Bibliography

[1] S. Abiteboul and R. Hull. IFO: A formal semantic database model. *ACM Transactions on Database Systems*, 12(4):525–565, December 1987.

[2] A. Adl-Tabatabai, G. Langdale, S. Lucco, and R. Wahbe. Efficient and language-independent mobile programs. In PLDI [206], pages 127–136. Published as *SIGPLAN Notices 31*(5), May 1996.

[3] A.V. Aho, R. Sethi, and J.D. Ullman. *Compilers: Principles, Techniques, and Tools*. Addison-Wesley Publishing Company, Reading, MA, 1986.

[4] A. Albano, L. Cardelli, and R. Orsini. Galileo: A strongly typed, interactive conceptual language. *ACM Transactions on Database Systems*, 10(2):230–260, June 1985.

[5] A. Albano, G. Ghelli, and R. Orsini. Fibonacci reference manual: A preliminary version. Technical Report FIDE/94/102, ESPRIT Basic Research Action, Project Number 6309—$FIDE_2$, 1994.

[6] A. Albano, G. Ghelli, and R. Orsini. An introduction to Fibonacci: A programming language for object databases. Technical Report FIDE/95/120, ESPRIT Basic Research Action, Project Number 6309—$FIDE_2$, 1995.

[7] T. Andrews, C. Harris, K. Sinkel, and J. Duhl. The ONTOS object database. Technical report, Ontologic Inc., Burlinghton, MA, 1989.

[8] A.W. Appel. Garbage collection can be faster than stack allocation. *Information Processing Letters*, 25(4):275–279, June 1987.

[9] A.W. Appel. A runtime system. *Lisp and Symbolic Computation*, 3(4):343–380, November 1990.

[10] A.W. Appel. *Compiling with Continuations*. Cambridge University Press, Cambridge, England, 1992.

[11] A.W. Appel and T. Jim. Continuation-passing, closure-passing style. In POPL [207], pages 293–302.

[12] A.W. Appel and D.B. MacQueen. A Standard ML compiler. In *Functional Programming Languages and Computer Architecture*, number 274 in Lecture Notes in Computer Science, pages 301–324. Springer-Verlag, NY, 1987.

[13] A.W. Appel and D.B. MacQueen. Standard ML of New Jersey. In J. Maluszynski and M. Wirsing, editors, *Third International Symposium on Programming Languages Implementation and Logic Programming (Passau, Germany, 26–28 August 1991)*, pages 1–13, Berlin, Germany, 1991. Springer-Verlag.

[14] A.W. Appel and Z. Shao. An empirical and analytical study of stack vs. heap cost for languages with closures. Technical Report CS-TR-450-94, Princeton University, Department of Computer Science, Princeton, NJ, March 1994.

[15] K. Arnold and J. Gosling. *The Java Programming Language*. The Java Series. Addison-Wesley Publishing Company, Reading, MA, 1996.

[16] M.P. Atkinson. Programming languages and databases. In S.B. Yao, editor, *The Fourth International Conference on Very Large Data Bases (Berlin, West Germany, September 1978)*, pages 408–419, September 1978.

[17] M.P. Atkinson. A vision of persistent systems. In *Proceedings of the International Conference on Deductive and Object-Oriented Databases (Munich, December 1991)*, pages 453–459, 1991. Invited paper.

[18] M.P. Atkinson. Persistent foundations for scalable multi-paradigmal systems. Invited paper. In Özsu et al. [197].

[19] M.P. Atkinson. SPF scalable persistent foundations: Well engineered support for very high performance persistent systems. SERC case for support, January 1992.

[20] M.P. Atkinson, editor. *Fully Integrated Data Environments*. Springer-Verlag, 1997. To be published.

[21] M.P. Atkinson, P.J. Bailey, K.J. Chisholm, W.P. Cockshott, and R. Morrison. An approach to persistent programming. *The Computer Journal*, 26(4):360–365, November 1983.

[22] M.P. Atkinson, F. Bancilhon, D. DeWitt, K. Dittrich, D. Maier, and S. Zdonik. The object-oriented database system manifesto. In W. Kim, J.-M. Nicolas, and S. Nishio, editors, *Deductive and Object-Oriented Databases. Proceedings of the First International Conference on Deductive and Object-Oriented Databases (Kyoto, Japan, 4th–6th December 1989)*. Elsevier Science Publisher B.V., 1989.

