

# Bibliography

- [AAC<sup>+</sup>05] Chris Allan, Pavel Avgustinov, Aske Simon Christensen, Bruno Dufour, Christopher Goard, Laurie Hendren, Sascha Kuzins, Jennifer Lhoták, Ondrej Lhoták, Oege de Moor, Damien Sereni, Ganesh Sittampalam, Julian Tibble, and Clark Verbrugge. abc: the AspectBench compiler for AspectJ – a workbench for aspect-oriented programming language and compilers research. In *OOPSLA '05: Companion to the 20th annual ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications*, pages 88–89, New York, NY, USA, 2005. ACM Press.
- [AC98] Ellen Agerbo and Aino Cornils. How to preserve the benefits of design patterns. In *OOPSLA '98: Proceedings of the 13th ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications*, pages 134–143. ACM Press, 1998.
- [AG05] David Abrahams and Aleksey Gurtovoy. *C++ Template Meta-Programming*. Addison-Wesley, Boston, MA, USA, 2005.
- [AL00] Uwe Assmann and Andreas Ludwig. Aspect weaving with graph rewriting. In *GCSE '99: Proceedings of the First International Symposium on Generative and Component-Based Software Engineering*, pages 24–36, London, UK, 2000. Springer-Verlag.
- [ANMM06] Chris Andreae, James Noble, Shane Markstrum, and Todd Millstein. A framework for implementing pluggable type systems. In *Proceedings of OOPSLA'06: Conference on Object-Oriented Programming, Systems, Languages, and Applications*, New York, NY, USA, 2006. ACM Press.
- [Art88] Lowell Jay Arthur. *Software evolution: the software maintenance challenge*. Wiley-Interscience, New York, NY, USA, 1988.
- [Ass03] U. Assmann. *Invasive Software Composition*. Springer-Verlag New York, Inc., 2003.
- [AT01] Paul Anderson and Tim Teitelbaum. Software inspection using CodeSurfer. In *Proceedings of WISE'01 (Intl Workshop on Inspection in Software Engineering)*, 2001.
- [Aug93] Lex Augustueijn. *Functional Programming, Program Transformations and Compiler Construction*. PhD thesis, Eindhoven Technical University, The Netherlands, October 1993.

- [Bat05] Don S. Batory. Feature models, grammars, and propositional formulas. In J. Henk Obbink and Klaus Pohl, editors, *SPLC*, volume 3714 of *Lecture Notes in Computer Science*, pages 7–20. Springer, 2005.
- [BBK<sup>+</sup>05] Anya Helene Bagge, Martin Bravenboer, Karl Trygve Kalleberg, Koen Mulwijk, and Eelco Visser. Adaptive code reuse by aspects, cloning and renaming. Technical Report UU-CS-2005-031, Utrecht University, 2005.
- [BDD06] Alexandre Borghi, Valentin David, and Akim Demaille. C-transformers: A framework to write C program transformations. *ACM Crossroads*, 12(3), April 2006.
- [BDHK06] Anya Bagge, Valentin David, Magne Haverdaen, and Karl Trygve Kalleberg. Stayin’ alert: Moulding failure and exceptions to your needs. In *Proceedings of the 5th International Conference on Generative Programming and Component Engineering (GPCE’06)*, Portland, Oregon, October 2006. ACM Press.
- [Bec98] Kent Beck. Extreme programming: A humanistic discipline of software development. In *FASE*, pages 1–6, 1998.
- [Bec02] Kent Beck. *Test-Driven Development: By Example*. Addison-Wesley, 2002.
- [BEG<sup>+</sup>87] H P Barendregt, M C J D Eekelen, J R W Glauert, J R Kennaway, M J Plasmeijer, and M R Sleep. Term graph rewriting. In *Volume II: Parallel Languages on PARLE: Parallel Architectures and Languages Europe*, pages 141–158, London, UK, 1987. Springer-Verlag.
- [Ber05] Clara Bertolissi. *The graph rewriting calculus: properties and expressive capabilities*. Thèse de doctorat, Institut National Polytechnique Lorraine - INPL, October 2005.
- [BG] Kent Beck and Erich Gamma. JUnit – Java Unit testing. <http://www.junit.org> and <http://junit.sourceforge.net/> per 2007-03-15.
- [BK06] Anya Helene Bagge and Karl Trygve Kalleberg. DSAL = library+notation: Program transformation for domain-specific aspect languages. In *Proceedings of the Domain-Specific Aspect Languages Workshop*, October 2006.

