

- abadi.96.theory [1] Martín Abadi and Luca Cardelli. *A Theory of Objects*. Monographs in Computer Science. Springer-Verlag, 1996.
- aigner.96.ecoop [2] Gerald Aigner and Urs Hölzle. Eliminating virtual function calls in C++ programs. In *In the Proceedings of the 10th European Conference on Object-Oriented Programming (ECOOP)*, volume 1098 of *Lecture Notes in Computer Science*, pages 142–167. Springer-Verlag, 1996.
- alexandrescu.01.book [3] Andrei Alexandrescu. *Modern C++ Design: Generic Programming and Design Patterns Applied*. Addison-Wesley, 2001.
- bacon.96.oopsla [4] David F. Bacon and Peter F. Sweeney. Fast static analysis of C++ virtual function calls. In *In the Proceedings of the ACM Conference on Object-Oriented Programming Systems, Languages and Applications (OOPSLA)*, pages 324–341, 1996.
- barton.94.secpp [5] John Barton and Lee Nackman. *Scientific and engineering C++*. Addison-Wesley, 1994.
- baumgartner.97.toplas [6] Gerald Baumgartner and Vincent F. Russo. Implementing signatures for C++. *ACM Transactions on Programming Languages and Systems*, 19(1):153–187, January 1997.
- boost.net [7] Boost libraries.
- bracha.98.oopsla [8] Gilad Bracha, Martin Odersky, David Stoutamire, and Philip Wadler. Making the future safe for the past: Adding genericity to the Java programming language. In Craig Chambers, editor, *In the Proceedings of the ACM Conference on Object-Oriented Programming Systems, Languages and Applications (OOPSLA)*, pages 183–200, Vancouver, BC, 1998.
- bruce.03.toplas [9] Kim B. Bruce, Adrian Fiech, Angela Schuett, and Robert van Gent. PolyTOIL: A type-safe polymorphic object-oriented language. *ACM Transactions on Programming Languages and Systems (ToPLAS)*, 25(2):225–290, March 2003.
- bruce.95.ecoop [10] Kim B. Bruce, Angela Schuett, , and Robert van Gent. PolyTOIL: A type-safe polymorphic object-oriented language. In *In the Proceedings of the 9th European Conference on Object-Oriented Programming (ECOOP)*, volume 952, pages 27–51, Århus, Denmark, August 1995. Springer-Verlag.
- bruce.96.tapos [11] Kim Bruce, Luca Cardelli, Giuseppe Castagna, The Hopkins Objects Group, Gary T. Leavens, and Benjamin Pierce. On binary methods. *Theory and Practice of Object Systems*, 1(3):221–242, 1996.
- bruce.97.ecoop [12] Kim B. Bruce, Leaf Petersen, and Adrian Fiech. Subtyping is not a good ”match” for object-oriented languages. In *In the Proceedings of the 11th European Conference on Object-Oriented Programming (ECOOP)*, volume 1241 of *Lecture Notes in Computer Science*, pages 104–127, Jyväskylä, Finland, 1997. Springer-Verlag.
- bruce.98.ecoop [13] Kim B. Bruce, Martin Odersky, and Philip Wadler. A statically safe alternative to virtual types. In *In the Proceedings of the 12th European Conference on Object-Oriented Programming (ECOOP)*, volume 1445 of *Lecture Notes in Computer Science*, pages 523–549, Brussels, Belgium, July 1998. Springer-Verlag.
- bruce96typing [14] Kim B. Bruce. Typing in object-oriented languages: Achieving expressibility and safety, 1996.
- canning.89.fpcap [15] Peter S. Canning, William R. Cook, Walter L. Hill, John C. Mitchell, and Walter G. Olthoff. F-bounded polymorphism for object-oriented programming. In *In the Proceedings of the 4th International Conference on Functional Programming Languages and Computer Architecture (FPCA ’89)*, pages 73–280, London, UK, September 1989. ACM.