[23] M.P. Atkinson, V. Benzaken, and D. Maier, editors. *Persistent Object Systems*. Workshops in Computing. Springer-Verlag in collaboration with the British Computer Society, 1994. Proceedings of the Sixth International Workshop on Persistent Object Systems (Tarascon, Provence, France, 5th–9th September 1994).

[24] M.P. Atkinson and O.P. Buneman. Types and persistence in database programming languages. *ACM Computing Surveys*, 19(2):105–190, June 1987.

[25] M.P. Atkinson, O.P. Buneman, and R. Morrison, editors. *Data Types and Persistence*. Topics in Information Systems, series editors M.L. Brodie, J. Mylopoulos and Schmidt, J.W. Springer-Verlag, 1988. Edited Papers from the Proceedings of the First Workshop on Persistent Object Systems (Appin, Scotland, August 1985).

[26] M.P. Atkinson, K.J. Chisholm, and W.P. Cockshott. PS-algol: An algol with a persistent heap. *ACM SIGPLAN Notices*, 17(7):24–31, July 1982.

[27] M.P. Atkinson, K.J. Chisholm, and W.P. Cockshott. CMS—a chunk management system. *Software Practice and Experience*, 13(3):273–285, March 1983.

[28] M.P. Atkinson, K.J. Chisholm, W.P. Cockshott, and R.M. Marshall. Algorithms for a persistent heap. *Software Practice and Experience*, 13(3):259–272, March 1983.

[29] M.P. Atkinson, L. Daynès, M.J. Jordan, T. Printezis, and S. Spence. An orthogonally persistent Java. *SIGMOD Record*, 25(4):68–75, December 1996.

[30] M.P. Atkinson, J.R. Lucking, R. Morrison, and G.D. Pratten. Persistent Information Space Architecture — PISA club rules. Technical Report PPRR-47-87, Universities of Glasgow and St Andrews, 1987.

[31] M.P. Atkinson and R. Morrison. Procedures as persistent data objects. *ACM Transactions on Programming Languages and Systems*, 4(7):539–559, October 1985.

[32] M.P. Atkinson and R. Morrison. Polymorphic names, types, constancy and magic in a type secure persistent object store. In Carrick and Cooper [70], pages 1–12. Proceedings of the Second International Workshop on Persistent Object Systems (Appin, Scotland, 25th–28th August 1987).

[33] M.P. Atkinson and R. Morrison. Types, bindings and parameters in a persistent environment. In Atkinson et al. [25], chapter 1, pages 3–20. Edited Papers from the Proceedings of the First Workshop on Persistent Object Systems (Appin, Scotland, August 1985).

[34] M.P. Atkinson and R. Morrison. Polymorphic names and iterations. In Bancilhon and Buneman [40], chapter 14, pages 241–256. Edited Proceedings of the Workshop on Database Programming Languages (Roscoff, Brittany, France, September 1987).

[35] M.P. Atkinson and R. Morrison. Orthogonal persistent object systems. *VLDB Journal*, 4(3), 1995.

[36] J. Auslander, M. Philipose, C. Chambers, S. Eggers, and B. Bershad. Fast, effective dynamic compilation. In PLDI [206], pages 149–159. Published as *SIGPLAN Notices 31*(5), May 1996.

[37] D.F. Bacon, S.L. Graham, and O.J. Sharp. Compiler transformations for high-performance computing. *ACM Computing Surveys*, 26(4):345–420, December 1994.

[38] P.J. Bailey, P. Maritz, and R. Morrison. The s-algol abstract machine. Technical Report CS-80-2, Department of Computational Science, University of St Andrews, 1980.

[39] F. Bancilhon, G. Barbedette, V. Benzaken, C. Delobel, S. Gamerman, C. Lécluse, P. Pfeffer, P. Richard, and F. Velez. The design and implementation of o_2, an object-oriented database system. In *Advances in Object-Oriented Database Systems, Proceedings of the Second International Workshop on Object-Oriented Database Systems*, number 334 in Lecture Notes in Computer Science, pages 1–22. Springer-Verlag, 1988.