- [BKHV03] Otto Skrove Bagge, Karl Trygve Kalleberg, Magne Haveraaen, and Eelco Visser. Design of the CodeBoost transformation system for domain-specific optimisation of C++ programs. In Dave Binkley and Paolo Tonella, editors, *Third International Workshop on Source Code Analysis and Manipulation (SCAM 2003)*, pages 65–75, Amsterdam, The Netherlands, September 2003. IEEE Computer Society Press.
- [BKK<sup>+</sup>04] P. Borovansky, C. Kirchner, H. Kirchner, P.-E. Moreau, and M. Vittek. ELAN: A logical framework based on computational systems. *Electronic Notes in Theoretical Computer Science*, 4(1):35–50(15), Jan 2004.
- [BKVV05] Martin Bravenboer, Karl Trygve Kalleberg, Rob Vermaas, and Eelco Visser. *Stratego/XT Tutorial, Examples, and Reference Manual*. Department of Information and Computing Sciences, Universiteit Utrecht, Utrecht, The Netherlands, August 2005. (Draft).
- [BKVV06] Martin Bravenboer, Karl Trygve Kalleberg, Rob Vermaas, and Eelco Visser. Stratego/XT 0.16: Components for transformation systems. In Frank Tip and John Hatcliff, editors, *PEPM'06: Workshop on Partial Evaluation and Program Manipulation*. ACM Press, January 2006.
- [BKVV07] Martin Bravenboer, Karl Trygve Kalleberg, Rob Vermaas, and Eelco Visser. Stratego/XT 0.16. a language and toolset for program transformation. *Science of Computer Programming*, 2007. Accepted for publication.
- [BLS98] D. Batory, B. Lofaso, and Y. Smaragdakis. JTS: Tools for implementing domain-specific languages. In *ICSR '98: Proceedings of the 5th International Conference on Software Reuse*, page 143, Washington, DC, USA, 1998. IEEE Computer Society.
- [BM91] Jean-Pierre Banâtre and Daniel Le Métayer. Introduction to Gamma. In *Research Directions in High-Level Parallel Programming Languages*, pages 197–202, 1991.
- [BM93] Jean-Pierre Banâtre and Daniel Le Métayer. Programming by multi-set transformation. *Commun. ACM*, 36(1):98–111, 1993.
- [BMV02] Johan Brichau, Kim Mens, and Kris De Volder. Building composable aspect-specific languages with logic metaprogramming. In *GPCE '02: The ACM SIGPLAN/SIGSOFT Conference on Generative*

- Programming and Component Engineering*, pages 110–127, London, UK, 2002. Springer-Verlag.
- [Bos98] Jan Bosch. Design patterns as language constructs. *Journal of Object-Oriented Programming*, 11(2):18–32, 1998.
- [BP] Jean Bovet and Terence Parr. ANTLRWorks: The ANTLR GUI development environment. Home page at [www.antlr.org/works/](http://www.antlr.org/works/) (visited 2007-04-15).
- [BPM04] Ira D. Baxter, Christopher Pidgeon, and Michael Mehlich. DMS: Program transformations for practical scalable software evolution. In *ICSE '04: Proceedings of the 26th International Conference on Software Engineering*, pages 625–634, Washington, DC, USA, 2004. IEEE Computer Society.
- [BPSM<sup>+</sup>] Tim Bray, Jean Paoli, C. M. Sperberg-McQueen, Eve Maler, and François Yergeau. *Extensible Markup Language (XML) 1.0 (Fourth Edition)*. World-Wide Web Consortium.
- [BV00] M.G.J. van den Brand and J.J. Vinju. Rewriting with layout. In Claude Kirchner and Nachum Dershowitz, editors, *Proceedings of RULE'00*, 2000.
- [BV04] Martin Bravenboer and Eelco Visser. Concrete syntax for objects. Domain-specific language embedding and assimilation without restrictions. In Douglas C. Schmidt, editor, *Proceedings of the 19th ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA'04)*, pages 365–383, Vancouver, Canada, October 2004. ACM Press.
- [BvDOV06] Martin Bravenboer, Arthur van Dam, Karina Olmos, and Eelco Visser. Program transformation with scoped dynamic rewrite rules. *Fundamenta Informaticae*, 69(1–2):123–178, 2006.
- [CBE<sup>+</sup>00] Constantinos A. Constantinides, Atef Bader, Tzilla H. Elrad, P. Netinant, and Mohamed E. Fayad. Designing an aspect-oriented framework in an object-oriented environment. *ACM Comput. Surv.*, 32(1es):41, 2000.
- [CC02] Patrick Cousot and Radhia Cousot. Systematic design of program transformation frameworks by abstract interpretation. In *POPL '02: Proceedings of the 29th ACM SIGPLAN-SIGACT symposium on*

- Principles of programming languages*, pages 178–190, New York, NY, USA, 2002. ACM Press.
- [CCH05] James R. Cordy, Ian H. Carmichael, and Russell Halliday. *The TXL Programming Language, Version 8*, January 2005.
- [CDE<sup>+</sup>03] Manuel Clavel, Francisco Durán, Steven Eker, Patrick Lincoln, Narciso Martí-Oliet, José Meseguer, and Carolyn Talcott. The Maude 2.0 system. In Robert Nieuwenhuis, editor, *Rewriting Techniques and Applications (RTA 2003)*, number 2706 in Lecture Notes in Computer Science, pages 76–87. Springer-Verlag, June 2003.
- [CDK<sup>+</sup>01] Robit Chandra, Leonardo Dagum, Dave Kohr, Dror Maydan, Jeff McDonald, and Ramesh Menon. *Parallel programming in OpenMP*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2001.
- [CFG96] P. Ciancarini, D. Fogli, and M. Gaspari. A Logic Language based on Gamma-like Multiset Rewriting. In R. Dyckhoff, H. Herre, and P. Schroeder-Heister, editors, *Proc. 5th Workshop on Extensions of Logic Programming*, volume 1050, pages 83–102, Leipzig, Germany, 1996. Springer-Verlag, Berlin.
- [CH00] Koen Claessen and John Hughes. Quickcheck: a lightweight tool for random testing of haskell programs. In *ICFP '00: Proceedings of the fifth ACM SIGPLAN international conference on Functional programming*, pages 268–279, New York, NY, USA, 2000. ACM Press.
- [CHHP91] James R. Cordy, Charles D. Halpern-Hamu, and Eric Promislow. TXL: a rapid prototyping system for programming language dialects. *Comput. Lang.*, 16(1):97–107, 1991.
- [Chi95] Shigeru Chiba. A metaobject protocol for C++. In *OOPSLA '95: Proceedings of the tenth annual conference on Object-oriented programming systems, languages, and applications*, pages 285–299, New York, NY, USA, 1995. ACM Press.
- [CKK06] Krzysztof Czarnecki, Chang Hwan Peter Kim, and Karl Trygve Kalleberg. Feature models are views on ontologies. In *Proceedings of the 10th International Software Product Line Conference (SPLC'06)*, pages 41–51, Washington, DC, USA, August 2006. IEEE Computer Society.
- [Cla99] James Clark. XSL transformations (XSLT) version 1.0. W3C recommendation, W3C, November 1999.

- [CLR97] Thomas H. Cormen, Charles E. Leiserson, and Ronald L. Rivest. *Introduction to Algorithms*. MIT Press, 1997.
- [CNFP06] Paolina Centonze, Gleb Naumovich, Stephen J. Fink, and Marco Pistoia. Role-based access control consistency validation. In *ISSTA '06: Proceedings of the 2006 international symposium on Software testing and analysis*, pages 121–132, New York, NY, USA, 2006. ACM Press.
- [CNR90] Yih-Farn Chen, Michael Y. Nishimoto, and C. V. Ramamoorthy. The C information abstraction system. *IEEE Trans. Softw. Eng.*, 16(3):325–334, 1990.
- [Cob] Cobertura. <http://cobertura.sourceforge.net> per 2007-03-15.
- [Coh81] Paul M Cohn. *Universal Algebra*. Springer, 1981.
- [Cop05] Tom Copeland. *PMD Applied*. Centennial Books, Nov 2005.
- [Cor04] James R. Cordy. TXL - a language for programming language tools and applications. *ENTCS*, 110:3–31, 2004.
- [CS99] Koen Claessen and David Sands. Observable sharing for functional circuit description. In *ASIAN '99: Proceedings of the 5th Asian Computing Science Conference on Advances in Computing Science*, pages 62–73, London, UK, 1999. Springer-Verlag.
- [DHB92] R. Kent Dybvig, Robert Hieb, and Carl Bruggeman. Syntactic abstraction in Scheme. *Lisp Symb. Comput.*, 5(4):295–326, 1992.
- [DM03] Kyung-Goo Doh and Peter D. Mosses. Composing programming languages by combining action-semantics modules. *Sci. Comput. Program.*, 47(1):3–36, 2003.
- [DO91] Richard A. DeMillo and A. Jefferson Offutt. Constraint-based automatic test data generation. *IEEE Trans. Softw. Eng.*, 17(9):900–910, 1991.
- [DR97] Dale Dougherty and Arnold Robbins. *sed & awk (2nd Edition)*. O'Reilly Media, Inc., Mar 1997.
- [DT96] Merlin Dorfman and Richard H. Thayer. *Software Engineering*. 1996.
- [Ecl] Eclipse. <http://www.eclipse.org> per 2007-03-15.

- [Erl00] Len Erlikh. Leveraging legacy system dollars for e-business. *IT Pro*, pages 17–23, May/June 2000.
- [Fea87] M. S. Feather. A survey and classification of some program transformation approaches and techniques. In *The IFIP TC2/WG 2.1 Working Conference on Program Specification and Transformation*, pages 165–195, Amsterdam, The Netherlands, 1987. North-Holland Publishing Co.
- [FH83] R. Fjeldstad and W. Hamlen. Application program maintenance-report to our respondents. In *Tutorial on Software Maintenance*, pages 13–27. IEEE Press, 1983.
- [GA95] Pascale Le Gall and Agnès Arnould. Formal specifications and test: Correctness and oracle. In Magne Haveræen, Olaf Owe, and Ole-Johan Dahl, editors, *COMPASS/ADT*, volume 1130 of *Lecture Notes in Computer Science*, pages 342–358. Springer, 1995.
- [GHJV95] Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides. *Design Patterns*. Addison-Wesley, 1995.
- [GL00] Samuel Z. Guyer and Calvin Lin. An annotation language for optimizing software libraries. *SIGPLAN Not.*, 35(1):39–52, 2000.
- [GMH81] John D. Gannon, Paul R. McMullin, and Richard G. Hamlet. Data-abstraction implementation, specification, and testing. *ACM Trans. Program. Lang. Syst.*, 3(3):211–223, 1981.
- [GR04] Jeff Gray and Suman Roychoudhury. A technique for constructing aspect weavers using a program transformation engine. In *AOSD '04: Proceedings of the 3rd international conference on Aspect-oriented software development*, pages 36–45, New York, NY, USA, 2004. ACM Press.
- [Gru93] Thomas R. Gruber. Towards principles for the design of ontologies used for knowledge sharing. Technical Report KSL93-04, Stanford University, Stanford, August 1993.
- [HB05] Magne Haveræen and Enida Brkic. Structured testing in Sophus. In Eivind Coward, editor, *Norsk informatikkonferanse NIK'2005*, pages 43–54. Tapir akademisk forlag, Trondheim, Norway, 2005.
- [Hed98] Görel Hedin. Language support for design patterns using attribute extension. In *ECOOP '97: Proceedings of the Workshops on Object-Oriented Technology*, pages 137–140. Springer-Verlag, 1998.