- cardelli.85.cs [16] Luca Cardelli and Peter Wegner. On understanding types, data abstraction, and polymorphism. *Computing Surveys*, 17(4):471–522, December 1985.
- chambers.92.tr [17] Craig Chambers. The design and implementation of the SELF compiler. Technical Report STAN-CS-92-1420, Department of Computer Science, Stanford University, March 1992.
- chambers.96.tr [18] Craig Chambers, Jeffrey Dean, and David Grove. Wholeprogram optimization of object-oriented languages. Technical Report UW-CSE-96-06-02, University of Washington, Department of Computer Science and Engineering, June 1996.

- `coplien.96.cppgems` [19] James Coplien. *Curiously Recurring Template Pattern*. In [20].
- `cppgems` [20] Stanley B. Lippman, editor. *C++ Gems*. Cambridge Press University & Sigs Books, 1998.
- `czarnecki.00.book` [21] Krzysztof Czarnecki and Ulrich Eisenecker. *Generative programming: Methods, Tools, and Applications*. Addison-Wesley, 2000.
- `day.95.oopsla` [22] Mark Day, Robert Gruber, Barbara Liskov, and Andrew C. Meyers. Subtypes vs. where clauses: Constraining parametric polymorphism. In *In the Proceedings of the ACM Conference on Object-Oriented Programming Systems, Languages and Applications (OOPSLA)*, SIGPLAN Notices volume 30 number 10, pages 156–168, 1995.
- `detlefs.99.ecoop` [23] David Detlefs and Ole Agesen. Inlining of virtual methods. In *In the Proceedings of the 13th European Conference on Object-Oriented Programming (ECOOP)*, Lisbon, Portugal, June 1999.
- `driesen.96.oopsla` [24] Karel Driesen and Urs Hölzle. The direct cost of virtual function calls in C++. In *In the Proceedings of the ACM Conference on Object-Oriented Programming Systems, Languages and Applications (OOPSLA)*, SIGPLAN Notices 31(10), pages 306–323, 1996.
- `duret.00.gcseyrw` [25] Alexandre Duret-Lutz. Olena: a component-based platform for image processing, mixing generic, generative and OO programming. In *In the Proceedings of 2nd International Symposium on Generative and Component-Based Software Engineering (GCSE 2000)*, pages 653–659, Erfurt, Germany, October 2000. Young Researchers Workshop (published in "Net.ObjectDays2000").
- `duret.01.adae` [26] Alexandre Duret-Lutz. Expression templates in Ada. In *In the Proceedings of the 6th International Conference on Reliable Software Technologies, Leuven, Belgium, May 2001 (Ada-Europe)*, volume 2043 of *Lecture Notes in Computer Science*, pages 191–202. Springer-Verlag, 2001.
- `duret.01.coots` [27] Alexandre Duret-Lutz, Thierry Géraud, and Akim Demaille. Design patterns for generic programming in C++. In *In the Proceedings of the 6th USENIX Conference on Object-Oriented Technologies and Systems (COOTS)*, pages 189–202, San Antonio, Texas, USA, January 2001. USENIX Association.
- `fabri.00.spe` [28] Andreas Fabri, Geert-Jan Giezeman, Lutz Kettner, Stefan Schirra, and Sven Schönherr. On the design of CGAL a computational geometry algorithms library. *Software - Practice and Experience*, 30(11):1167–1202, 2000.
- `furnish.97.cip` [29] Geoffrey Furnish. Disambiguated glomtable expression templates. *Computers in Physics*, 11(3):263–269, 1997.
- `geraud.00.eplop` [30] Thierry Géraud and Alexandre Duret-Lutz. Design patterns for generic programming. In M. Devos and A. Rüping, editors, *In the Proceedings of the 5th European Conference on Pattern Languages of Programs (EuroPLoP'2000)*. UVK, Univ. Verlag, Konstanz, July 2000.
- `geraud.01.ai` [31] Thierry Géraud, Yoann Fabre, and Alexandre Duret-Lutz. Applying generic programming to image processing. In M.H. Hamsa, editor, *In the Proceedings of the IASTED International Conference on Applied Informatics – Symposium Advances in Computer Applications*, pages 577–581, Innsbruck, Austria, February 2001. ACTA Press.
