast fourier transforms. In M. Broy and A. V Zamulin, editors, *Proc. 5th International Andrei Ershov Memorial Conference on Perspectives of Systems Informatics*, volume 2890 of *LNCS*, page 3037. Springer, July 2003.

[489] Marcin Zalewski. *Generic Programming with Concepts*. PhD thesis, Chalmers University, November 2008.

[490] Marcin Zalewski. A semantic definition of separate type checking in C++ with concepts. Abstract syntax and complete semantic definition. Technical Report 2008:12, Department of Computer Science and Engineering, Chalmers University, 2008.

[491] Marcin Zalewski, Assad Jarrahian, Jim Percent, Gary Moore, and Frank Vardaro. Circular buffer implementation of SGI's deque container, 2000. Undergraduate project report.

[492] Marcin Zalewski, Andreas Priesnitz, Cezar Ionescu, Nicola Botta, and Sibylle Schupp. Multi-language library development: From Haskell type classes to C++ concepts. In Jrg Striegnitz, editor, *Proc. 6th International Workshop on Multiparadigm Programming with Object-Oriented Languages*, July 2007.

[493] Marcin Zalewski and Sibylle Schupp. Algorithms that share: Polymorphic FFTs. Technical Report 02-07, Rensselaer Polytechnic Institute, 2002.

[494] Marcin Zalewski and Sibylle Schupp. Changing iterators with confidence. A case study of change impact analysis applied to conceptual specifications. In David Musser and Jeremy Siek, editors, *Proc. 1st Int. Workshop on Library-Centric Software Design*, 2005.

[495] Marcin Zalewski and Sibylle Schupp. Change impact analysis for generic libraries. In *Proc. of 22nd IEEE Int. Conf. on Software Maintenance*, pages 35–44, September 2006.

[496] Marcin Zalewski and Sibylle Schupp. C++ concepts as institutions: A specification view on concepts. In *Proc. Symposium on Library-Centric Software Design*, pages 76–87. ACM, 2007.

[497] Marcin Zalewski and Sibylle Schupp. A semantic definition of separate type checking in C++ with concepts. *Journal of Object Technology*, 8(5):105–132, 2009. Extended version available in Zalewski's PhD thesis.