

- [1] www.codeboost.org.
- [2] www.stratego-language.org.
- [3] M. Askari and R. Al-Ekram. Bringing smalltalk blocks to java through transformation techniques.
- [4] P. Avgustinov, E. Hajiyev, N. Ongkingco, O. de Moor, D. Sereni, J. Tibble, and M. Verbaere. Semantics of static pointcuts in AspectJ. In *POPL'07*, 2007.
- [5] A. H. Bagge, M. Bravenboer, K. T. Kalleberg, K. Muilwijk, and E. Visser. Adaptive code reuse by aspects, cloning and renaming. Technical Report UU-CS-2005-031, Department of Information and Computing Sciences, Universiteit Utrecht, Utrecht, The Netherlands, August 2005.
- [6] A. H. Bagge, V. David, M. Haverlaen, and K. T. Kalleberg. Stayin' alert: moulding failure and exceptions to your needs. In *GPCE'06*, 2006.
- [7] A. H. Bagge and K. T. Kalleberg. Dsal = library+notation: Program transformation for domain-specific aspect languages. In *DSAL'06*, 2006.
- [8] O. S. Bagge. CodeBoost: A framework for transforming C++ programs. Master's thesis, University of Bergen, P.O.Box 7800, N-5020 Bergen, Norway, March 2003.
- [9] O. S. Bagge, M. Haverlaen, and E. Visser. A C++ transformation framework (extended abstract). In *Proceedings of the Nordic Workshop on Programming Theory (NWPT'00)*, Bergen, Norway, October 2000.
- [10] O. S. Bagge, K. T. Kalleberg, M. Haverlaen, and E. Visser. Design of the CodeBoost transformation system for domain-specific optimisation of C++ programs. In D. Binkley and P. Tonella, editors, *Third IEEE International Workshop on Source Code Analysis and Manipulation (SCAM'03)*, pages 65–74, Amsterdam, The Netherlands, September 2003. IEEE Computer Society Press.
- [11] A. Borghi, V. David, and A. Demaille. C-transformers: A framework to write c program transformations.
- [12] E. Bouwers, M. Bravenboer, and E. Visser. Grammar engineering support for precedence rule recovery and compatibility checking. In A. Sloane and A. Johnstone, editors, *Seventh Workshop on Language Descriptions, Tools, and Applications (LDTA'07)*, pages 82–96, Braga, Portugal, March 2007.
- [13] M. G. J. Brandvan den Brand, J. Scheerder, J. Vinju, and E. Visser. Disambiguation filters for scannerless generalized LR parsers. In N. Horspool, editor, *Compiler Construction (CC'02)*, volume 2304 of *Lecture Notes in Computer Science*, pages 143–158, Grenoble, France, April 2002. Springer-Verlag.
- [14] M. G. J. Brandvan den Brand and E. Visser. From Box to T_EX: An algebraic approach to the generation of documentation tools. Technical Report P9420, Programming Research Group, University of Amsterdam, July 1994.

- [15] M. Bravenboer. Connecting XML processing and term rewriting with tree grammars. Master's thesis, Utrecht University, Utrecht, The Netherlands, November 2003.
- [16] M. Bravenboer, R. de Groot, and E. Visser. MetaBorg in action: Examples of domain-specific language embedding and assimilation using Stratego/XT. In R. Lämmel and J. Saraiva, editors, *Proceedings of the Summer School on Generative and Transformational Techniques in Software Engineering (GTTSE'05)*, volume 4143 of *Lecture Notes in Computer Science*, pages 297–311, Braga, Portugal, 2006. Springer Verlag.
- [17] M. Bravenboer, E. Dolstra, and E. Visser. Preventing injection attacks with syntax embeddings. A host and guest language independent approach. In J. Lawall, editor, *Generative Programming and Component Engineering (GPCE'07)*, pages 3–12, New York, NY, USA, October 2007. ACM.
- [18] M. Bravenboer, E. Dolstra, and E. Visser. Preventing injection attacks with syntax embeddings. A host and guest language independent approach. *Science of Computer Programming*, 2008. Under construction; Invited for special issue of Generative Programming and Component Engineering (GPCE'07).
- [19] M. Bravenboer, K. T. Kalleberg, R. Vermaas, and E. Visser. *Stratego/XT Tutorial, Examples, and Reference Manual for Stratego/XT 0.16*. Department of Information and Computing Sciences, Universiteit Utrecht, Utrecht, The Netherlands, November 2005.
- [20] M. Bravenboer, K. T. Kalleberg, R. Vermaas, and E. Visser. Stratego/XT 0.16. Components for transformation systems. In *ACM SIGPLAN 2006 Workshop on Partial Evaluation and Program Manipulation (PEPM'06)*, Charleston, South Carolina, January 2006. ACM SIGPLAN.
- [21] M. Bravenboer, K. T. Kalleberg, R. Vermaas, and E. Visser. *Stratego/XT Tutorial, Examples, and Reference Manual (latest)*. Department of Information and Computing Sciences, Universiteit Utrecht, Utrecht, The Netherlands, 2006. <http://www.strategoxt.org>.
- [22] M. Bravenboer, K. T. Kalleberg, R. Vermaas, and E. Visser. Stratego/XT 0.17. A language and toolset for program transformation. *Science of Computer Programming*, 2008. (To appear).
- [23] M. Bravenboer, E. Tanter, and E. Visser. Declarative, formal, and extensible syntax definition for AspectJ. A case for scannerless generalized-lr parsing. In W. R. Cook, editor, *Proceedings of the 21th ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA'06)*, pages 209–228, Portland, Oregon, USA, October 2006. ACM Press.
- [24] M. Bravenboer, A. van Dam, K. Olmos, and E. Visser. Program transformation with scoped dynamic rewrite rules. *Fundamenta Informaticae*, 69(1–2):123–178, 2006.

- [25] M. Bravenboer, R. Vermaas, J. Vinju, and E. Visser. Generalized type-based disambiguation of meta programs with concrete object syntax. In R. Glück and M. Lowry, editors, *Proceedings of the Fourth International Conference on Generative Programming and Component Engineering (GPCE'05)*, volume 3676 of *Lecture Notes in Computer Science*, pages 157–172, Tallinn, Estonia, September 2005. Springer.
- [26] M. Bravenboer and E. Visser. Guiding visitors: Separating navigation from computation. Technical Report UU-CS-2001-42, Institute of Information and Computing Sciences, Utrecht University, 2001.
- [27] M. Bravenboer and E. Visser. Rewriting strategies for instruction selection. In S. Tison, editor, *Rewriting Techniques and Applications (RTA'02)*, volume 2378 of *Lecture Notes in Computer Science*, pages 237–251, Copenhagen, Denmark, July 2002. Springer-Verlag.
- [28] M. Bravenboer and E. Visser. Concrete syntax for objects. Domain-specific language embedding and assimilation without restrictions. In D. 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.
- [29] M. Bravenboer and E. Visser. Reusable and adaptable strategies for generative programming. In M. Haveraaen et al., editors, *GPCE Software Transformation Systems Workshop (STSW'04)*, Vancouver, Canada, October 2004. Position paper.
- [30] M. Bravenboer and E. Visser. Designing syntax embeddings and assimilations for language libraries. In *4th International Workshop on Software Language Engineering (ATEM'07)*, volume 4/2007 of *Mainzer Informatik-Berichte*, pages 100–116, Nashville, USA, 2007. Johannes Gutenberg Universität.
- [31] M. Bravenboer and E. Visser. Designing syntax embeddings and assimilations for language libraries. In *MoDELS'07 Satellite Events Proceedings*, Lecture Notes in Computer Science, Nashville, USA, 2008. Springer. Selected paper from the Workshop on Language Engineering (ATEM'07).
- [32] M. Buss, S. A. Edwards, B. Yao, and D. Waddington. Pointer analysis for source-to-source transformations. In J. Krinke and G. Antoniol, editors, *Fifth IEEE International Workshop on Source Code Analysis and Manipulation (SCAM'05)*, Budapest, Hungary, 2005. IEEE Computer Society Press.
- [33] W. Caarls, P. P. Jonker, and H. Corporaal. Algorithmic skeletons for stream programming in embedded heterogeneous parallel image processing.
- [34] V. David, A. Demaille, and O. Gournet. Attribute grammars for modular disambiguation.
- [35] A. de Bruijn. Implementation of inlining in Stratego. Master's thesis, Utrecht University, Utrecht, The Netherlands, August 2001.

- [36] R. de Groot. Design and implementation of embedded domain-specific languages. Master's thesis, Utrecht University, Utrecht, The Netherlands, September 2005. INF/SCR-05-10.
- [37] M. de Jonge. A pretty-printer for every occasion. In I. Ferguson, J. Gray, and L. Scott, editors, *Proceedings of the 2nd International Symposium on Constructing Software Engineering Tools (CoSET2000)*. University of Wollongong, Australia, 2000.
- [38] M. de Jonge. Source tree composition. In C. Gacek, editor, *Proceedings: Seventh International Conference on Software Reuse*, volume 2319 of *LNCS*, pages 17–32. Springer-Verlag, Apr. 2002.
- [39] M. de Jonge and R. Monajemi. Cost-effective maintenance tools for proprietary languages. In *Proceedings: International Conference on Software Maintenance (ICSM 2001)*, pages 240–249. IEEE Computer Society Press, Nov. 2001.
- [40] M. de Jonge, E. Visser, and J. Visser. XT: A bundle of program transformation tools. In M. G. J. van den Brand and D. Parigot, editors, *Workshop on Language Descriptions, Tools and Applications (LDTA'01)*, volume 44 of *Electronic Notes in Theoretical Computer Science*. Elsevier Science Publishers, April 2001.
- [41] M. de Jonge and J. Visser. XT capita selecta. In E. Visser, editor, *Proceedings of the Second Stratego Users Day (SUD'01)*, pages 18–35. Institute for Information and Computing Sciences, Utrecht University, February 2001.
- [42] A. Demaille, S. Peyronnet, and B. Sigoure. Modeling of sensor networks using xrm.
- [43] E. Dolstra. First-class rules and generic traversal for program transformation languages. Master's thesis, Utrecht University, Utrecht, The Netherlands, August 2001. INF/SCR-2001-15.
- [44] E. Dolstra. Functional stratego. In E. Visser, editor, *Proceedings of the Second Stratego Users Day (SUD'01)*, pages 10–17, 2001.
- [45] E. Dolstra and E. Visser. First-class rules and generic traversal. Technical Report UU-CS-2001-38, Institute of Information and Computing Sciences, Utrecht University, Utrecht, The Netherlands, 2001.
- [46] E. Dolstra and E. Visser. Building interpreters with rewriting strategies. In M. van den Brand and R. Laemmel, editors, *Workshop on Language Descriptions, Tools and Applications (LDTA'02)*, volume 65/3 of *Electronic Notes in Theoretical Computer Science*, Grenoble, France, April 2002. Elsevier Science Publishers.
- [47] E. Dolstra and H. Westra. Transformaties op Mondrian met Stratego. Term paper (in dutch) <http://www.cs.uu.nl/~visser/ftp/DW00.pdf> Utrecht University, July 2000.

- [48] J. Fabry, E. Tanter, and T. D'Hondt. ReLax: Implementing KALA over the Reflex AOP kernel. In *Proceedings of the 2nd Workshop on Domain-Specific Aspect Languages (DSAL 2007)*, Vancouver, Canada, March 2007. ACM.
- [49] B. Fischer and E. Visser. Adding concrete syntax to a Prolog-based program synthesis system (extended abstract). In M. Bruynooghe, editor, *Pre-proceedings of the International Symposium on Logic Based Program Synthesis and Transformation (LOPSTR'03)*, number Report CW 365, pages 39–48, Uppsala, Sweden, August 2003. Department of Computer Science, K. U. Leuven.
- [50] B. Fischer and E. Visser. Adding concrete syntax to a prolog-based program synthesis system (extended abstract). In M. Bruynooghe, editor, *13th International Symposium Logic Based Program Synthesis and Transformation: LOPSTR 2003*, volume 3018 of *Lecture Notes in Computer Science*, pages 56–58, Heidelberg, November 2004. Springer-Verlag.
- [51] B. Fischer and E. Visser. Retrofitting the AutoBayes program synthesis system with concrete object syntax. In C. Lengauer et al., editors, *Domain-Specific Program Generation*, volume 3016 of *Lecture Notes in Computer Science*, pages 239–253. Springer-Verlag, 2004.
- [52] S. Fleissner and E. Baniassad. A commensalistic software system. In *Companion to the 21st ACM SIGPLAN conference on Object-oriented programming systems, languages, and applications*, pages 560 – 573. ACM SIGPLAN, 2006.
- [53] S. V. Goncharov, A. Y. Rudich, and Y. S. Usenko. An implementation of a linearization algorithm using rewriting strategies. In *Proc. 1st International Conference on Theoretical and Applied Aspects of Program Systems Development (TAAPSD'04)*, pages 13–17, Kyiv, Ukraine, October 5-8 2004.
- [54] L. Hamey and S. Goldrei. Implementing a domain-specific language using stratego/xt. In A. Sloane and A. Johnstone, editors, *Seventh Workshop on Language Descriptions, Tools, and Applications (LDTA'07)*, pages 32–46, Braga, Portugal, March 2007.
- [55] N. Janssen. Transformation tool composition. Master's thesis, Utrecht University, Utrecht, The Netherlands, March 2005. INF/SCR-04-64.
- [56] P. Johann and E. Visser. Warm fusion in Stratego: A case study in the generation of program transformation systems. *Annals of Mathematics and Artificial Intelligence*, 29(1–4):1–34, 2000.
- [57] P. Johann and E. 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.
- [58] P. Johann and E. 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.

- [59] K. T. Kalleberg. User-configurable, high-level transformations with CodeBoost. Master's thesis, University of Bergen, P.O.Box 7800, N-5020 Bergen, Norway, March 2003.
- [60] K. T. Kalleberg and E. Visser. Combining aspect-oriented and strategic programming. In H. Cirstea and N. Marti-Oliet, editors, *Workshop on Rule-Based Programming (RULE'05)*, volume 147 of *Electronic Notes in Theoretical Computer Science*, pages 5–30, Nara, Japan, 2006. Elsevier Science Publishers.
- [61] K. T. Kalleberg and E. Visser. Fusing a transformation language with an open compiler. In A. Sloane and A. Johnstone, editors, *Seventh Workshop on Language Descriptions, Tools, and Applications (LDTA'07)*, pages 18–31, Braga, Portugal, March 2007.
- [62] K. T. Kalleberg and E. Visser. Spoofox: An interactive development environment for program transformation with Stratego/XT. In A. Sloane and A. Johnstone, editors, *Seventh Workshop on Language Descriptions, Tools, and Applications (LDTA'07)*, pages 47–50, Braga, Portugal, March 2007.
- [63] R. Kieburtz. A logic for rewriting strategies. In *Workshop on Strategies in Automated Deduction*, Siena, Italy, 2001.
- [64] P. Klint and E. Visser. Using filters for the disambiguation of context-free grammars. In G. Pighizzini and P. San Pietro, editors, *Proc. ASMICS Workshop on Parsing Theory*, pages 1–20, Milano, Italy, October 1994. Tech. Rep. 126–1994, Dipartimento di Scienze dell'Informazione, Università di Milano.
- [65] J. Kruger. Specification of loop optimizations in Stratego. Improving the autobayes system. Master's thesis, Utrecht University, Utrecht, The Netherlands, November 2003.
- [66] R. Laemmel, E. Visser, and J. Visser. Strategic programming meets adaptive programming. In *Proceedings of Aspect-Oriented Software Development (AOSD'03)*, pages 168–177, Boston, USA, March 2003. ACM Press.
- [67] R. Lämmel. Towards typeful stratego. In E. Visser, editor, *Proceedings of the Second Stratego Users Day (SUD'01)*, pages 4–9. Institute for Information and Computing Sciences, Utrecht University, February 2001.
- [68] F. Latry, J. Mercadal, and C. Consel. Processing domain-specific modeling languages: A case study in telephony services.
- [69] B. Luttik and E. Visser. Specification of rewriting strategies. In M. P. A. Sellink, editor, *2nd International Workshop on the Theory and Practice of Algebraic Specifications (ASF+SDF'97)*, Electronic Workshops in Computing, Berlin, November 1997. Springer-Verlag.
- [70] K. Olmos and E. Visser. Strategies for source-to-source constant propagation. In B. Gramlich and S. Lucas, editors, *Workshop on Reduction Strategies (WRS'02)*, volume 70 of *Electronic Notes in Theoretical Computer Science*, page 20, Copenhagen, Denmark, July 2002. Elsevier Science Publishers.

- [71] K. Olmos and E. Visser. Turning dynamic typing into static typing by program specialization. In D. Binkley and P. Tonella, editors, *Third IEEE International Workshop on Source Code Analysis and Manipulation (SCAM'03)*, pages 141–150, Amsterdam, The Netherlands, September 2003. IEEE Computer Society Press.
- [72] K. Olmos and E. Visser. Composing source-to-source data-flow transformations with rewriting strategies and dependent dynamic rewrite rules. In R. Bodik, editor, *14th International Conference on Compiler Construction (CC'05)*, volume 3443 of *Lecture Notes in Computer Science*, pages 204–220. Springer-Verlag, April 2005.
- [73] K. J. Olmos. Reverse engineering transformations for DSP code generation. Master's thesis, Utrecht University and Universidad Mayor de San Simón, Utrecht, The Netherlands, November 2000.
- [74] J. Riehl. Assimilating metaborg: Embedding language tools in languages. In *GPCE'06*.
- [75] <http://www.syntax-definition.org>.
- [76] <http://www.stratego-language.org>.
- [77] L. Swart. Partial evaluation using rewrite rules. A specification of a partial evaluator for Similix in Stratego. Master's thesis, Utrecht University, Utrecht, The Netherlands, August 2002. INF/SCR-2002-12.
- [78] É. Tanter. An extensible kernel language for AOP. In *Proceedings of AOSD Workshop on Open and Dynamic Aspect Languages*, Bonn, Germany, 2006.
- [79] R. van Beusekom. A vectorizer for octave. Master's thesis, Utrecht University, Utrecht, The Netherlands, February 2005. INF/SRC-04-53.
- [80] A. van Dam. Simplifying the simplifier. HsOpt: a modular, rewrite rule based simplifier for the helium compiler, a non-strict functional compiler. Master's thesis, Utrecht University, Utrecht, The Netherlands, July 2003. INF/SCR-03-25.
- [81] A. van Dam. Extending dynamic rules. An application-oriented study into Stratego's new dynamic rules. Master's thesis, Utrecht University, Utrecht, The Netherlands, February 2004. INF/SCR-04-25.
- [82] J. van Wijngaarden. Code generation from a domain specific language. designing and implementing complex program transformations. Master's thesis, Utrecht University, Utrecht, The Netherlands, July 2003. INF/SCR-03-29.
- [83] R. B. Vermaas. xDoc. An extensible documentation generator. Master's thesis, Utrecht University, Utrecht, The Netherlands, February 2004. INF/SCR-03-41.
- [84] E. Visser. A case study in optimizing parsing schemata by disambiguation filters. In S. Fischer and M. Trautwein, editors, *Proceedings Accolade95*, pages 153–167, Amsterdam, September 1995. The Dutch Graduate School in Logic.

- [85] E. Visser. A family of syntax definition formalisms. In M. G. J. Brand-van den Brand et al., editors, *ASF+SDF'95. A Workshop on Generating Tools from Algebraic Specifications*, pages 89–126. Technical Report P9504, Programming Research Group, University of Amsterdam, May 1995.
- [86] E. Visser. Polymorphic syntax definition (extended abstract). In A. Nijholt, G. Scollo, and R. Steetskamp, editors, *Algebraic Methods in Language Processing (AMILP'95)*, volume 10 of *Twente Workshops in Language Technology*, pages 43–54, Enschede, The Netherlands, December 1995. Twente University of Technology.
- [87] E. Visser. A case study in optimizing parsing schemata by disambiguation filters. In *International Workshop on Parsing Technology (IWPT'97)*, pages 210–224, Boston, USA, September 1997. Massachusetts Institute of Technology.
- [88] E. Visser. Character classes. Technical Report P9708, Programming Research Group, University of Amsterdam, August 1997.
- [89] E. Visser. A family of syntax definition formalisms. Technical Report P9706, Programming Research Group, University of Amsterdam, July 1997.
- [90] E. Visser. From context-free grammars with priorities to character class grammars. In *Liber Amicorum Paul Klint*. CWI, Amsterdam, November 1997.
- [91] E. Visser. Scannerless generalized-LR parsing. Technical Report P9707, Programming Research Group, University of Amsterdam, July 1997.
- [92] E. Visser. *Syntax Definition for Language Prototyping*. PhD thesis, University of Amsterdam, September 1997.
- [93] E. Visser. Polymorphic syntax definition. *Theoretical Computer Science*, 199:57–86, 1998.
- [94] E. Visser. A bootstrapped compiler for strategies (extended abstract). In B. Gramlich, H. Kirchner, and F. Pfenning, editors, *Strategies in Automated Deduction (STRATEGIES'99)*, pages 73–83, Trento, Italy, July 5 1999.
- [95] E. 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.
- [96] E. Visser. Language independent traversals for program transformation. In J. Jeuring, editor, *Workshop on Generic Programming (WGP'00)*, Ponte de Lima, Portugal, July 2000. Technical Report UU-CS-2000-19, Department of Information and Computing Sciences, Universiteit Utrecht.
- [97] E. Visser. Scoped dynamic rewrite rules. In M. van den Brand and R. Verma, editors, *Rule Based Programming (RULE'01)*, volume 59/4 of *Electronic Notes in Theoretical Computer Science*. Elsevier Science Publishers, September 2001.

- [98] E. Visser. Stratego: A language for program transformation based on rewriting strategies. System description of Stratego 0.5. In A. Middeldorp, editor, *Rewriting Techniques and Applications (RTA'01)*, volume 2051 of *Lecture Notes in Computer Science*, pages 357–361. Springer-Verlag, May 2001.
- [99] E. Visser. A survey of rewriting strategies in program transformation systems. 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.
- [100] E. Visser. Meta-programming with concrete object syntax. In D. Batory, C. Consel, and W. 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.
- [101] E. Visser. *Tiger in Stratego: An Exercise in Compilation by Transformation*. Institute of Information and Computing Sciences, Utrecht University, 2002. Technical Documentation.
- [102] E. Visser. Stratego wiki / lift definitions to top level. <http://www.stratego-language.org/Stratego/LiftDefinitionsToTopLevel>, August 2003. Notes on component of the Stratego Compiler.
- [103] E. Visser. Program transformation with Stratego/XT: Rules, strategies, tools, and systems in StrategoXT-0.9. In C. Lengauer et al., editors, *Domain-Specific Program Generation*, volume 3016 of *Lecture Notes in Computer Science*, pages 216–238. Springer-Verlag, June 2004.
- [104] E. Visser. *Strategies for Program Transformation*. 2004. Draft book. Superseded by Stratego/XT manual.
- [105] E. Visser. *The Stratego/XT BibTeX Tools. Tool documentation*. Department of Information and Computing Sciences, Universiteit Utrecht, Utrecht, The Netherlands, 0.2 edition, November 2005.
- [106] E. 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.
- [107] E. Visser. Transformations for abstractions. In J. Krinke and G. 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).
- [108] E. Visser. *The Stratego/XT BibTeX Tools. Tool documentation*. Department of Information and Computing Sciences, Universiteit Utrecht, Utrecht, The Netherlands, 0.3pre edition, August 2006. (Draft).
- [109] E. Visser. Domain-specific language engineering. In R. Lämmel, J. Saraiva, and J. Visser, editors, *Generative and Transformational Techniques in Software Engineering (GTTSE 2007)*, pages 265–318, Braga, Portugal, July 2007. Universidade do Minho. International Summer School GTTSE 2007, Pre-Proceedings.

- [110] E. Visser. WebDSL: A case study in domain-specific language engineering. In *Generative and Transformational Techniques in Software Engineering (GTTSE 2007)*, Lecture Notes in Computer Science. Springer, 2008. Tutorial for International Summer School GTTSE 2007; to appear.
- [111] E. Visser and Z.-e.-A. Benaïssa. 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.
- [112] E. Visser, Z.-e.-A. Benaïssa, and A. 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.
- [113] J. Visser. Stratego + SDF. Stratego Users Day 2000.
- [114] D. G. Waddington and B. Yao. High fidelity C++ code transformation. In *Proceedings of the 5th workshop on Language Descriptions, Tools and Applications (LDTA 2005)*, Electronic Notes in Theoretical Computer Science, Edinburgh University, UK, April 3 2005.
- [115] H. Westra. Configurable transformations for high-quality automatic program improvement. CobolX: a case study. Master's thesis, Utrecht University, Utrecht, The Netherlands, February 2002.
- [116] J. Wijngaardenvan Wijngaarden and E. 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., May 2003.
- [117] <http://www.stratego-language.org/Stratego/StrategoDocumentation>.
- [118] <http://www.stratego-language.org/Tiger>.
- [119] <http://www.program-transformation.org/xt/>.