[40] F. Bancilhon and O.P. Buneman, editors. *Advances in Database Programming Languages.* ACM Press, Frontier Series. Addison-Wesley Publishing Company and ACM Press, 1990. Edited Proceedings of the Workshop on Database Programming Languages (Roscoff, Brittany, France, September 1987).

[41] F. Bancilhon, C. Delobel, and P. Kanellakis, editors. *Building an Object-Oriented Database System: The Story of O_2*. Morgan Kaufmann Publishers, 1992.

[42] Joel F. Bartlett. Compacting garbage collection with ambiguous roots. Technical Report WRL, Research Report 88/2, DEC Western Research Laboratory, Palo Alto, California, February 1988.

[43] Joel F. Bartlett. SCHEME→C: a portable Scheme-to-C compiler. Technical Report WRL, Research Report 89/1, DEC Western Research Laboratory, Palo Alto, California, January 1989.

[44] A. Bawden, R. Greenblatt, J. Holloway, T. Knight, D. Moon, and D. Weinreb. LISP machine progress report. A.I. Lab Memo 444, MIT, August 1977.

[45] C. Beeri, S. Naqvi, R. Ramakrishan, O. Schmueli, and S. Tsur. Sets and negation in a logic and database language (LDL1). In *Proceedings of the Sixth ACM Symposium on Principles of Database Systems.* ACM Press, 1987.

[46] M.E. Benitez and J.W. Davidson. The advantages of machine-dependent global optimization. In J. Gutknecht, editor, *Programming Languages and System Architectures, International Conference (Zurich, Switzerland, March 1994) Proceedings*, volume 782 of *Lecture Notes in Computer Science*, pages 105–124. Springer-Verlag, 1994.

[47] V. Benzaken and C. Delobel. Enhancing performance in a persistent object store: Clustering strategies in O_2. In A. Dearle, G.M. Shaw, and S.B. Zdonik, editors, *Implementing Persistent Object Bases, Principles and Practice*, pages 403–412. San Mateo, CA: Morgan Kaufmann Publishers, 1990. Proceedings of the Fourth International Workshop on Persistent Object Systems, Their Design, Implementation and Use (Martha's Vineyard, USA, September 1990).

[48] T. Berners-Lee, R. Cailliau, and B. Pollermann. World-wide web: The information universe. *Electronic Networking: Research, Applications and Policy*, 1(2):52–58, 1992.

[49] D.G. Bobrow and B. Wegbreit. A model and stack implementation of multiple environments. *Communications of the ACM*, 16(10):591–603, October 1973.

[50] A. Boehm. Simple GC-safe compilation. In *OOPSLA '91, Workshop on Garbage Collection in Object Oriented Systems*, 1991. Position paper.

[51] A. Boehm and M. Wieser. Garbage collection in an uncooperative environment. *Software Practice and Experience*, 18(9):807–820, September 1988.

[52] B. Bolognesi and E. Brinksma. Introduction to the ISO specification language LOTOS. In *The formal description Technique LOTOS*, pages 303–326. North-Holland Publishing Company, Amsterdam, 1989.

[53] Grady Booch. *Object Oriented Design with applications.* Benjamin/Cummings, Redwood City, CA, 1991.

[54] Borland International. *PARADOX Relational Database, User's Guide, version 3.5*, 1990.

[55] M.M. Brandis. *Optimizing Compilers for Structured Programming Languages.* PhD thesis, Swiss Federal Institute of Technology Zurich, ETH Zürich, 1995.

[56] M.M. Brandis and H. Mössenböck. Single-pass generation of static single-assignment form for structured languages. *ACM Transactions on Programming Languages and Systems*, 16(6):1684–1698, November 1994.

[57] M.L Brodie. On the development of data models. In M.L. Brodie, J. Mylopoulos, and Schmidt J.W., editors, *On Conceptual Modelling: Perspectives from Artificial Intelligence, Databases and Programming Languages.* Springer-Verlag, New York, Berlin, Heidelberg, Tokyo, 1984.

[58] A.L. Brown. *Persistent Object Stores.* PhD thesis, University of St Andrews, 1989.

[59] A.L. Brown, R. Carrick, R.C.H. Connor, Q.I. Cutts, A. Dearle, G.N.C. Kirby, R. Morrison, and D.S. Munro. The Persistent Abstract Machine Version 10 / Napier88 (Release 2.0). Universities of St Andrews and Adelaide, 1994.

[60] A.L. Brown, R. Carrick, R.C.H. Connor, A. Dearle, and R. Morrison. The Persistent Abstract Machine. Technical Report PPRR-59-88, Universities of Glasgow and St Andrews, 1988.

[61] A.L. Brown, G. Mainetto, F. Matthes, R. Mueller, and D.J. McNally. An open system architecture for a persistent object store. In Morrison and Atkinson (minitrack coordinators) [178], pages 766–776.

[62] S.J. Bushell, A.L. Brown, A. Dearle, and F.A Vaugham. Native code generation in persistent systems. In Atkinson et al. [23]. Proceedings of the Sixth International Workshop on Persistent Object Systems (Tarascon, Provence, France, 5th–9th September 1994).

[63] L. Cardelli. The functional abstract machine. *Polymorphism*, 1(1), 1983.

[64] L. Cardelli. Amber. In G. Cousineau, P. Curien, and B. Robinet, editors, *Combinators and Functional Programming Languages*, number 242 in Lecture Notes in Computer Science, pages 21–47. Springer-Verlag, 1986.

[65] L. Cardelli. Typeful programming. Digital Systems Research Center Report 45, Digital Equipment Corporation, Systems Research Centre, 130 Lytton Avenue, Palo Alto, Calif., USA, May 1989.

[66] L. Cardelli and P. Wegner. On understanding types, data abstraction and polymorphism. *ACM Computing Surveys*, 17(4):471–523, December 1985.

[67] M. Carey, D. DeWitt, and S. Vandenberg. A data model and query language for EXODUS. In *Proceedings of the ACM SIGMOD Conference on the Management of Data*, Chicago, USA, May 1988.

[68] M.J. Carey and D.J. DeWitt. Of objects and databases: A decade of turmoil. In *Proceedings of the 22th International Conference on Very Large Data Bases (Mumbay (Bombay), India, September 3-6th 1996)*, pages 1–12, 1996.

[69] M.J. Carey, D.J. DeWitt, M.J. Franklin, N.E. Hall, M.L. McAuliffe, J.F. Naughton, D.T. Schuh, M.H. Solomon, C.K. Tan, O.G. Tsatalos, S.J. White, and M.J. Zwilling. Shoring up persistent applications. In *Proceedings of the ACM SIGMOD International Conference on Management of Data (Minneapolis, Minnesota, May 24-27, 1994)*, pages 383–394, 1994.

[70] R. Carrick and R.L. Cooper, editors. *Persistent Object Systems: Their Design, Implementation and Use.* Universities of Glasgow and St Andrews Technical Report PPRR-44-87, 1987. Proceedings of the Second International Workshop on Persistent Object Systems (Appin, Scotland, 25th–28th August 1987).

[71] C. Carson. *CASE*DESIGNER User's Guide and Tutorial.* ORACLE Corporation UK Limited, Chertsey, Surrey, November 1989.

[72] R.G.G. Cattell. Next generation database systems. *Communications of the ACM*, 34(10), October 1991. Editor.

[73] R.G.G. Cattell. *Object Data Management.* Addison-Wesley Publishing Company, Reading, MA, 1991.

[74] R.G.G. Cattell, editor. *The Object Database Standard: ODMG-93.* Morgan Kaufmann Publishers, 1994.

[75] C. Chambers and D. Ungar. Making pure object-oriented languages practical. In Andreas Paepcke, editor, *Proceedings of the Conference on Object-Oriented Programming Systems, Languages and Applications OOPSLA '91 (Phoenix, Arizona, October 1991)*, pages 1–15, 1991. Published as *SIGPLAN Notices 26*(11), November 1993.

[76] David Chase. Private communication (posted to the USENET newsgroup comp.compilers), August 1990.

[77] P.P. Chen. The entity-relationship model — toward a unified view of data. *ACM Transactions on Database Systems*, 1(1):9–36, 1976.

[78] P. Coad and E. Yourdon. *Object-oriented Analysis.* Yourdon Press and Prentice-Hall, Englewood Cliffs, New Jersey, 1990.

[79] W.P. Cockshott, M.P. Atkinson, K.J. Chisholm, P.J. Bailey, and R. Morrison. POMS: a persistent object management system. *Software Practice and Experience*, 14(1):49–71, January 1984.

[80] Codasyl Committee on Data System Languages. Codasyl data base task group report. Technical report, Association for Computing Machinery, 1971.

[81] E.F. Codd. A relational model of data for large shared data banks. *Communications of the ACM*, 13(6):377–387, June 1970.

[82] E.F. Codd. Further normalisation of the data base relational model. In R. Rustin, editor, *Data Base Systems, Courant Computer Science Symposia Series*, volume 6, pages 33–64. Prentice-Hall, Englewood Cliffs, NJ, 1972.

[83] E.F. Codd. Extending the relational model of data to capture more meaning. *ACM Transactions on Database Systems*, 4(4):397–434, December 1979.

[84] R.C.H. Connor. *Types and Polymorphism in Persistent Programming Systems.* PhD thesis, University of St Andrews, 1991.

[85] R.C.H. Connor, A.L. Brown, R. Carrick, A. Dearle, and R. Morrison. The Persistent Abstract Machine. In Rosenberg and Koch [214], pages 353–366. Proceedings of the Third International Workshop on Persistent Object Systems (10th–13th January 1989, Newcastle, New South Wales, Australia).

[86] R.L. Cooper, M.P. Atkinson, A. Dearle, and D. Abderrahmane. Constructing database systems in a persistent environment. In P.M. Stocker and W. Kent, editors, *Proceedings of the Thirteenth International Conference on Very Large Data Bases (Brighton, England, 1987)*, pages 117–125. Los Altos, CA: Morgan Kaufmann Publishers, September 1987.

[87] B.J. Cox. Message/object programming: An evolutionary change in programming technology. *IEEE Software*, 1(1):12–18, August 1984.

[88] Q.I. Cutts. *Delivering the Benefits of Persistence to System Construction and Execution.* PhD thesis, University of St Andrews, 1993.

[89] Q.I. Cutts, R.C.H. Connor, and R. Morrison. The PamCase machine. In Atkinson [20], chapter 2.1.3. To be published.

[90] R. Cytron, J. Ferrante, B.K. Rosen, M.N. Wegman, and F.K Zadeck. An efficient method of computing static single assignment form. In POPL [207], pages 25–35.

[91] S. Davelaar and S.T. van Kooten. *Custom Development Systems Design and Generation using Designer/2000.* Oracle Corporation, 1996.

[92] Antony J.T. Davie. Variable access in languages in which procedures are first class citizens. Technical Report CS/79/2, Department of Computational Science, University of St Andrews, 1979.

[93] Antony J.T. Davie and David J. McNally. CASE — a lazy version of an SECD machine with a flat environment. Technical Report CS/90/19, University of St Andrews, 1990.

[94] Antony J.T. Davie and David J. McNally. The Staple language reference manual. Technical Report CS/90/16, University of St Andrews, 1990.

[95] Antony J.T. Davie and David J. McNally. PCASE — a persistent lazy version of an SECD machine. Technical Report CS/92/7, University of St Andrews, 1992.

[96] Antony J.T. Davie and R. Morrison. *Recursive Descent Compiling.* Ellis Horwood Publishers, Chichester, UK, 1981.

[97] A. Dearle. *On the Construction of Persistent Programming Environments.* PhD thesis, University of St Andrews, 1988.

[98] A. Dearle. Environments: A flexible binding mechanism to support system evolution. In B.H. Shriver, editor, *Proceedings of the Twenty-Second Annual Hawaii International Conference on System Sciences, Volume II Software Track (January 1989)*, pages 46–45, 1989.

[99] A. Dearle, R. di Bona, J. Farrow, F. Henskens, D . Hulse, A. Lindström, S. Norris, J. Rosenberg, and F. Vaughan. Protection in Grasshopper: A Persistent Operating System. In Atkinson et al. [23]. Proceedings of the Sixth International Workshop on Persistent Object Systems (Tarascon, Provence, France, 5th–9th September 1994).

[100] Defence Research Agency. TDF specification part I. Technical report, United Kingdom's Defence Research Agency, RSRE, Malvern, 1991. Contact N.E. Peeling.

[101] Defence Research Agency. TDF facts & figures. Technical report, United Kingdom's Defence Research Agency, RSRE, Malvern, 1992.

[102] Defence Research Agency. A guide to TDF specification. Technical report, United Kingdom's Defence Research Agency, RSRE, Malvern, June 1994. Issue 3.0.

[103] C. Delobel, C. Lécluse, and P. Richard. *Databases: From Relational to Object-Oriented Systems.* International Thomson Publishing, London, UK, 1995.

[104] Tom DeMarco. *Structured Design and System Specification.* Yourdon Press and Prentice-Hall, Englewood Cliffs, NJ, 1978.

[105] P.L. Deutsch and A.M. Schiffman. Efficient implementation of the Smalltalk-80 system. In *Conference Record of the Eleventh ACM Symposium on Principles of Programming Languages—POPL (Salt Lake City, Utah, January 1984)*, pages 297–30. Association for Computing Machinery, 1984.

[106] A. Dilles. *Z: An Introduction to Formal Methods.* Wiley, 1990.

[107] A. Diwan, J.E.B. Moss, and R. Hudson. Compiler support for garbage collection in a statically typed language. In *Proceedings of the SIGPLAN '92 Conference on Programming Language Design and Implementation (San Francisco, CA, June 1992)*, pages 273–282. Association for Computing Machinery, 1992. Published as *SIGPLAN Notices 27*(7), July 1992.

[108] A. Diwan, D. Tarditi, and J.E.B. Moss. Memory subsystem performance of programs with intensive heap allocation. An earlier version is available as a CMU technical report: CMU-CS-93-227, 1995.

[109] M.A. Ellis and B. Stroustrup. *The Annotated C++ Reference Manual.* Addison-Wesley Publishing Company, 1990.

[110] D.R. Engler. VCODE: A retargetable, extensible, very fast dynamic code generation system. In PLDI [206], pages 160–170. Published as *SIGPLAN Notices 31*(5), May 1996.

[111] M. Evered. *Leibniz — a Language to Support Software Engineering.* PhD thesis, Technical University of Darmstad, 1985.

[112] S.I. Feldman, D.M. Gay, M.W. Maimone, and N.L. Schryer. A Fortran-to-C converter. Technical Report Computer Science, No. 149, AT&T Bell Laboratories, Murray Hill, NJ 07974, 1990.

[113] J. Ferrante, K.J. Ottenstein, and J.D. Warren. The program dependence graph and its use in optimization. *ACM Transactions on Programming Languages and Systems*, 9(3):319–349, July 1987.

[114] C.N. Fischer and R.J. Leblanc, Jr. *Crafting a Compiler.* Benjamin/Cummings, Menlo Park, CA, 1988.

[115] D.H. Fishman, D. Beech, H.P. Cate, E.C. Chow, T. Connors, J.W. Davis, N. Derrett, C.G. Hoch, W. Kent, P. Lyngbaek, B. Mahbod, M.A. Neimat, T.A. Ryan, and M.C. Shan. Iris, an object-oriented database management system. *ACM Transactions on Office Information Systems*, 5(1):48–69, January 1987.

[116] C. Flanagan, A. Sabry, B.F. Duba, and M. Felleisen. The essence of compiling with continuations. *ACM SIGPLAN Notices*, 28(6):237–247, June 1993.

[117] M.S.O. Franz. *Code-Generation On-the-Fly: A Key to Portable Software.* PhD thesis, Swiss Federal Institute of Technology Zurich, ETH Zürich, 1995.

[118] C.H. Fraser, D.R. Hanson, and T.A. Proebsting. Engineering a simple, efficient code-generator generator. *ACM Letters on Programming Languages and Systems*, 1(3):213–226, September 1992.

[119] T. Gane and C. Sarson. *Structured Systems Design.* McDonell Douglas, 1982.

[120] A. Gawecki and F. Matthes. The Tycoon Machine Language TML an optimizable persistent program representation. Technical Report FIDE/94/100, ESPRIT Basic Research Action, Project Number 6309—FIDE$_2$, 1994.

[121] L. George, F. Guillame, and J.H. Reppy. A portable and optimizing back end for the SML/NJ compiler. In P.A. Fritzson, editor, *Compiler Construction, 5th International Conference, CC'94, (Edinburgh, U.K., April 1994) Proceedings*, volume 786 of *Lecture Notes in Computer Science*, pages 83–97. Springer-Verlag, 1994.

[122] D. Gillespie. The p2c translator. Available from csvax.cs.caltech.edu by anonymous ftp under the GNUcopyleft, 1989.

[123] A. