

Books and Book Chapters

- [1] M. G. J. van den Brand, A. van Deursen, T. B. Dinesh, J. F. T. Kamperman, and E. Visser, editors. *Proceedings of the Workshop on Generating Tools from Algebraic Specifications (ASF+SDF 1995)*. Technical Report P9504, Programming Research Group, University of Amsterdam, May 1995.
- [2] B. Fischer and E. Visser, editors. *Proceedings of the 2002 ACM SIGPLAN Workshop on Rule-Based Programming (RULE 2002)*, Pittsburgh, Pennsylvania, USA, October 5, 2002. Association for Computing Machinery.
- [3] 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.
- [4] K. T. Kalleberg and E. Visser, editors. *Proceedings of the Sixth Stratego Users Day*. Institute for Information and Computing Sciences, Utrecht University, May 2005. Technical Report UU-CS-2005-053.
- [5] G. Karsai and E. Visser, editors. *Proceedings of the Third International Conference on Generative Programming and Component Engineering (GPCE 2004)*, volume 3286 of *Lecture Notes in Computer Science*, Vancouver, Canada, October 2004. Springer.
- [6] G. Ramalingam and E. Visser, editors. *Proceedings of the 2007 ACM SIGPLAN Workshop on Partial Evaluation and Semantics-based Program Manipulation*, Nice, France, January 2007. ACM.
- [7] E. Visser. Multi-level specifications. In A. van Deursen, J. Heering, and P. Klint, editors, *Language Prototyping. An Algebraic Specification Approach*, volume 5 of *AMAST Series in Computing*, pages 105–196. World Scientific, Singapore, September 1996.
- [8] E. Visser. From context-free grammars with priorities to character class grammars. In *Liber Amicorum Paul Klint*. CWI, Amsterdam, November 1997.
- [9] E. Visser. *Syntax Definition for Language Prototyping*. PhD thesis, University of Amsterdam, September 1997.
- [10] E. Visser, editor. *Proceedings of the Second Stratego Users Day (SUD 2001)*, Utrecht, The Netherlands, February 2001. Institute for Information and Computing Sciences, Utrecht University.
- [11] 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.

Journal Articles

- [12] M. G. J. van den Brand and E. Visser. Generation of formatters for context-free languages. *ACM Transactions on Software Engineering and Methodology*, 5(1):1–41, January 1996.
- [13] M. Bravenboer, E. Dolstra, and E. Visser. Preventing injection attacks with syntax embeddings. A host and guest language independent approach. *Science of Computer Programming*, 2009. (in press).
- [14] 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*, 72(1-2):52–70, June 2008. Special issue on experimental software and toolkits.
- [15] 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.
- [16] Z. Hemel, D. M. Groenewegen, L. C. L. Kats, and E. Visser. Code generation by model transformation. A case study in transformation modularity. *Software and Systems Modeling*, 2009. (accepted modulo minor revision).
- [17] 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, February 2000.
- [18] E. Visser. Polymorphic syntax definition. *Theoretical Computer Science*, 199:57–86, 1998.
- [19] 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.

In Proceedings

- [20] 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 2003)*, pages 65–74, Amsterdam, The Netherlands, September 2003. IEEE Computer Society Press.
- [21] 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 2007)*, volume 203 of *Electronic Notes in Theoretical Computer Science*, pages 85–101, Braga, Portugal, March 2008. Elsevier.
- [22] M. G. J. van den Brand, J. Scheerder, J. Vinju, and E. Visser. Disambiguation filters for scannerless generalized LR parsers. In N. Horspool,

- editor, *Compiler Construction (CC 2002)*, volume 2304 of *Lecture Notes in Computer Science*, pages 143–158, Grenoble, France, April 2002. Springer-Verlag.
- [23] 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 2005)*, volume 4143 of *Lecture Notes in Computer Science*, pages 297–311, Braga, Portugal, 2006. Springer Verlag.
- [24] 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 2007)*, pages 3–12, New York, NY, USA, October 2007. ACM.
- [25] 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 2006)*, pages 209–228, Portland, Oregon, USA, October 2006. ACM Press.
- [26] 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 2005)*, volume 3676 of *Lecture Notes in Computer Science*, pages 157–172, Tallinn, Estonia, September 2005. Springer.
- [27] M. Bravenboer and E. Visser. Rewriting strategies for instruction selection. In S. Tison, editor, *Rewriting Techniques and Applications (RTA 2002)*, 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 2004)*, pages 365–383, Vancouver, Canada, October 2004. ACM Press.
- [29] M. Bravenboer and E. Visser. Designing syntax embeddings and assimilations for language libraries. In H. Giese, editor, *Models in Software Engineering. Workshops and Symposia at MoDELS 2007*, volume 5002 of *Lecture Notes in Computer Science*, pages 34–46, Heidelberg, 2008. Springer. Selected paper from the Workshop on Language Engineering (ATEM 2007).
- [30] M. Bravenboer and E. Visser. Parse table composition. separate compilation and binary extensibility of grammars. In D. Gasevic and E. van Wyk, editors, *Software Language Engineering (SLE 2008)*, volume 5452 of *Lecture Notes in Computer Science*, pages 74–94, Heidelberg, March 2009. Springer.