- `gp.98.proc` [32] M. Jazayeri, R. Loos, and D. Musser, editors. *Generic Programming: International Seminar, Dagstuhl Castle, Germany, 1998, Selected Papers*, volume 1766 of *Lecture Notes in Computer Science*. Springer-Verlag, 2000.
- `haney.96.cip` [33] Scott W. Haney. Beating the abstraction penalty in C++ using expression templates. *Computers in Physics*, 10(6):552–557, 1996.
- `iso.98.cpp` [34] International standard: Programming language – C++. ISO/IEC 14882:1998(E), 1998.
- `jarvi.01.tmpw` [35] Jaakko Järvi and Gary Powell. The lambda library: Lambda abstraction in C++. In *In the Proceedings of the 2nd Workshop on Template Programming (in conjunction with OOPSLA)*, Tampa Bay, Florida, USA, October 2001.
- `jazayeri.95.esec` [36] Mehdi Jazayeri. Component programming: a fresh look at software components. In *In the Proceedings of the 5th European Software Engineering Conference (ESEC)*, pages 457–478, September 1995.
- `langer.00.oopsla` [37] Angelika Langer. Implementing design patterns using C++ templates. Tutorial at the ACM Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA), October 2000.

- `maes.03.dpcool` [38] Francis Maes. Program templates: Expression templates applied to program evaluation. In Jörg Striegnitz and Kei Davis, editors, *In the Proceedings of the Workshop on Declarative Programming in the Context of Object-Oriented Languages (DP-COOL; in conjunction with PLI)*, number FZJ-ZAM-IB-2003-10 in John von Neumann Institute for Computing (NIC), Uppsala, Sweden, August 2003.
- `mcmamara.00.tmpw` [39] Brian McNamara and Yannis Smaragdakis. Static interfaces in C++. In *First Workshop on C++ Template Programming*, Erfurt, Germany, October 2000.
- `mpl` [40] A. Gurtovoy and D. Abrahams. The boost C++ metaprogramming library, March 2002.
- `myers.95.cppr` [41] Nathan C. Myers. Traits: a new and useful template technique. *C++ Report*, 7(5):32–35, June 1995.
- `olena.net` [42] Olena image processing library.
- `oonumerics.net` [43] Scientific computing in object-oriented languages.
- `palsberg.94.scp` [44] Jens Palsberg and Michael I. Schwartzbach. Static typing for object-oriented programming. *Science of Computer Programming*, 23(1):19–53, 1994.
- `regisgianas.03.poosc` [45] Yann Régis-Gianas and Raphaël Poss. On orthogonal specialization in C++: Dealing with efficiency and algebraic abstraction in Vaucanson. In Jörg Striegnitz and Kei Davis, editors, *In the Proceedings of the Parallel/High-performance Object-Oriented Scientific Computing (POOSC; in conjunction with ECOOP)*, number FZJ-ZAM-IB-2003-09 in John von Neumann Institute for Computing (NIC), Darmstadt, Germany, July 2003.
- `schultz.01.lnsc` [46] Ulrik P. Schultz. Partial evaluation for class-based object-oriented languages. In *Program as Data Objects: International Conference on the Theory and Application of Cryptographic Techniques, Innsbruck, Austria, May 2001, Proceedings*, volume 2053 of *Lecture Notes in Computer Science*, pages 173–198. Springer-Verlag, 2001.
- `seidewitz.94.oopsla` [47] Ed Seidewitz. Genericity versus inheritance reconsidered: Self-reference using generics. In *In the Proceedings of the 9th Conference on Object Oriented Programming Systems Languages and Applications (OOPSLA)*, pages 153–163, Portland, OR, USA, October 1994.
- `siek.00.tmpw` [48] Jeremy Siek and Andrew Lumsdaine. Concept checking: Binding parametric polymorphism in C++. In *Proceedings of the First Workshop on C++ Template Programming*, Erfurt, Germany, October 2000.