- [Hir03] Robert Hirschfeld. Aspects - aspect-oriented programming with Squeak. In *NODE '02: Revised Papers from the International Conference NetObjectDays on Objects, Components, Architectures, Services, and Applications for a Networked World*, pages 216–232. Springer-Verlag, 2003.
- [HK02] Jan Hannemann and Gregor Kiczales. Design pattern implementation in Java and AspectJ. In *OOPSLA '02: Proceedings of the 17th ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications*, pages 161–173. ACM Press, 2002.
- [HM03] Görel Hedin and Eva Magnusson. JastAdd – an aspect-oriented compiler construction system. *Sci. Comput. Program.*, 47(1):37–58, 2003.
- [HP04] David Hovemeyer and William Pugh. Finding bugs is easy. *SIGPLAN Not.*, 39(12):92–106, 2004.
- [HS96] Merlin Hughes and David Stotts. Daistish: systematic algebraic testing for OO programs in the presence of side-effects. In *ISSTA '96: Proceedings of the 1996 ACM SIGSOFT international symposium on Software testing and analysis*, pages 53–61, New York, NY, USA, 1996. ACM Press.
- [HT05] Akinori Yonezawa Hideaki Tatsuzawa, Hidehiko Masuhara. Aspectual Caml: an aspect-oriented functional language. In Curtis Clifton, Ralf Lämmel, , and Gary T. Leavens, editors, *FOAL 2005 Proceedings: Foundations of Aspect-Oriented Languages Workshop at AOSD 2005*, Chicago, IL, March 2005.
- [HVdMdV05] Elnar Hajiyev, Mathieu Verbaere, Oege de Moor, and Kris de Volder. CodeQuest: querying source code with Datalog. In *OOPSLA '05: Companion to the 20th annual ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications*, pages 102–103, New York, NY, USA, 2005. ACM Press.
- [HWS00] Richard C. Holt, Andreas Winter, and Andy Schürr. GXL: Toward a standard exchange format. In *WCRE '00: Proceedings of the Seventh Working Conference on Reverse Engineering (WCRE'00)*, page 162, Washington, DC, USA, 2000. IEEE Computer Society.
- [JAM99] Paul Jansen, Lex Augusteijn, and Harm Monk. An introduction to Elegant. Technical report, Philips Research Laboratories, Eindhoven, The Netherlands, 1999.

- [Jav] Java 5.0 api. <http://java.sun.com/j2se/1.5.0/docs/api/> per 2007-03-15.
- [Joh78] S. Johnson. *Lint, a C Program Checker*. AT&T Bell Laboratories, 1978.
- [Jon] Joel Jones. Abstract syntax tree implementation idioms. In Brian Marick, editor, *Proceedings of The 10th Conference on Pattern Languages of Programs (PLOP'03)*. Hillside Group. Online publication.
- [JV01] Patricia Johann and Eelco Visser. Fusing logic and control with local transformations: An example optimization. In B. Gramlich and S. Lucas, editors, *Workshop on Reduction Strategies in Rewriting and Programming (WRS'01)*, volume 57 of *Electronic Notes in Theoretical Computer Science*, Utrecht, The Netherlands, May 2001. Elsevier Science Publishers.
- [JV03] Patricia Johann and Eelco Visser. Strategies for fusing logic and control via local, application-specific transformations. Technical Report UU-CS-2003-050, Institute of Information and Computing Sciences, Utrecht University, February 2003.
- [Kah99] Wolfram Kahl. The term graph programming system HOPS. In Rudolf Berghammer and Yassine Lakhnech, editors, *Tool Support for System Specification, Development and Verification*, Advances in Computing Science, pages 136–149, Wien, March 1999. Springer. ISBN: 3-211-83282-3.
- [Kal03] Karl Trygve Kalleberg. User-configurable, high-level transformations with CodeBoost. Master's thesis, University of Bergen, P.O.Box 7800, N-5020 Bergen, Norway, March 2003.
- [Kal06] Karl Trygve Kalleberg. Stratego: A programming language for program manipulation. *ACM Crossroads*, 12(3), April 2006.
- [KD01] Wolfram Kahl and Frank Derichsweiler. Declarative term graph attribution for program generation. *J. UCS*, 7(1):54–70, 2001.
- [KH] Karl Trygve Kalleberg and Magne Haverdaen. JAxT – Java Axiomatic Testing. <http://www.ii.uib.no/mouldable/testing> per 2007-03-15.
- [KHH<sup>+</sup>01] Gregor Kiczales, Erik Hilsdale, Jim Hugunin, Mik Kersten, Jeffrey Palm, and Willian G. Griswold. An overview of AspectJ. In J. Lindskov Knudsen, editor, *ECOOP 2001: Object-Oriented Programming:*

- 15th European Conference*, volume 2072 of *LNCS*, pages 327–353. Springer-Verlag, June 2001.
- [KK04] Günter Kniesel and Helge Koch. Static composition of refactorings. *Sci. Comput. Program.*, 52(1-3):9–51, 2004.
- [KL03] J. Kort and R. Lämmel. Parse-Tree Annotations Meet Re-Engineering Concerns. In *Proc. Source Code Analysis and Manipulation (SCAM'03)*. IEEE Computer Society Press, September 2003.
- [KLM<sup>+</sup>97] Gregor Kiczales, John Lamping, Anurag Menhdhekar, Chris Maeda, Cristina Lopes, Jean-Marc Loingtier, and John Irwin. Aspect-oriented programming. In Mehmet Akşit and Satoshi Matsuoka, editors, *Proceedings European Conference on Object-Oriented Programming*, volume 1241 of *Lecture Notes in Computer Science*, pages 220–242, Berlin, Heidelberg, and New York, 1997. Springer-Verlag.
- [KP97] Philipp W. Kutter and Alfonso Pierantonio. Montages specifications of realistic programming languages. *J. UCS*, 3(5):416–442, 1997.
- [KV05] Karl Trygve Kalleberg and Eelco Visser. Combining aspect-oriented and strategic programming. In Horatiu Cirstea and Narciso Marti-Oliet, editors, *Workshop on Rule-Based Programming (RULE'05)*, Electronic Notes in Theoretical Computer Science, Nara, Japan, April 2005. Elsevier Science Publishers.
- [KV06] Karl Trygve Kalleberg and Eelco Visser. Strategic graph rewriting: Transforming and traversing terms with references. In *Proceedings of the 6th International Workshop on Reduction Strategies in Rewriting and Programming*, Seattle, Washington, August 2006. Online publication.
- [KV07a] Karl Trygve Kalleberg and Eelco Visser. Fusing a transformation language with an open compiler. In Tony Sloane and A. Johnstone, editors, *Seventh Workshop on Language Descriptions, Tools, and Applications (LDTA'07)*, ENTCS, pages 18–31, Braga, Portugal, March 2007. Elsevier.
- [KV07b] Karl Trygve Kalleberg and Eelco Visser. Spoofax: An interactive development environment for program transformation with Stratego/XT. In Tony Sloane and Adrian Johnstone, editors, *Seventh Workshop on Language Descriptions, Tools, and Applications (LDTA'07)*, ENTCS, pages 47–50, Braga, Portugal, March 2007. Elsevier.

- [Läm99] Ralf Lämmel. Declarative aspect-oriented programming. In Olivier Danvy, editor, *Proceedings PEPM'99, 1999 ACM SIGPLAN Workshop on Partial Evaluation and Semantics-Based Program Manipulation PEPM'99, San Antonio (Texas), BRICS Notes Series NS-99-1*, pages 131–146, January 1999.
- [Läm03] Ralf Lämmel. Typed Generic Traversal With Term Rewriting Strategies. *Journal of Logic and Algebraic Programming*, 54, 2003. Also available as arXiv technical report cs.PL/0205018.
- [Lin05] Christian Lindig. Random testing of C calling conventions. In *AADEBUG'05: Proceedings of the sixth international symposium on Automated analysis-driven debugging*, pages 3–12, New York, NY, USA, 2005. ACM Press.
- [LK97] C.V Lopes and G. Kiczales. D: A language framework for distributed programming. Technical Report SPL97-010, P9710047, Xerox Palo Alto Research Center, 1997.
- [LL00] Martin Lippert and Cristina Videira Lopes. A study on exception detection and handling using aspect-oriented programming. *ICSE*, 00:418, 2000.
- [LVV03] Ralf Lämmel, Eelco Visser, and Joost Visser. Strategic programming meets adaptive programming. In *Proceedings of Aspect-Oriented Software Development (AOSD'03)*, pages 168–177, Boston, USA, March 2003. ACM Press.
- [McK84] J McKee. Maintenance as a function of design. In *Proceedings of the AFIPS National Computer Conference*, pages 187–193, 1984.
- [MdVHC03] C Mundie, P de Vries, P Haynes, and M Corwine. Trustworthy computing. Technical report, 2003. White paper.
- [MGR05] Pierre-Etienne Moreau, Julien Guyon, and Antoine Reilles. *Tom User's Guide*. Lorraine Laboratory of IT Research and its Applications, Nancy, France, Jul 2005.
- [MLL05] Michael Martin, Benjamin Livshits, and Monica S. Lam. Finding application errors and security flaws using PQL: a program query language. In *OOPSLA '05: Proc. of the ACM SIGPLAN Conf. Object oriented programming, systems, languages, and applications*, pages 365–383, New York, NY, USA, 2005. ACM Press.

- [MM95] J. Meseguer and N. Martí-Oliet. From abstract data types to logical frameworks. In E. Astesiano, G. Reggio, and A. Tarlecki, editors, *Recent Trends in Data Type Specification: 10th Workshop on Specification of Abstract Data Types joint with the 5th COMPASS Workshop*, volume 906 of *Lecture Notes in Computer Science*, pages 48–80. Springer, 1995.
- [MO03] Mira Mezini and Klaus Ostermann. Conquering aspects with Caesar. In *AOSD '03: Proceedings of the 2nd international conference on Aspect-oriented software development*, pages 90–99. ACM Press, 2003.
- [Moa90] J. Moad. Maintaining the competitive edge. *Datamation*, 36(4):61–62,62,66, Feb 1990.
- [Mog91] Eugenio Moggi. Notions of computation and monads. *Inf. Comput.*, 93(1):55–92, 1991.
- [Mor98] Pierre-Etienne Moreau. A choice-point library for backtrack programming. In Konstantinos Sagonas, editor, *Proceedings of the JICSLP-98 Workshop on Implementation Technologies for Programming Languages based on Logic*, pages 16–31, 1998.
- [Mos04a] Peter D. Mosses. Modular language descriptions. In Gabor Karsai and Eelco Visser, editors, *GPCE*, volume 3286 of *Lecture Notes in Computer Science*, page 489. Springer, 2004.
- [Mos04b] Peter D. Mosses. Modular structural operational semantics. *J. Log. Algebr. Program.*, 60-61:195–228, 2004.
- [MRB<sup>+</sup>] MohammadReza Mousavi, Michel Reniers, Twan Basten, Michel Chaudron, Giovanni Russello, Angelo Cursaro, Sandeep Shukla, Rajesh Gupta, and Douglas C. Schmidt. Using Aspect-GAMMA in the design of embedded systems. In *Proceedings of Seventh Annual IEEE International Workshop on High Level Design Validation and Test*, pages 69–74, Cannes, France. IEEE Computer Society Press, Los Alamitos, CA, USA, 2002.
- [MRV03] Pierre-Etienne Moreau, Christophe Ringeissen, and Marian Vittek. A pattern matching compiler for multiple target languages. In *12th International Conference on Compiler Construction*, LNCS, pages 61–76. Springer, 2003.
- [mtj06] Matrix Toolkits for Java. <http://rs.cipr.uib.no/mtj/>, 2006.

- [MWT94] H. Müller, K. Wong, and S. Tilley. Understanding software systems using reverse engineering technology. In *The 62nd Congress of L'Association Canadienne Francaise pour L'Avancement des Sciences Proceedings (ACFAS)*, volume 4, pages 41–48, 1994.
- [NCM03] Nathaniel Nystrom, Michael R. Clarkson, and Andrew C. Myers. Polyglot: An extensible compiler framework for Java. In *12th International Conference on Compiler Construction*, LNCS, pages 138–152. Springer, 2003.
- [NNZ04] U. Nickel, J. Niere, and A. Zundorf. Tool demonstration: The FUJABA environment. In *The 22nd International Conference on Software Engineering (ICSE)*. ACM Press, 2004.
- [NoU] NoUnit. <http://nunit.sourceforge.net> per 2007-03-15.
- [Opd92] William F. Opdyke. *Refactoring object-oriented frameworks*. PhD thesis, Champaign, IL, USA, 1992.
- [OV05] Karina Olmos and Eelco Visser. Composing source-to-source data-flow transformations with rewriting strategies and dependent dynamic rewrite rules. In Rastislav Bodik, editor, *CC'05: 14th International Conference on Compiler Construction*, volume 3443 of *Lecture Notes in Computer Science*, pages 204–220. Springer-Verlag, April 2005.
- [Pag93] Robert Page. APTS external specification manual (rough draft). Available at <http://www.cs.nyu.edu/jessie/>, 1993. Unpublished manuscript.
- [Pai96] Robert Paige. Future directions in program transformations. *ACM Comput. Surv.*, 28(4es):170, 1996.
- [Par72] D. L. Parnas. On the criteria to be used in decomposing systems into modules. *Commun. ACM*, 15(12):1053–1058, 1972.
- [Par86] H. Partsch. Transformational program development in a particular problem domain. *Science of Computer Programming*, 7(2):99–241, 1986.
- [Pfl05] Shari Lawrence Pfleeger. *Software engineering: theory and practice (3rd ed)*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 2005.
- [PK82] Robert Paige and Shaye Koenig. Finite differencing of computable expressions. *ACM Trans. Program. Lang. Syst.*, 4(3):402–454, 1982.

- [Plu99] D. Plump. *Handbook of graph grammars and computing by graph transformation: vol. 2: applications, languages, and tools*, chapter Term graph rewriting, pages 3–61. World Scientific Publishing Co., Inc., River Edge, NJ, USA, 1999.
- [Plu01] Detlef Plump. Essentials of term graph rewriting. *Electr. Notes Theor. Comput. Sci.*, 51, 2001.
- [PP96] Alberto Pettorossi and Maurizio Proietti. Future directions in program transformation. *ACM Comput. Surv.*, 28(4es):171, 1996.
- [PQ95] T. J. Parr and R. W. Quong. ANTLR: a predicated-LL(k) parser generator. *Softw. Pract. Exper.*, 25(7):789–810, 1995.
- [PS83] H. Partsch and R. Steinbrüggen. Program transformation systems. *ACM Computing Surveys*, 15(3):199–236, 1983.
- [PvE98] M. Plasmeijer and M. van Eekelen. Language report: Concurrent Clean. Technical Report CSI-R9816, Computing Science Inst., U. of Nijmegen, Nijmegen, The Netherlands, 1998.
- [Sch04] Andreas Schürr. *The PROGRES Language Manual Version 9.x*. Lehrstuhl für Informatik III, RWTH Aachen, Aachen, Germany, 2004.
- [sgm86] *ISO 8879:1986 – Standard Generalized Markup Language*. International Organisation for Standardization (ISO), 1986.
- [SGSP02] Olaf Spinczyk, Andreas Gal, and Wolfgang Schröder-Preikschat. AspectC++: an aspect-oriented extension to the C++ programming language. In *CRPITS '02: Proceedings of the Fortieth International Conference on Tools Pacific*, pages 53–60. Australian Computer Society, Inc., 2002.
- [SLA02] P. David Stotts, Mark Lindsey, and Angus Antley. An informal formal method for systematic junit test case generation. In Don Wells and Laurie A. Williams, editors, *XP/Agile Universe*, volume 2418 of *Lecture Notes in Computer Science*, pages 131–143. Springer, 2002.
- [SLS03] Macneil Shonle, Karl Lieberherr, and Ankit Shah. XAspects: an extensible system for domain-specific aspect languages. In *OOP-SLA '03: Companion of the 18th annual ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications*, pages 28–37, New York, NY, USA, 2003. ACM Press.

- [SLTM91] Kari Smolander, Kalle Lyytinen, Veli-Pekka Tahvanainen, and Pentti Marttiin. MetaEdit: a flexible graphical environment for methodology modelling. In *CAiSE '91: Proceedings of the third international conference on Advanced information systems engineering*, pages 168–193, New York, NY, USA, 1991. Springer-Verlag New York, Inc.
- [SNDH04] Rozalia Szabo-Nacsa, Peter Divianszky, and Zoltan Horvath. Prototype environment for refactoring Clean programs. In *Proceedings of the Sixth International Conference on Applied Informatics (ICAI'04)*, 2004.
- [Som00] I. Sommerville. *Software Engineering (6th Edition)*. Addison-Wesley, 2000.
- [ST88] Donald Sannella and Andrzej Tarlecki. Specifications in an arbitrary institution. *Information and Computation*, 76:165–210, 1988.
- [SY86] R E Strom and S Yemini. Tpestate: A programming language concept for enhancing software reliability. *IEEE Trans. Softw. Eng.*, 12(1):157–171, 1986.
- [SY93] R. E. Strom and D. M. Yellin. Extending tpestate checking using conditional liveness analysis. *IEEE Trans. Softw. Eng.*, 19(5):478–485, 1993.
- [Tae04] Gabriele Taentzer. AGG: A graph transformation environment for modeling and validation of software. In *Proceedings of the 2nd International Workshop and Symposium on Applications of Graph Transformation with Industrial Relevance (AGTIVE 2003)*, volume 3062 of *Lecture Notes in Computer Science*, pages 446 – 453. Springer, Jan 2004.
- [TCIK00] Michiaki Tatsubori, Shigeru Chiba, Kozo Itano, and Marc-Olivier Killijian. OpenJava: A class-based macro system for java. In *Proceedings of the 1st OOPSLA Workshop on Reflection and Software Engineering*, pages 117–133, London, UK, 2000. Springer-Verlag.
- [Ter03] Terese. *Term Rewriting Systems*, volume 55 of *Cambridge Tracts in Theoretical Computer Science*. Cambridge University Press, 2003.
- [TL02] K. C. Tai and Y. Lie. A test generation strategy for pairwise testing. *IEEE Trans. Softw. Eng.*, 28(1):109–111, 2002.

- [Ulr90] W. Ulrich. The evolutionary growth of software engineering and the decade ahead. *American Programmer*, 3(10):12–20, 1990.
- [VB98] Eelco Visser and Zine-el-Abidine Benaissa. A core language for rewriting. In C. Kirchner and H. Kirchner, editors, *Second International Workshop on Rewriting Logic and its Applications (WRLA'98)*, volume 15 of *Electronic Notes in Theoretical Computer Science*, Pont-à-Mousson, France, September 1998. Elsevier Science Publishers.
- [VBT98] Eelco Visser, Zine-el-Abidine Benaissa, and Andrew Tolmach. Building program optimizers with rewriting strategies. In *Proceedings of the third ACM SIGPLAN International Conference on Functional Programming (ICFP'98)*, pages 13–26. ACM Press, September 1998.
- [vdBdJKO00] M. G. T. van den Brand, H. A. de Jong, P. Klint, and P. A. Olivier. Efficient annotated terms. *Softw. Pract. Exper.*, 30(3):259–291, 2000.
- [vdBHKO02] Mark G. J. van den Brand, J. Heering, P. Klint, and P. A. Olivier. Compiling language definitions: the ASF+SDF compiler. *ACM Trans. Program. Lang. Syst.*, 24(4):334–368, 2002.
- [vdBV00] M.G.J. van den Brand and J.J. Vinju. Rewriting with layout. In Claude Kirchner and Nachum Dershowitz, editors, *Proceedings of the Second International Workshop on Rule-based Programming (RULE'00)*, 2000.
- [vdBvDH<sup>+</sup>01] Mark G. J. van den Brand, Arie van Deursen, Jan Heering, H. A. de Jong, Merijn de Jonge, Tobias Kuipers, Paul Klint, Leon Moonen, Pieter A. Olivier, Jeroen Scheerder, Jurgen J. Vinju, Eelco Visser, and Joost Visser. The ASF+SDF meta-environment: A component-based language development environment. In *CC '01: Proceedings of the 10th International Conference on Compiler Construction*, pages 365–370, London, UK, 2001. Springer-Verlag.
- [vdDKT93] Arie van Deursen, Paul Klint, and Frank Tip. Origin tracking. *J. Symb. Comput.*, 15(5/6):523–545, 1993.
- [VEdM06] Mathieu Verbaere, Ran Ettinger, and Oege de Moor. JunGL: a scripting language for refactoring. In *ICSE '06: Proceeding of the 28th international conference on Software engineering*, pages 172–181, New York, NY, USA, 2006. ACM Press.

