

Bibliography

- [Aasa et al. 1988] Aasa, A., Petersson, K., & Synek, D. (1988). Concrete syntax for data objects in functional languages. In *Proceedings of the 1988 ACM Conference on LISP and Functional Programming*, (pp. 96–105). ACM Press. (Cited on page 61.)
- [Aho & Johnson 1974] Aho, A. V. & Johnson, S. C. (1974). LR parsing. *ACM Computing Surveys*, 6(2), 99–124. (Cited on page 154.)
- [Aho et al. 1975] Aho, A. V., Johnson, S. C., & Ullman, J. D. (1975). Deterministic parsing of ambiguous grammars. *Communications of the ACM*, 18(8), 441–452. (Cited on pages 184 and 188.)
- [Aho et al. 1986] Aho, A. V., Sethi, R., & Ullman, J. (1986). *Compilers: Principles, Techniques, and Tools*. Reading, Massachusetts: Addison Wesley. (Cited on pages 154 and 158.)
- [Aldrich 2005] Aldrich, J. (2005). Open modules: Modular reasoning about advice. In *Proceedings of the European Conference on Object-Oriented Programming (ECOOP'05)*, volume 3586 of *LNCS*, (pp. 144–168). Springer. (Cited on page 130.)
- [Allan et al. 2005] Allan, C., Avgustinov, P., Christensen, A. S., Hendren, L., Kuzins, S., Lhoták, O., de Moor, O., Sereni, D., Sittampalam, G., & Tibble, J. (2005). Adding trace matching with free variables to AspectJ. In *Proceedings of the 20th ACM SIGPLAN Conference on Object-Oriented Programming Systems, Languages and Applications (OOPSLA 2005)*, (pp. 345–364)., San Diego, California, USA. ACM Press. (Cited on page 94.)
- [Anley 2002a] Anley, C. (2002a). Advanced SQL injection. http://www.ngssoftware.com/papers/advanced_sql_injection.pdf. (Cited on page 90.)
- [Anley 2002b] Anley, C. (2002b). (more) Advanced SQL injection. http://www.ngssoftware.com/papers/more_advanced_sql_injection.pdf. (Cited on page 90.)
- [Arnoldus et al. 2007] Arnoldus, B. J., Bijpost, J. W., & van den Brand, M. G. J. (2007). Repleo: A syntax-safe template engine. In proceedings GPCE 2007 [Lawall 2007]. (Cited on page 148.)
- [ASF+SDF MetaEnv] ASF+SDF Meta-Environment website. <http://www.meta-environment.org>. (Cited on page 36.)
- [AspectJ] AspectJ. *AspectJ Documentation*. With links to the AspectJ Programming Guide and the AspectJ 5 Developer's Notebook. (Cited on page 99.)

- [Avgustinov et al. 2005] Avgustinov, P., Christensen, A. S., Hendren, L., Kuzins, S., Lhoták, J., Lhoták, O., de Moor, O., Sereni, D., Sittampalam, G., & Tibble, J. (2005). abc: an extensible AspectJ compiler. In *Proceedings of the 4th International Conference on Aspect-Oriented Software Development (AOSD'04)*, (pp. 87–98)., New York, NY, USA. ACM Press. (Cited on pages 94 and 101.)
- [Avgustinov et al. 2007] Avgustinov, P., Hajiyev, E., Ongkingco, N., de Moor, O., Sereni, D., Tibble, J., & Verbaere, M. (2007). Semantics of static pointcuts in AspectJ. In Felleisen, M. (Ed.), *POPL '07: Proceedings of the 34th annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, (pp. 11–23)., New York, NY, USA. ACM Press. (Cited on page 142.)
- [Aycock & Horspool 1999] Aycock, J. & Horspool, R. N. (1999). Faster generalized LR parsing. In *Proceedings of 8th International Conference on Compiler Construction (CC'99)*, volume 1575, (pp. 32–46)., Amsterdam. Springer-Verlag. (Cited on pages 141 and 206.)
- [Bachrach & Playford 2001] Bachrach, J. & Playford, K. (2001). The Java syntactic extender (JSE). In *Proceedings of the 16th ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA 2001)*, (pp. 31–42). ACM Press. (Cited on pages 42 and 61.)
- [Baker & Hsieh 2002] Baker, J. & Hsieh, W. (2002). Maya: multiple-dispatch syntax extension in Java. In *PLDI '02: Proceedings of the ACM SIGPLAN 2002 Conference on Programming Language Design and Implementation*, (pp. 270–281). ACM Press. (Cited on pages 62 and 181.)
- [Batory et al. 1998] Batory, D., Lofaso, B., & Smaragdakis, Y. (1998). JTS: tools for implementing domain-specific languages. In *Proceedings Fifth International Conference on Software Reuse (ICSR'98)*, (pp. 143–153). IEEE Computer Society Press. (Cited on pages 6, 11, 39, 47, 48, 61, 67, 73, 85, 140, and 178.)
- [Baxter et al. 2004] Baxter, I. D., Pidgeon, C., & Mehlich, M. (2004). DMS®: Program transformations for practical scalable software evolution. In proceedings ICSE 2004 [Estublier & Rosenblum 2004], (pp. 625–634). (Cited on pages 48 and 61.)
- [BCEL Website] Byte code engineering library (BCEL). <http://jakarta.apache.org/bcel/>. (Cited on page 44.)
- [Begel & Graham 2004] Begel, A. & Graham, S. L. (2004). Language analysis and tools for input stream ambiguities. In proceedings LDTA 2004 [Hedin & Wyk 2004], (pp. 75–96). (Cited on pages 11, 38, and 139.)
- [Bierman et al. 2005] Bierman, G., Meijer, E., & Schulte, W. (2005). The essence of data access in C ω . In *ECOOP 2005 - Object-Oriented Programming, 19th European Conference*, volume 3586 of LNCS, (pp. 287–311). Springer. (Cited on page 88.)