- `smaragdakis.00.gcse` [49] Yannis Smaragdakis and Don Batory. Mixin-based programming in C++. In *In the Proceedings of the 2nd International Conference on Generative and Component-based Software Engineering (GCSE)*, pages 464–478. transIT Verlag, Germany, October 2000.
- `smaragdakis.02.spe` [50] Yannis Smaragdakis and Brian McNamara. FC++: Functional tools for object-oriented tasks. *Software - Practice and Experience*, 32(10):1015–1033, August 2002.
- `smith.03.accu` [51] Julian Smith. C++ & multi-methods. *ACCU spring 2003 conference*, 2003.
- `stepanov.00.book` [52] Alexander Stepanov, Meng Lee, and David Musser. *The C++ Standard Template Library*. Prentice-Hall, 2000.
- `striegnitz.00.tmpw` [53] Jörg Striegnitz and Stephen A. Smith. An expression template aware lambda function. In *In the Proceedings of the 1st Workshop on Template Programming*, Erfurt, Germany, October 2000.
- `stroustrup.88.cpp` [54] Bjarne Stroustrup. Parameterized types for C++. In *In the Proceedings of the USENIX C++ Conference*, Denver, USA, October 1988.
- `stroustrup.94.dne` [55] Bjarne Stroustrup. *The Design and Evolution of C++*. Addison-Wesley, 1994.
- `stroustrup.95.oopsla` [56] Bjarne Stroustrup. Why C++ isn't just an object-oriented programming language. In *In the Proceedings of the ACM Conference on Object-Oriented Programming Systems, Languages and Applications (OOPSLA)*, October 1995.
- `surazhsky02` [57] Vitaly Surazhsky and Joseph Y. Gil. Type-safe covariance in C++, 2002. Unpublished.
- `thorup.99.ecoop` [58] Kresten Krab Thorup and Mads Torgersen. Unifying genericity: Combining the benefits of virtual types and parameterized classes. In Rachid Guerraoui, editor, *In the Proceedings of the 13th European Conference on Object-Oriented Programming (ECOOP)*, volume 1628 of *Lecture Notes in Computer Science*, pages 186–204, Lisbon, Portugal, June 1999. Springer-Verlag.

- `torgersen.98.fool` [59] Mads Torgersen. Virtual types are statically safe. In *In the Proceedings of the 5th Workshop on Foundations of Object-Oriented Languages (FOOL)*, San Diego, CA, January 1998.
- `vandevoorde.03.book` [60] David Vandevoorde and Nicolai M. Josuttis. *C++ Templates: The Complete Guide*. Addison-Wesley, 2003.
- `veldhuizen.00.tmpw` [61] Todd L. Veldhuizen. Five compilation models for C++ templates. In *Proceedings of the 1st Workshop on C++ Template Programming*, Erfurt, Germany, October 2000.
- `veldhuizen.02.sas` [62] Todd L. Veldhuizen and Andrew Lumsdaine. Guaranteed optimization: Proving nullspace properties of compilers. In *Static Analysis, 9th International Symposium, SAS 2002, Madrid, Spain, September 17-20, 2002, Proceedings*, volume 2477 of *Lecture Notes in Computer Science*, pages 263–277. Springer-Verlag, 2002.
- `veldhuizen.98.cppgems` [63] Todd Veldhuizen. *Expression Templates*, pages 475–487. In [20].
- `veldhuizen.99.tech` [64] Todd L. Veldhuizen. Techniques for scientific C++, August 1999.
- `zendra.97.oopsla` [65] Olivier Zendra, Dominique Colnet, and Suzanne Collin. Efficient Dynamic Dispatch without Virtual Function Tables. The SmallEiffel Compiler. In *In the Proceedings of the 12th ACM Conference on Object-Oriented Programming Systems, Languages and Applications (OOPSLA)*, volume 32 of *Issue 10*, pages 125–141, Atlanta, GA, USA, October 1997.