- [vEM02] Eva van Emden and Leon Moonen. Java quality assurance by detecting code smells. In *Proceedings of the 9th Working Conference on Reverse Engineering*. IEEE Computer Society Press, October 2002.
- [VG98] Todd L. Veldhuizen and Dennis Gannon. Active libraries: Rethinking the roles of compilers and libraries. In *Proceedings of the SIAM Workshop on Object Oriented Methods for Inter-operable Scientific and Engineering Computing (OO'98)*, Philadelphia, PA, USA, 1998. SIAM.
- [Vis97] Eelco Visser. *Syntax Definition for Language Prototyping*. PhD thesis, University of Amsterdam, September 1997.
- [Vis99] Eelco Visser. Strategic pattern matching. In P. Narendran and M. Rusinowitch, editors, *Rewriting Techniques and Applications (RTA'99)*, volume 1631 of *Lecture Notes in Computer Science*, pages 30–44, Trento, Italy, July 1999. Springer-Verlag.
- [Vis02] Eelco Visser. Meta-programming with concrete object syntax. In Don Batory, Charles Consel, and Walid Taha, editors, *Generative Programming and Component Engineering (GPCE'02)*, volume 2487 of *Lecture Notes in Computer Science*, pages 299–315, Pittsburgh, PA, USA, October 2002. Springer-Verlag.
- [Vis05a] Eelco Visser. A survey of strategies in rule-based program transformation systems. *Journal of Symbolic Computation*, 40(1):831–873, 2005. Special issue on Reduction Strategies in Rewriting and Programming.
- [Vis05b] Eelco Visser. Transformations for abstractions. In Jens Krinke and Giulio Antoniol, editors, *Fifth IEEE International Workshop on Source Code Analysis and Manipulation (SCAM'05)*, pages 3–12, Budapest, Hungary, October 2005. IEEE Computer Society Press. (Keynote paper).
- [vV02] Arie van Deursen and Joost Visser. Building program understanding tools using visitor combinators. In *Proceedings 10th Int. Workshop on Program Comprehension, IWPC 2002*, pages 137–146. IEEE Computer Society, 2002.
- [vWV03] Jonne van Wijngaarden and Eelco Visser. Program transformation mechanics. a classification of mechanisms for program transformation with a survey of existing transformation systems. Technical

- Report UU-CS-2003-048, Institute of Information and Computing Sciences, Utrecht University, 2003.
- [Wad87] P. Wadler. Views: a way for pattern matching to cohabit with data abstraction. In *POPL '87: Proceedings of the 14th ACM SIGACT-SIGPLAN symposium on Principles of programming languages*, pages 307–313, New York, NY, USA, 1987. ACM Press.
- [Wad92] Philip Wadler. The essence of functional programming. In *POPL '92: Proceedings of the 19th ACM SIGPLAN-SIGACT symposium on Principles of programming languages*, pages 1–14, New York, NY, USA, 1992. ACM Press.
- [WAKS97] Daniel C. Wang, Andrew W. Appel, Je L. Korn, and Christopher S. Serra. The Zephyr abstract syntax description language. In *Proceedings of the USENIX Conference on Domain-Specific Languages*, pages 213–228, 1997.
- [War89] M. Ward. Proving program refinements and transformations. Dphil thesis, Oxford University, 1989.
- [War99] Martin Ward. Assembler to C migration using the FermaT transformation system. In *ICSM '99: Proceedings of the IEEE International Conference on Software Maintenance*, page 67, Washington, DC, USA, 1999. IEEE Computer Society.
- [War02] M. Ward. Program slicing via FermaT transformations. In *COMP-SAC'02: 26th Annual International Computer Software and Applications Conference*, Los Alamitos, California, USA, August 2002. IEEE Computer Society Press.
- [WFW<sup>+</sup>94] Robert P. Wilson, Robert S. French, Christopher S. Wilson, Saman P. Amarasinghe, Jennifer M. Anderson, Steve W. K. Tjiang, Shih-Wei Liao, Chau-Wen Tseng, Mary W. Hall, Monica S. Lam, and John L. Hennessy. SUIF: an infrastructure for research on parallelizing and optimizing compilers. *SIGPLAN Not.*, 29(12):31–37, 1994.
- [Win99] Victor L. Winter. An overview of HATS: A language independent High Assurance Transformation System. In *ASSET '99: Proceedings of the 1999 IEEE Symposium on Application - Specific Systems and Software Engineering and Technology*, page 222, Washington, DC, USA, 1999. IEEE Computer Society.

- 
- [Win03] Thomas (Rho) Windeln. LogicAJ – eine erweiterung von AspectJ um logische meta-programmierung. Master's thesis, University of Bonn, Bonn, Germany, 2003.
- [ZSJG79] M. Zelkowitz, A. Shaw, and J J. Gannon. *Principles of Software Engineering and Design*. Prentice-Hall, 1979.