- [Blasband 2001] Blasband, D. (2001). Parsing in a hostile world. In *Proceedings of the Eighth Working Conference on Reverse Engineering (WCRE'01)*, (pp. 291)., Washington, DC, USA. IEEE Computer Society. (Cited on page 139.)
- [Bodden & Stolz] Bodden, E. & Stolz, V. J-LO, the Java Logical Observer. <http://www-i2.informatik.rwth-aachen.de/Research/RV/JLO/>. (Cited on page 94.)
- [Borghini et al. 2006] Borghini, A., David, V., & Demaille, A. (2006). C-Transformers: a framework to write C program transformations. *Crossroads*, 12(3), 3-3. (Cited on pages 184 and 197.)
- [Bouma] Bouma, F. LLBLGen Pro, the n-tier generator and O/R mapper for .NET. <http://www.11blgen.com>. (Cited on page 13.)
- [Bourret 2007] Bourret, R. (2007). XML data binding resources. <http://www.rpbourret.com/xml/XMLDataBinding.htm>. (Cited on page 12.)
- [Bouwers et al. 2007] Bouwers, E., Bravenboer, M., & Visser, E. (2007). Grammar engineering support for precedence rule recovery and compatibility checking. In Johnstone, A. & Sloane, T. (Eds.), *Proceedings of the Seventh Workshop on Language Descriptions, Tools and Applications (LDTA'07)*, Electronic Notes in Theoretical Computer Science. Elsevier Science Publishers. (Cited on page 8.)
- [Brabrand & Schwartzbach 2002] Brabrand, C. & Schwartzbach, M. I. (2002). Growing languages with metamorphic syntax macros. In *Proceedings of the 2002 ACM SIGPLAN Workshop on Partial Evaluation and Semantics-based Program Manipulation (PEPM'02)*, (pp. 31-40). ACM Press. (Cited on pages 11, 41, 42, and 47.)
- [Brabrand et al. 2003] Brabrand, C., Schwartzbach, M. I., & Vanggaard, M. (2003). The Metafront system: Extensible parsing and transformation. In proceedings LDTA 2003 [Bryant & Saraiva 2003]. (Cited on page 42.)
- [Bracha & Cook 1990] Bracha, G. & Cook, W. (1990). Mixin-based inheritance. In *Proceedings of the European Conference on Object-Oriented Programming on Object-Oriented Programming systems, Languages, and Applications (OOPSLA/ECOOP '90)*, (pp. 303-311)., New York, NY, USA. ACM Press. (Cited on pages 124 and 152.)
- [Bracha et al. 1998] Bracha, G., Odersky, M., Stoutamire, D., & Wadler, P. (1998). GJ specification. (Cited on page 106.)
- [van den Brand et al. 2000] van den Brand, M. G. J., de Jong, H., Klint, P., & Olivier, P. (2000). Efficient annotated terms. *Software: Practice & Experience*, 30(3), 259-291. (Cited on pages 13, 29, 32, and 39.)
- [van den Brand et al. 2001] van den Brand, M. G. J., Heering, J., de Jong, H., de Jonge, M., Kuipers, T., Klint, P., Moonen, L., Olivier, P., Scheerder, J.,

- Vinju, J., Visser, E., & Visser, J. (2001). The ASF+SDF Meta-Environment: a component-based language laboratory. In Wilhelm, R. (Ed.), *Compiler Construction, 10th International Conference, CC 2001*, volume 2027 of LNCS, (pp. 365–368). Springer. (Cited on pages 37 and 148.)
- [van den Brand et al. 2003] van den Brand, M. G. J., Klusener, S., Moonen, L., & Vinju, J. (2003). Generalized parsing and term rewriting - semantics directed disambiguation. In proceedings LDTA 2003 [Bryant & Saraiva 2003]. (Cited on page 62.)
- [van den Brand et al. 2005] van den Brand, M. G. J., Moreau, P., & Vinju, J. (2005). A generator of efficient strongly typed abstract syntax trees in Java. *IEE Proceedings - Software*, 152(2), 70–78. (Cited on page 52.)
- [van den Brand & Perigot 2001] van den Brand, M. G. J. & Perigot, D. (Eds.). (2001). *Proceedings of LDTA'01, First Workshop on Language Descriptions, Tools and Applications*, volume 44 of *Electronic Notes in Theoretical Computer Science*. Elsevier Science Publishers. (Cited on pages 216 and 220.)
- [van den Brand & Ringeissen 2000] van den Brand, M. G. J. & Ringeissen, C. (2000). ASF+SDF parsing tools applied to ELAN. In *Third International Workshop on Rewriting Logic and Applications*, ENTCS. (Cited on page 60.)
- [van den Brand et al. 2002] van den Brand, M. G. J., Scheerder, J., Vinju, J. J., & Visser, E. (2002). Disambiguation filters for scannerless generalized LR parsers. In Horspool, R. N. (Ed.), *Compiler Construction, 11th International Conference, CC 2002*, volume 2304 of LNCS, (pp. 143–158). Springer. (Cited on pages 12, 30, 31, 32, 37, 48, 60, 62, 95, 99, 120, 137, and 189.)
- [Bravenboer et al. 2006] Bravenboer, M., de Groot, R., & Visser, E. (2006). MetaBorg in action: Examples of domain-specific language embedding and assimilation using Stratego/XT. In Lämmel, R. & Saraiva, J. (Eds.), *Proceedings of the Summer School on Generative and Transformational Techniques in Software Engineering (GTTSE'05)*, LNCS, Braga, Portugal. Springer. (Cited on page 6.)
- [Bravenboer et al. 2007 (Chapter 4)] Bravenboer, M., Dolstra, E., & Visser, E. (2007). Preventing injection attacks with syntax embedding – a host and guest language independent approach. In proceedings GPCE 2007 [Lawall 2007]. (Cited on pages 6, 8, and 150.)
- [Bravenboer et al. 2008] Bravenboer, M., Kalleberg, K. T., Vermaas, R., & Visser, E. (2008). Stratego/XT 0.17. A language and toolset for program transformation. *Science of Computer Programming. Special Issue on Experimental Software and Toolkits (EST)*. (To appear). (Cited on page 147.)
- [Bravenboer et al. 2006 (Chapter 5)] Bravenboer, M., Tanter, E., & Visser, E. (2006). Declarative, formal, and extensible syntax definition for AspectJ – A case for scannerless generalized-LR parsing. In *Proceedings of the 21st ACM*

- SIGPLAN Conference on Object-Oriented Programming Systems, Languages, and Applications (OOPSLA 2006)*, New York, NY, USA. ACM Press. (Cited on pages 6, 8, 73, 152, 153, 175, and 176.)
- [Bravenboer et al. 2005 (Chapter 3)] Bravenboer, M., Vermaas, R., Vinju, J. J., & Visser, E. (2005). Generalized type-based disambiguation of meta programs with concrete object syntax. In proceedings GPCE 2005 [Glück & Lowry 2005], (pp. 157–172). (Cited on pages 6, 8, 67, 73, 86, 115, and 137.)
- [Bravenboer & Visser 2004 (Chapter 2)] Bravenboer, M. & Visser, E. (2004). Concrete syntax for objects. Domain-specific language embedding and assimilation without restrictions. In Schmidt, D. C. (Ed.), *Proceedings of the 19th ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA'04)*, (pp. 365–383). ACM Press. (Cited on pages 6, 8, 50, 61, 69, 73, 78, 90, 115, 137, 142, 146, 149, 152, 175, and 176.)
- [Bryant & Saraiva 2003] Bryant, B. & Saraiva, J. (Eds.). (2003). *Proceedings of the Third Workshop on Language Descriptions, Tools and Applications (LDTA'03)*, volume 82 of *Electronic Notes in Theoretical Computer Science*. Elsevier Science Publishers. (Cited on pages 213 and 214.)
- [Buehrer et al. 2005] Buehrer, G. T., Weide, B. W., & Sivilotti, P. A. (2005). Using parse tree validation to prevent SQL injection attacks. In *Fifth International Workshop on Software Engineering and Middleware (SEM 2005)*. ACM Press. (Cited on page 89.)
- [Cardelli 1993] Cardelli, L. (1993). An implementation of $F_{<}$. SRC Research Report 97, Digital Equipment Corporation, Systems Research Center. (Cited on page 37.)
- [Cardelli et al. 1994] Cardelli, L., Matthes, F., & Abadi, M. (1994). Extensible syntax with lexical scoping. SRC Research Report 121, Digital Systems Research Center, Palo Alto, California. (Cited on pages 11, 37, 146, and 181.)
- [Castor Website] Castor databinding framework for Java. <http://www.castor.org>. (Cited on page 13.)
- [Charles 1991] Charles, P. (1991). *A Practical Method for Constructing Efficient LALR(k) Parsers with Automatic Error Recovery*. PhD thesis, New York University. (Cited on page 206.)
- [Christensen et al. 2003] Christensen, A. S., Møller, A., & Schwartzbach, M. I. (2003). Precise analysis of string expressions. In *Static Analysis Symposium (SAS 2003)*, volume 2694 of *LNCS*, (pp. 1–18). Springer. (Cited on page 89.)
- [Cocoon Website] The Apache Cocoon project. <http://cocoon.apache.org>. (Cited on page 15.)
- [Cook & Rai 2005] Cook, W. R. & Rai, S. (2005). Safe query objects: Statically typed objects as remotely executable queries. In proceedings ICSE 2005 [Roman et al. 2005], (pp. 97–106). (Cited on pages 69 and 88.)

- [Cordy et al. 1991] Cordy, J., Halpern-Hamu, C., & Promislow, E. (1991). TXL: A rapid prototyping system for programming language dialects. *Computer Languages*, 16(1), 97–107. (Cited on pages 48 and 61.)
- [C++ Manual] The C preprocessor manual. <http://gcc.gnu.org/onlinedocs/cpp/>. (Cited on page 42.)
- [de Jonge & Monajemi 2001] de Jonge, M. & Monajemi, R. (2001). Cost-effective maintenance tools for proprietary languages. In *Proceedings International Conference on Software Maintenance (ICSM 2001)*, (pp. 240–249)., Los Alamitos, CA, USA. IEEE Computer Society Press. (Cited on page 199.)
- [de Jonge et al. 2001] de Jonge, M., Visser, E., & Visser, J. (2001). XT: A bundle of program transformation tools. In *proceedings LDTA 2001* [van den Brand & Perigot 2001]. (Cited on pages 12 and 37.)
- [de Rauglaudre 2003] de Rauglaudre, D. (2003). *Camlp4 Reference Manual*. (Cited on pages 62 and 181.)
- [DeRemer 1971] DeRemer, F. (1971). Simple LR(k) grammars. *Communications of the ACM*, 14(7), 453–460. (Cited on page 167.)
- [DeRemer & Pennello 1979] DeRemer, F. & Pennello, T. J. (1979). Efficient computation of LALR(1) look-ahead sets. In *Proceedings of the 1979 SIGPLAN symposium on Compiler Construction (CC 1979)*, (pp. 176–187)., New York, NY, USA. ACM Press. (Cited on page 171.)
- [van Deursen et al. 1996] van Deursen, A., Heering, J., & Klint, P. (Eds.). (1996). *Language Prototyping. An Algebraic Specification Approach*, volume 5 of *AMAST Series in Computing*. Singapore: World Scientific. (Cited on pages 37, 48, and 60.)
- [Earley 1970] Earley, J. (1970). An efficient context-free parsing algorithm. *Communications of the ACM*, 13(2), 94–102. (Cited on pages 34 and 180.)
- [Ekman & Hedin 2004] Ekman, T. & Hedin, G. (2004). Rewritable reference attributed grammars. In Odersky, M. (Ed.), *Proceedings of 18th European Conference on Object-Oriented Programming (ECOOP 2004)*, volume 3086 of *LNCS*, (pp. 144–169)., Oslo, Norway. Springer. (Cited on pages 146 and 152.)
- [Estublier & Rosenblum 2004] Estublier, J. & Rosenblum, D. S. (Eds.). (2004). *Proceedings of the 26th International Conference on Software Engineering (ICSE 2004)*, Washington, DC, USA. IEEE Computer Society Press. (Cited on pages 212 and 217.)
- [Eve & Kurki-Suonio 1966] Eve, J. & Kurki-Suonio, R. (1966). On computing the transitive closure of a relation. *Acta Informatica*, 8(4), 303–314. (Cited on page 171.)

- [Filman 2006] Filman, R. (Ed.). (2006). *Proceedings of the 5th International Conference on Aspect-Oriented Software Development (AOSD 2006)*, New York, NY, USA. ACM Press. (Cited on pages 218 and 222.)
- [Fischer & Visser 2004] Fischer, B. & Visser, E. (2004). Retrofitting the Auto-Bayes program synthesis system with concrete object syntax. In proceedings DSPG 2004 [Lengauer et al. 2004], (pp. 239–253). (Cited on page 37.)
- [Freeman & Pryce 2006] Freeman, S. & Pryce, N. (2006). Evolving an embedded domain-specific language in Java. In *OOPSLA '06: Companion to the 21st ACM SIGPLAN Conference on Object-Oriented Programming Systems, Languages, and Applications*, (pp. 855–865)., New York, NY, USA. ACM Press. (Cited on page 4.)
- [Gagnon & Hendren 1998] Gagnon, E. M. & Hendren, L. J. (1998). SableCC, an object-oriented compiler framework. In *Proceedings of the Technology of Object-Oriented Languages and Systems (TOOLS'98)*, (pp. 140–154). IEEE Computer Society. See also <http://www.sablecc.org>. (Cited on page 29.)
- [Glück & Lowry 2005] Glück, R. & Lowry, M. (Eds.). (2005). *Generative Programming and Component Engineering: 4th International Conference (GPCE 2005)*, volume 3676 of LNCS, Tallinn, Estonia. Springer. (Cited on pages 215 and 224.)
- [GNU M4 Website] GNU m4. <http://www.gnu.org/software/m4/>. (Cited on page 42.)
- [Gould et al. 2004a] Gould, C., Su, Z., & Devanbu, P. (2004a). JDBC checker: A static analysis tool for SQL/JDBC applications. In proceedings ICSE 2004 [Estublier & Rosenblum 2004], (pp. 697–698). (Cited on pages 69 and 88.)
- [Gould et al. 2004b] Gould, C., Su, Z., & Devanbu, P. (2004b). Static checking of dynamically generated queries in database applications. In proceedings ICSE 2004 [Estublier & Rosenblum 2004], (pp. 645–654). (Cited on pages 69 and 88.)
- [Grimm 2006] Grimm, R. (2006). Better extensibility through modular syntax. In Cook, W. R. (Ed.), *Proceedings of the ACM SIGPLAN 2006 Conference on Programming Language Design and Implementation (PLDI 2006)*. ACM Press. (Cited on page 178.)
- [Grune & Jacobs 1990] Grune, D. & Jacobs, C. J. H. (1990). *Parsing Techniques - A Practical Guide*. Upper Saddle River, NJ, USA: Ellis Horwood. (Cited on page 157.)
- [Halfond & Orso 2005] Halfond, W. G. & Orso, A. (2005). AMNESIA: Analysis and Monitoring for NEutralizing SQL-Injection Attacks. In *20th IEEE/ACM International Conference on Automated Software Engineering (ASE 2005)*, (pp. 174–183)., Long Beach, California, USA. (Cited on pages 66, 69, and 89.)

- [Halfond et al. 2006] Halfond, W. G., Orso, A., & Manolios, P. (2006). Using positive tainting and syntax-aware evaluation to counter SQL injection attacks. In *Foundations of Software Engineering (FSE 14)*. (Cited on pages 69 and 89.)
- [Halfond et al. 2006] Halfond, W. G., Viegas, J., & Orso, A. (2006). A classification of SQL-injection attacks and countermeasures. In *Proceedings of the International Symposium on Secure Software Engineering (ISSSE 2006)*. (Cited on pages 65 and 81.)
- [Harbulot & Gurd 2006] Harbulot, B. & Gurd, J. (2006). A join point for loops in AspectJ. In proceedings AOSD 2006 [Filman 2006]. (Cited on page 94.)
- [Hedin 2003] Hedin, G. (Ed.). (2003). *Compiler Construction, 12th International Conference, CC 2003*, volume 2622 of LNCS. Springer. (Cited on page 222.)
- [Hedin & Wyk 2004] Hedin, G. & Wyk, E. V. (Eds.). (2004). *Proceedings of the Fourth Workshop on Language Descriptions, Tools and Applications (LDTA'04)*, volume 110 of *Electronic Notes in Theoretical Computer Science*. Elsevier Science Publishers. (Cited on pages 212 and 219.)
- [Heering et al. 1989] Heering, J., Hendriks, P. R. H., Klint, P., & Rekers, J. (1989). The syntax definition formalism SDF – reference manual. *SIGPLAN Notices*, 24(11), 43–75. (Cited on pages 37, 39, 137, 139, 185, 189, and 190.)
- [Heering et al. 1990] Heering, J., Klint, P., & Rekers, J. (1990). Incremental generation of parsers. *IEEE Transactions on Software Engineering*, 16(12), 1344–1351. (Cited on pages 146 and 180.)
- [Hendren et al. 2004] Hendren, L., de Moor, O., Christensen, A. S., & the abc team (2004). The abc scanner and parser, including an LALR(1) grammar for AspectJ. Techrep, Programming Tools Group, Oxford University and the Sable research group, McGill University. (Cited on pages 94, 101, and 114.)
- [Hopcroft et al. 2006] Hopcroft, J. E., Motwani, R., & Ullman, J. D. (2006). *Introduction to Automata Theory, Languages, and Computation (3rd Edition)*. Boston, MA, USA: Addison-Wesley. (Cited on pages 72, 157, and 158.)
- [Horspool 1990] Horspool, R. N. (1990). Incremental generation of LR parsers. *Computer Languages*, 15(4), 205–223. (Cited on pages 146, 177, and 180.)
- [Hosoya & Pierce 2000] Hosoya, H. & Pierce, B. C. (2000). XDuce: A typed XML processing language. In *The World Wide Web and Databases, Third International Workshop WebDB 2000, Selected Papers*, volume 1997 of *Lecture Notes in Computer Science*, (pp. 226–244). Springer. (Cited on pages 4 and 45.)
- [Huang et al. 2004] Huang, Y.-W., Yu, F., Hang, C., Tsai, C.-H., Lee, D.-T., & Kuo, S.-Y. (2004). Securing web application code by static analysis and runtime protection. In *13th International World Wide Web Conference (WWW2004)*, (pp. 40–52). ACM Press. (Cited on pages 66 and 88.)

- [Hudak 1996] Hudak, P. (1996). Building domain-specific embedded languages. *ACM Computing Surveys*, 28(4es). (Cited on page 11.)
- [Hutton 1992] Hutton, G. (1992). Higher-order functions for parsing. *Journal of Functional Programming*, (2(3)), 323–343. (Cited on page 140.)
- [ISO 1992] ISO (1992). *ISO/IEC 9075:1992: Database Language SQL*. (Cited on pages 4, 67, and 90.)
- [JavaCC Website] Java Compiler Compiler (JavaCC). <https://javacc.dev.java.net/>. (Cited on pages 29 and 39.)
- [JAXB Website] Java architecture for XML binding JAXB. <https://jaxb.dev.java.net>. (Cited on page 13.)
- [JDT Website] Eclipse Java Development Tools (JDT) website. <http://www.eclipse.org/jdt/>. (Cited on pages 51, 59, and 102.)
- [Jeffery 2003] Jeffery, C. L. (2003). Generating LR syntax error messages from examples. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, 25(5), 631–640. (Cited on pages 141 and 206.)
- [Johnson 1975] Johnson, S. C. (1975). YACC—yet another compiler-compiler. Technical Report CS-32, AT&T Bell Laboratories, Murray Hill, N.J. (Cited on pages 184, 185, and 188.)
- [Johnstone & Scott 2002] Johnstone, A. & Scott, E. (2002). Generalised reduction modified LR parsing for domain specific language prototyping. In *35th Annual Hawaii International Conference on System Sciences (HICSS'02)*, (pp. 282), Washington, DC, USA. IEEE Computer Society Press. (Cited on page 157.)
- [Johnstone et al. 2004] Johnstone, A., Scott, E., & Economopoulos, G. (2004). The grammar tool box: A case study comparing GLR parsing algorithms. In proceedings LDTA 2004 [Hedin & Wyk 2004], (pp. 97–113). (Cited on page 207.)
- [Johnstone et al. 2006] Johnstone, A., Scott, E., & Economopoulos, G. (2006). Evaluating GLR parsing algorithms. *Science of Computer Programming*, 61(3), 228–244. (Cited on page 169.)
- [de Jong & Olivier 2004] de Jong, H. & Olivier, P. (2004). Generation of abstract programming interfaces from syntax definitions. *Journal of Logic and Algebraic Programming*, 59(1-2), 35–61. See also <http://www.cwi.nl/htbin/sen1/twiki/bin/view/SEN1/ApiGen>. (Cited on pages 13 and 29.)
- [Kats 2007] Kats, L. (2007). Supporting language extension and separate compilation by mixing Java and bytecode. Master's thesis, Utrecht University, Utrecht, The Netherlands. (Cited on pages 25 and 44.)

- [Kiczales et al. 2001] Kiczales, G., Hilsdale, E., Hugunin, J., Kersten, M., Palm, J., & Griswold, W. G. (2001). An overview of AspectJ. In Lindskov Knudsen, J. (Ed.), *ECOOP 2001: Object-Oriented Programming: 15th European Conference*, volume 2072 of *LNCS*, (pp. 327–353). Springer. (Cited on page 94.)
- [Kirby et al. 1998] Kirby, G., Morrison, R., & Stemple, D. (1998). Linguistic reflection in Java. *Software: Practice & Experience*, 28(10), 1045–1077. (Cited on pages 43 and 44.)
- [Klint et al. 2005] Klint, P., Lämmel, R., & Verhoef, C. (2005). Toward an engineering discipline for grammarware. *ACM Transactions on Software Engineering Methodology*, 14(3), 331–380. (Cited on pages 184 and 199.)
- [Klint & Visser 1994] Klint, P. & Visser, E. (1994). Using filters for the disambiguation of context-free grammars. In Pighizzini, G. & San Pietro, P. (Eds.), *Proceedings of ASMICS Workshop on Parsing Theory*, (pp. 1–20). Tech. Rep. 126, Università di Milano. (Cited on page 62.)
- [Knuth 1965] Knuth, D. E. (1965). On the translation of languages from left to right. *Information and Control*, 8(6), 607–639. (Cited on pages 154 and 192.)
- [Kolbly 2002] Kolbly, D. M. (2002). *Extensible Language Implementation*. PhD thesis, University of Texas at Austin. (Cited on page 180.)
- [Kort et al. 2002] Kort, J., Lämmel, R., & Verhoef, C. (2002). The grammar deployment kit – system demonstration. In van den Brand, M. G. J. & Lämmel, R. (Eds.), *Proceedings of LDTA 2002, Second Workshop on Language Descriptions, Tools and Applications*, volume 65 of *Electronic Notes in Theoretical Computer Science*, (pp. 117–123). Elsevier Science Publishers. (Cited on pages 184 and 199.)
- [Kuipers & Visser 2001] Kuipers, T. & Visser, J. (2001). Object-oriented tree traversal with JForester. In proceedings LDTA 2001 [van den Brand & Perigot 2001]. (Cited on page 29.)
- [Lämmel 2001a] Lämmel, R. (2001a). Grammar Adaptation. In *Formal Methods Europe (FME) 2001*, volume 2021 of *LNCS*, (pp. 550–570). Springer. (Cited on pages 184 and 199.)
- [Lämmel 2001b] Lämmel, R. (2001b). Grammar testing. In *Fundamental Approaches to Software Engineering, 4th International Conference, FASE 2001*, volume 2029 of *LNCS*, (pp. 201–216). Springer. (Cited on page 200.)
- [Lämmel & Verhoef 2001] Lämmel, R. & Verhoef, C. (2001). Semi-automatic grammar recovery. *Software: Practice & Experience*, 31(15), 1395–1438. (Cited on pages 184 and 199.)
- [Lawall 2007] Lawall, J. (Ed.). (2007). *GPCE '07: Proceedings of the 6th International Conference on Generative Programming and Component Engineering*, New York, NY, USA. ACM Press. (Cited on pages 211, 214, and 225.)

- [Lea 2000] Lea, D. (2000). *Concurrent Programming in Java, Second Edition, Design Principles and Patterns*. Addison Wesley Longman Publishing Co., Inc. (Cited on page 44.)
- [Leavenworth 1966] Leavenworth, B. M. (1966). Syntax macros and extended translation. *Communications of the ACM*, 9(11), 790–793. (Cited on pages 11, 40, 41, and 61.)
- [Lehman 1980] Lehman, M. M. (1980). On understanding laws, evolution, and conservation in the large-program life cycle. *Journal of Systems and Software*, 1(3), 213–231. (Cited on page 93.)
- [Leijen & Meijer 1999] Leijen, D. & Meijer, E. (1999). Domain specific embedded compilers. In *2nd USENIX Conference on Domain-Specific Languages (DSL)*. USENIX. (Cited on page 88.)
- [Lengauer et al. 2004] Lengauer, C., Batory, D., Consel, C., & Odersky, M. (Eds.). (2004). *Domain-Specific Program Generation*, volume 3016 of LNCS, Dagstuhl Castle, Germany. Springer. (Cited on pages 217 and 225.)
- [Leslie 1995] Leslie, T. (1995). Efficient approaches to subset construction. Master’s thesis, University of Waterloo, Waterloo, Ontario, Canada. (Cited on page 165.)
- [Livshits & Lam 2005] Livshits, V. B. & Lam, M. S. (2005). Finding security vulnerabilities in Java applications with static analysis. In *14th USENIX Security Symposium*, (pp. 271–286). USENIX. (Cited on pages 66 and 88.)
- [Maor & Shulman 2004] Maor, O. & Shulman, A. (2004). SQL injection signatures evasion. White paper, <http://www.imperva.com/>. (Cited on page 87.)
- [Masuhara & Kawauchi 2003] Masuhara, H. & Kawauchi, K. (2003). Dataflow pointcut in aspect-oriented programming. In *Proceedings of the First Asian Symposium on Programming Languages and Systems (APLAS’03)*, volume 2895 of LNCS, (pp. 105–121). Springer. (Cited on page 94.)
- [McClure & Krüger 2005] McClure, R. A. & Krüger, I. H. (2005). SQL DOM: Compile time checking of dynamic SQL statements. In proceedings ICSE 2005 [Roman et al. 2005], (pp. 88–96). (Cited on pages 66, 69, and 87.)
- [McPeak & Necula 2004] McPeak, S. & Necula, G. C. (2004). Elkhound: A fast, practical GLR parser generator. In Duesterwald, E. (Ed.), *Proceedings of 13th International Conference on Compiler Construction (CC’04)*, volume 2985 of LNCS, (pp. 73–88)., Berlin. Springer. (Cited on pages 141 and 206.)
- [Meijer & Schulte 2003a] Meijer, E. & Schulte, W. (2003a). Programming with rectangles, triangles, and circles. In *Proceedings of XML Conference & Exposition 2003 (XML 2003)*. IDEAlliance. (Cited on pages 4 and 45.)

- [Meijer & Schulte 2003b] Meijer, E. & Schulte, W. (2003b). Unifying tables, objects, and documents. In *Proceedings of the Workshop on Declarative Programming in the Context of Object-Oriented Languages (DP-COOL'03)*. (Cited on pages 4 and 45.)
- [Meijer & van Velzen 2001] Meijer, E. & van Velzen, D. (2001). Haskell Server Pages: Functional programming and the battle for the middle tier. In *2000 ACM SIGPLAN Haskell Workshop*, volume 41/1 of *Electronic Notes in Theoretical Computer Science*. Elsevier Science Publishers. (Cited on page 88.)
- [Mernik et al. 2005] Mernik, M., Heering, J., & Sloane, A. M. (2005). When and how to develop domain-specific languages. *ACM Computing Surveys*, 37(4), 316–344. (Cited on pages 10 and 11.)
- [Møller 2005] Møller, A. (2005). dk.brics.automaton — finite-state automata for Java. <http://www.brics.dk/automaton/>. (Cited on page 75.)
- [Moreau et al. 2003] Moreau, P.-E., Ringeissen, C., & Vittek, M. (2003). A pattern matching compiler for multiple target languages. In proceedings CC 2003 [Hedin 2003], (pp. 61–76). (Cited on page 40.)
- [van Noord 2000] van Noord, G. (2000). Treatment of epsilon moves in subset construction. *Computational Linguistics*, 26(1), 61–76. (Cited on page 165.)
- [Nuutila 1995] Nuutila, E. (1995). *Efficient Transitive Closure Computation in Large Digraphs*. PhD thesis, Helsinki University of Technology. (Cited on page 174.)
- [Nystrom et al. 2003] Nystrom, N., Clarkson, M. R., & Myers, A. C. (2003). Polyglot: An extensible compiler framework for Java. In proceedings CC 2003 [Hedin 2003], (pp. 138–152). (Cited on pages 108, 128, 146, 152, and 178.)
- [Odersky & Zenger 2005] Odersky, M. & Zenger, M. (2005). Independently extensible solutions to the expression problem. In *Proceedings of the Twelfth International Workshop on Foundations of Object-Oriented Languages (FOOL 12)*. (Cited on page 146.)
- [Ongkingco et al. 2006] Ongkingco, N., Avgustinov, P., Tibble, J., Hendren, L., de Moor, O., & Sittampalam, G. (2006). Adding open modules to AspectJ. In proceedings AOSD 2006 [Filman 2006]. (Cited on pages 94 and 130.)
- [Onzon 2007] Onzon, E. (2007). Dypgen: Self-extensible parsers for ocaml. <http://dypgen.free.fr>. (Cited on pages 38 and 180.)
- [Parr] Parr, T. ANTLR Parser Generator. <http://www.antlr.org>. (Cited on page 136.)
- [Rekers 1992] Rekers, J. (1992). *Parser Generation for Interactive Environments*. PhD thesis, University of Amsterdam. (Cited on pages 31, 34, 95, 136, 156, 180, 206, and 207.)

- [Roman et al. 2005] Roman, G.-C., Griswold, W. G., & Nuseibeh, B. (Eds.). (2005). *Proceedings of the 27th International Conference on Software Engineering (ICSE 2005)*, Washington, DC, USA. IEEE Computer Society Press. (Cited on pages 215 and 221.)
- [Sakurai et al. 2004] Sakurai, K., Masuhara, H., Ubayashi, N., Matsuura, S., & Komiya, S. (2004). Association aspects. In Lieberherr, K. (Ed.), *Proceedings of the 3rd International Conference on Aspect-Oriented Software Development (AOSD 2004)*, (pp. 16–25)., Lancaster, UK. ACM Press. (Cited on page 94.)
- [Salomon & Cormack 1989] Salomon, D. J. & Cormack, G. V. (1989). Scannerless NSLR(1) parsing of programming languages. In *Proceedings of the ACM SIGPLAN 1989 Conference on Programming Language Design and Implementation (PLDI 1989)*, (pp. 170–178). ACM Press. (Cited on pages 31, 36, 95, 136, and 175.)
- [Salomon & Cormack 1995] Salomon, D. J. & Cormack, G. V. (1995). The disambiguation and scannerless parsing of complete character-level grammars for programming languages. Technical Report 95/06, Department of Computer Science, University of Manitoba, Winnipeg, Canada. (Cited on pages 95 and 98.)
- [Schatborn 2005] Schatborn, E. P. (2005). GTL, a grammar transformation language for SDF specifications. Master's thesis, Programming Research Group, Faculty of Science, University of Amsterdam, Amsterdam. (Cited on page 200.)
- [Scott & Johnstone 2006] Scott, E. & Johnstone, A. (2006). Right nulled GLR parsers. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, 28(4), 577–618. (Cited on pages 141 and 206.)
- [SDF Website] <http://www.syntax-definition.org>. (Cited on pages 12, 31, and 36.)
- [Sellink & Verhoef 2000] Sellink, M. & Verhoef, C. (2000). Development, assessment, and reengineering of language descriptions. In Ebert, J. & Verhoef, C. (Eds.), *Proceedings of the Fourth European Conference on Software Maintenance and Reengineering*, (pp. 151–160). IEEE Computer Society Press. (Cited on pages 184 and 199.)
- [Shalit 1996] Shalit, A. (1996). *The Dylan reference manual: the definitive guide to the new object-oriented dynamic language*. Addison Wesley Longman Publishing Co., Inc. (Cited on pages 11 and 41.)
- [Smaragdakis & Batory 2000] Smaragdakis, Y. & Batory, D. (2000). Application generators. *Encyclopedia of Electrical and Electronics Engineering*. J.G. Webster (Ed.), John Wiley and Sons. (Cited on page 11.)
- [Stratego Website] <http://www.stratego-language.org>. (Cited on pages 12, 20, 37, and 209.)

- [Su & Wassermann 2006] Su, Z. & Wassermann, G. (2006). The essence of command injection attacks in web applications. In *POPL '06: Conference Record of the 33rd ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, (pp. 372–382)., New York, NY, USA. ACM Press. (Cited on pages 66, 70, and 83.)
- [Szyperski 1996] Szyperski, C. (1996). Independently extensible systems – software engineering potential and challenges. In *Proceedings of the 19th Australian Computer Science Conference*, Melbourne, Australia. (Cited on page 146.)
- [Tanter et al. 2006] Tanter, É., Gybels, K., Denker, M., & Bergel, A. (2006). Context-aware aspects. In *Proceedings of the 5th International Symposium on Software Composition (SC 2006)*, LNCS, (pp. 227–249). Springer. (Cited on pages 94, 131, and 132.)
- [Tanter & Noyé 2005] Tanter, É. & Noyé, J. (2005). A versatile kernel for multi-language AOP. In proceedings GPCE 2005 [Glück & Lowry 2005], (pp. 173–188). (Cited on pages 131 and 142.)
- [Tao et al.] Tao, K., Wang, W., & Palsberg, J. Java Tree Builder (JTB). <http://compilers.cs.ucla.edu/jtb/>. (Cited on pages 13 and 29.)
- [Tarjan 1972] Tarjan, R. E. (1972). Depth-first search and linear graph algorithms. *SIAM Journal on Computing*, 1(2), 146–160. (Cited on page 171.)
- [Tomita 1985] Tomita, M. (1985). *Efficient Parsing for Natural Languages. A Fast Algorithm for Practical Systems*. Kluwer Academic Publishers. (Cited on pages 31, 34, 95, and 156.)
- [Tratt 2005] Tratt, L. (2005). The Converge programming language. Technical Report TR-05-01, Department of Computer Science, King’s College London. (Cited on page 180.)
- [Valkering 2007] Valkering, R. (2007). Syntax error handling in scannerless generalized LR parsers. Master’s thesis, Programming Research Group, University of Amsterdam, Amsterdam, The Netherlands. (Cited on page 206.)
- [Vinju 2005] Vinju, J. (2005). A type driven approach to concrete meta programming. Technical Report SEN-E0507, Centrum voor Wiskunde en Informatica. (Cited on pages 54, 60, and 61.)
- [Visser 1997a] Visser, E. (1997a). Scannerless generalized-LR parsing. Technical Report P9707, Programming Research Group, University of Amsterdam. (Cited on pages 30, 31, 36, 95, 99, 116, 121, and 137.)
- [Visser 1997b] Visser, E. (1997b). *Syntax Definition for Language Prototyping*. PhD thesis, University of Amsterdam. (Cited on pages 12, 30, 31, 37, 48, 60, 69, 71, 95, 99, 137, 148, 176, 178, 184, 185, 189, 190, and 195.)

- [Visser 2002] Visser, E. (2002). Meta-programming with concrete object syntax. In Batory, D., Consel, C., & Taha, W. (Eds.), *Generative Programming and Component Engineering, ACM SIGPLAN/SIGSOFT Conference, GPCE 2002*, volume 2487 of LNCS, (pp. 299–315). Springer. (Cited on pages v, 6, 7, 21, 37, 48, 50, 60, 61, 67, 73, 85, and 147.)
- [Visser 2004] Visser, E. (2004). Program transformation with Stratego/XT: Rules, strategies, tools, and systems in StrategoXT-0.9. In proceedings DSPG 2004 [Lengauer et al. 2004], (pp. 216–238). (Cited on pages 12, 20, 37, 60, 69, 71, and 185.)
- [Visser et al. 1998] Visser, E., Benaissa, Z.-e.-A., & Tolmach, A. (1998). Building program optimizers with rewriting strategies. In *Proceedings of the Third ACM SIGPLAN International Conference on Functional Programming (ICFP'98)*, (pp. 13–26). ACM Press. (Cited on page 37.)
- [Visser 2001] Visser, J. (2001). Visitor combination and traversal control. In *Proceedings of the 16th ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA'01)*, (pp. 270–282). ACM Press. (Cited on page 44.)
- [Weise & Crew 1993] Weise, D. & Crew, R. (1993). Programmable syntax macros. In *Proceedings of the ACM SIGPLAN 1993 Conference on Programming Language Design and Implementation (PLDI'93)*, (pp. 156–165). ACM Press. (Cited on pages 11 and 41.)
- [van Wyk et al. 2007] van Wyk, E., Krishnan, L., Schwerdfeger, A., & Bodin, D. (2007). Attribute grammar-based language extensions for Java. In *European Conference on Object Oriented Programming (ECOOP 2007)*, LNCS. Springer. (Cited on pages 146, 149, and 152.)
- [van Wyk & Schwerdfeger 2007] van Wyk, E. & Schwerdfeger, A. (2007). Context-aware scanning for parsing extensible languages. In proceedings GPCE 2007 [Lawall 2007]. (Cited on page 178.)
- [Xie & Aiken 2006] Xie, Y. & Aiken, A. (2006). Static detection of security vulnerabilities in scripting languages. In *15th USENIX Security Symposium*, (pp. 179–192). USENIX. (Cited on pages 66 and 88.)
- [Xtatic] The Xtatic project: Native XML processing for C#. <http://www.cis.upenn.edu/~bcpierce/xtatic/>. (Cited on pages 4 and 45.)
- [Zook et al. 2004] Zook, D., Huang, S. S., & Smaragdakis, Y. (2004). Generating AspectJ programs with Meta-AspectJ. In Karsai, G. & Visser, E. (Eds.), *Generative Programming and Component Engineering: Third International Conference, GPCE 2004*, volume 3286 of LNCS, (pp. 1–19)., Vancouver, Canada. Springer. (Cited on pages 4, 7, 47, 48, 62, 67, 73, 86, 137, and 147.)