- [31] G. de Geest, S. D. Vermolen, A. van Deursen, and E. Visser. Generating version convertors for domain-specific languages. In A. Zaidman, M. D. Penta, and A. Hassan, editors, *Proceedings 15th Working Conference on Reverse Engineering (WCRE 2008)*, pages 197–201. IEEE Press, October 2008.
- [32] 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 2001)*, volume 44 of *Electronic Notes in Theoretical Computer Science*. Elsevier Science Publishers, April 2001.
- [33] E. Dolstra, M. Bravenboer, and E. Visser. Service configuration management. In J. E. James Whitehead and A. P. Dahlqvist, editors, *12th International Workshop on Software Configuration Management (SCM-12)*, pages 83–98, Lisbon, Portugal, September 2005. ACM.
- [34] E. Dolstra, M. de Jonge, and E. Visser. Nix: A safe and policy-free system for software deployment. In L. Damon, editor, *18th Large Installation System Administration Conference (LISA 2004)*, pages 79–92, Atlanta, Georgia, USA, November 2004. USENIX.
- [35] 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 2002)*, volume 65/3 of *Electronic Notes in Theoretical Computer Science*, Grenoble, France, April 2002. Elsevier Science Publishers.
- [36] E. Dolstra and E. Visser. The Nix build farm: A declarative approach to continuous integration. In *International Workshop on Advanced Software Development Tools and Techniques (WASDETT 2008)*, July 2008.
- [37] E. Dolstra, E. Visser, and M. de Jonge. Imposing a memory management discipline on software deployment. In J. Estublier and D. Rosenblum, editors, *26th International Conference on Software Engineering (ICSE 2004)*, pages 583–592, Edinburgh, Scotland, May 2004. IEEE Computer Society.
- [38] F. Duran, M. Roldán, E. Balland, M. van den Brand, S. Eker, K. T. Kalleberg, L. C. L. Kats, P.-E. Moreau, R. Schevchenko, and E. Visser. The second rewrite engines competition. In G. Rosu, editor, *Proceedings of the Seventh International Workshop on Rewriting Logic and its Applications (WRLA 2008)*, volume 238 of *Electronic Notes in Theoretical Computer Science*, pages 281–291. Elsevier, June 2009.
- [39] 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.
- [40] D. M. Groenewegen, Z. Hemel, L. C. L. Kats, and E. Visser. WebDSL: A domain-specific language for dynamic web applications. In N. Mielke and O. Zimmermann, editors, *Companion to the 23rd ACM SIGPLAN*

Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA 2008), pages 779–780, New York, NY, USA, October 2008. ACM. (poster).

- [41] D. M. Groenewegen, Z. Hemel, L. C. L. Kats, and E. Visser. When frameworks let you down. platform-imposed constraints on the design and evolution of domain-specific languages. In J. Gray, J. Sprinkle, J.-P. Tolvanen, and M. Rossi, editors, *Proceedings of the 8th OOPSLA Workshop on Domain Specific Modelling (DSM'08)*, pages 64–66, Nashville, Tennessee, USA, October 2008.
- [42] D. M. Groenewegen and E. Visser. Declarative access control for WebDSL: Combining language integration and separation of concerns. In D. Schwabe and F. Curbera, editors, *Eighth International Conference on Web Engineering (ICWE 2008)*, pages 175–188. IEEE CS Press, July 2008. best paper award.
- [43] Z. Hemel, L. C. L. Kats, and E. Visser. Code generation by model transformation. A case study in transformation modularity. In J. Gray, A. Pierantonio, and A. Vallecillo, editors, *Theory and Practice of Model Transformations. First International Conference on Model Transformation (ICMT 2008)*, volume 5063 of *Lecture Notes in Computer Science*, pages 183–198, Heidelberg, July 2008. Springer.
- [44] Z. Hemel, R. Verhaaf, and E. Visser. WebWorkFlow: An object-oriented workflow modeling language for web applications. In K. Czarnecki, I. Ober, J.-M. Bruel, A. Uhl, and M. Völter, editors, *Proceedings of the 11th International Conference on Model Driven Engineering Languages and Systems (MODELS 2008)*, volume 5301 of *Lecture Notes in Computer Science*, pages 113–127, Heidelberg, September 2008. Springer.
- [45] 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 2001)*, volume 57 of *Electronic Notes in Theoretical Computer Science*, Utrecht, The Netherlands, May 2001. Elsevier Science Publishers.
- [46] 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 2005)*, volume 147 of *Electronic Notes in Theoretical Computer Science*, pages 5–30, Nara, Japan, 2006. Elsevier Science Publishers.
- [47] K. T. Kalleberg and E. Visser. Fusing a transformation language with an open compiler. In A. Sloane and A. Johnstone, editors, *Proceedings of the Seventh Workshop on Language Descriptions, Tools, and Applications (LDTA 2007)*, volume 203 of *Electronic Notes in Theoretical Computer Science*, pages 21–36. Elsevier, April 2008.
- [48] L. C. L. Kats, M. Bravenboer, and E. Visser. Mixing source and bytecode. A case for compilation by normalization. In G. Kiczales, editor, *Proceedings of the 23rd ACM SIGPLAN Conference on Object-Oriented Programming,*

Systems, Languages, and Applications (OOPSLA 2008), pages 91–108, New York, NY, USA, October 2008. ACM.

- [49] L. C. L. Kats, M. de Jonge, E. Nilsson-Nyman, and E. Visser. Providing rapid feedback in generated modular language environments. adding error recovery to scannerless generalized-lr parsing. In G. T. Leavens, editor, *Proceedings of the 24th ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA 2009)*, ACM SIGPLAN Notices, New York, NY, USA, October 2009. ACM Press. (to appear).
- [50] L. C. L. Kats, K. T. Kalleberg, and E. Visser. Generating editors for embedded languages. integrating SGLR into IMP. In A. Johnstone and J. Vinju, editors, *Proceedings of the Eighth Workshop on Language Descriptions, Tools, and Applications (LDTA 2008)*, Budapest, Hungary, April 2008.
- [51] L. C. L. Kats, K. T. Kalleberg, and E. Visser. Domain-specific languages for composable editor plugins. In T. Ekman and J. Vinju, editors, *Proceedings of the Ninth Workshop on Language Descriptions, Tools, and Applications (LDTA 2009)*, Electronic Notes in Theoretical Computer Science. Elsevier Science Publishers, April 2009.
- [52] L. C. L. Kats, A. M. Sloane, and E. Visser. Decorated attribute grammars. Attribute evaluation meets strategic programming. In O. de Moor and M. I. Schwartzbach, editors, *International Conference on Compiler Construction (CC 2009)*, volume 5501 of *Lecture Notes in Computer Science*, pages 142–157, York, United Kingdom, March 2009. Springer.
- [53] 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.
- [54] R. Laemmel, E. Visser, and J. Visser. Strategic programming meets adaptive programming. In *Proceedings of Aspect-Oriented Software Development (AOSD 2003)*, pages 168–177, Boston, USA, March 2003. ACM Press.
- [55] 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 1997)*, Electronic Workshops in Computing, Berlin, November 1997. Springer-Verlag.
- [56] K. Olmos and E. Visser. Strategies for source-to-source constant propagation. In B. Gramlich and S. Lucas, editors, *Workshop on Reduction Strategies (WRS 2002)*, volume 70 of *Electronic Notes in Theoretical Computer Science*, page 20, Copenhagen, Denmark, July 2002. Elsevier Science Publishers.
- [57] 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 2003)*, pages 141–150, Amsterdam, The Netherlands, September 2003. IEEE Computer Society Press.

- [58] 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 2005)*, volume 3443 of *Lecture Notes in Computer Science*, pages 204–220. Springer-Verlag, April 2005.
- [59] A. M. Sloane, L. C. L. Kats, and E. Visser. A pure object-oriented embedding of attribute grammars. In T. Ekman and J. Vinju, editors, *Proceedings of the Ninth Workshop on Language Descriptions, Tools, and Applications (LDTA 2009)*, *Electronic Notes in Theoretical Computer Science*. Elsevier Science Publishers, April 2009. (to appear).
- [60] S. van der Burg, E. Dolstra, and E. Visser. Software deployment in a dynamic cloud: From device to service orientation in a hospital environment. In K. Bhattacharya, M. Bichler, and S. Tai, editors, *Workshop on Software Engineering Challenges of Cloud Computing (ICSE-Cloud'09)*, Vancouver, Canada, May 2009.
- [61] A. van Deursen and E. Visser. The reengineering wiki. In *Proceedings 6th European Conference on Software Maintenance and Reengineering (CSMR 2002)*., pages 217–220. IEEE Computer Society, 2002.
- [62] A. van Deursen, E. Visser, and J. Warmer. Model-driven software evolution: A research agenda. In D. Tamzalit, editor, *CSMR Workshop on Model-Driven Software Evolution (MoDSE 2007)*, pages 41–49, Amsterdam, The Netherlands, March 2007.
- [63] S. D. Vermolen and E. Visser. Heterogeneous coupled evolution of software languages. In K. Czarnecki, I. Ober, J.-M. Bruel, A. Uhl, and M. Völter, editors, *Proceedings of the 11th International Conference on Model Driven Engineering Languages and Systems (MODELS 2008)*, volume 5301 of *Lecture Notes in Computer Science*, pages 630–644, Heidelberg, September 2008. Springer.
- [64] E. Visser. A family of syntax definition formalisms. In M. G. J. van den Brand et al., editors, *ASF+SDF 1995. A Workshop on Generating Tools from Algebraic Specifications*, pages 89–126. Technical Report P9504, Programming Research Group, University of Amsterdam, May 1995.
- [65] E. Visser. A case study in optimizing parsing schemata by disambiguation filters. In *International Workshop on Parsing Technology (IWPT 1997)*, pages 210–224, Boston, USA, September 1997. Massachusetts Institute of Technology.
- [66] E. Visser. Strategic pattern matching. In P. Narendran and M. Rusinowitch, editors, *Rewriting Techniques and Applications (RTA 1999)*, volume 1631 of *Lecture Notes in Computer Science*, pages 30–44, Trento, Italy, July 1999. Springer-Verlag.
- [67] E. Visser. Language independent traversals for program transformation. In J. Jeuring, editor, *Workshop on Generic Programming (WGP 2000)*, Ponte de Lima, Portugal, July 2000. Technical Report UU-CS-2000-19, Department of Information and Computing Sciences, Universiteit Utrecht.

- [68] E. Visser. Scoped dynamic rewrite rules. In M. van den Brand and R. Verma, editors, *Rule Based Programming (RULE 2001)*, volume 59/4 of *Electronic Notes in Theoretical Computer Science*. Elsevier Science Publishers, September 2001.
- [69] 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 2001)*, volume 57 of *Electronic Notes in Theoretical Computer Science*, Utrecht, The Netherlands, May 2001. Elsevier Science Publishers.
- [70] E. Visser. Meta-programming with concrete object syntax. In D. Batory, C. Consel, and W. Taha, editors, *Generative Programming and Component Engineering (GPCE 2002)*, volume 2487 of *Lecture Notes in Computer Science*, pages 299–315, Pittsburgh, PA, USA, October 2002. Springer-Verlag.
- [71] E. Visser. Transformations for abstractions. In J. Krinke and G. Antoniol, editors, *Fifth IEEE International Workshop on Source Code Analysis and Manipulation (SCAM 2005)*, pages 3–12, Budapest, Hungary, October 2005. IEEE Computer Society Press. (Keynote paper).
- [72] E. Visser. WebDSL: A case study in domain-specific language engineering. In R. Lämmel, J. Visser, and J. Saraiva, editors, *International Summer School on Generative and Transformational Techniques in Software Engineering (GTTSE 2007)*, volume 5235 of *Lecture Notes in Computer Science*, pages 291–373, Heidelberg, October 2008. Springer.
- [73] 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 1998)*, volume 15 of *Electronic Notes in Theoretical Computer Science*, Pont-à-Mousson, France, September 1998. Elsevier Science Publishers.
- [74] 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 1998)*, pages 13–26. ACM Press, September 1998.

Theses

- [75] E. Visser. Combinatory algebraic specification & compilation of list matching. Master’s thesis, Department of Computer Science, University of Amsterdam, Amsterdam, June 1993.
- [76] E. Visser. *Syntax Definition for Language Prototyping*. PhD thesis, University of Amsterdam, September 1997.

Tool Descriptions

- [77] M. G. J. van den Brand, J. Heering, H. de Jong, M. de Jonge, T. Kuipers, P. Klint, L. Moonen, P. Olivier, J. Scheerder, J. Vinju, E. Visser, and

- J. Visser. The Asf+Sdf Meta-Environment: a component-based language laboratory. In R. Wilhelm, editor, *Compiler Construction (CC 2001)*, volume 2027 of *Lecture Notes in Computer Science*, pages 365–368, Genova, Italy, April 2001. Springer-Verlag.
- [78] M. G. J. van den Brand and E. Visser. The ASF+SDF Meta-environment: Documentation tools for free! In P. D. Mosses, M. Nielsen, and M. I. Schwartzbach, editors, *Theory and Practice of Software Development (TAPSOFT 1995)*, volume 915 of *Lecture Notes in Computer Science*, pages 803–804, Aarhus, Denmark, May 1995. Springer-Verlag. Tool demonstration.
- [79] 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 2006)*, pages 95–99, Charleston, South Carolina, January 2006. ACM SIGPLAN.
- [80] K. T. Kalleberg and E. Visser. Spoofox: An interactive development environment for program transformation with Stratego/XT. In A. Sloane and A. Johnstone, editors, *Proceedings of the Seventh Workshop on Language Descriptions, Tools, and Applications (LDTA 2007)*, pages 47–50, Braga, Portugal, March 2007.
- [81] 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 2001)*, volume 2051 of *Lecture Notes in Computer Science*, pages 357–361. Springer-Verlag, May 2001.

Position Papers

- [82] O. S. Bagge, M. Haverdaen, and E. Visser. A C++ transformation framework (extended abstract). In *Proceedings of the Nordic Workshop on Programming Theory (NWPT 2000)*, Bergen, Norway, October 2000.
- [83] M. Bravenboer and E. Visser. Reusable and adaptable strategies for generative programming. In M. Haverdaen et al., editors, *GPCE Software Transformation Systems Workshop (STSW 2004)*, Vancouver, Canada, October 2004. Position paper.
- [84] E. Dolstra, G. Florijn, M. de Jonge, and E. Visser. Capturing timeline variability with transparent configuration environments. In J. Bosch and P. Knauber, editors, *IEEE Workshop on Software Variability Management (SVM 2003)*, Portland, Oregon, May 2003. IEEE.
- [85] E. Dolstra, G. Florijn, and E. Visser. Timeline variability: The variability of binding time of variation points. In *Workshop on Software Variability Modeling (SVM 2003)*, number IWI preprint 2003-7-01, Groningen, The Netherlands, 2003. Reseach Institute of Computer Science and Mathematics, University of Groningen.

- [86] E. Visser. A bootstrapped compiler for strategies (extended abstract). In B. Gramlich, H. Kirchner, and F. Pfenning, editors, *Strategies in Automated Deduction (STRATEGIES 1999)*, pages 73–83, Trento, Italy, July 5 1999.

Non-refereed Journal Articles

- [87] E. Dolstra, G. Florijn, M. de Jonge, and E. Visser. Transparantie in variabiliteit: Heldere keuzes in een eenduidige configuratieomgeving. *Informatie*, 46(1):18–21, Feb. 2004.

Technical Reports

- [88] 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.
- [89] M. G. J. van den Brand and E. Visser. From Box to \TeX : An algebraic approach to the generation of documentation tools. Technical Report P9420, Programming Research Group, University of Amsterdam, July 1994.
- [90] 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.
- [91] M. de Jonge, E. Visser, and J. Visser. Collaborative software development. Technical Report SEN-R0113, CWI, Amsterdam, The Netherlands, 2001.
- [92] 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.
- [93] 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.
- [94] E. Visser. Solving type equations in multi-level specifications (preliminary version). Technical Report P9606, Programming Research Group, University of Amsterdam, Amsterdam, May 1996.
- [95] E. Visser. Character classes. Technical Report P9708, Programming Research Group, University of Amsterdam, August 1997.
- [96] E. Visser. A family of syntax definition formalisms. Technical Report P9706, Programming Research Group, University of Amsterdam, July 1997.
- [97] E. Visser. Scannerless generalized-LR parsing. Technical Report P9707, Programming Research Group, University of Amsterdam, July 1997.

- [98] J. van 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.

Documentation

- [99] 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.
- [100] 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.

Unpublished Documents

- [101] E. Visser. Syntax and static semantics of Eiffel. A case study in algebraic specification techniques. Unpublished technical report, December 1992.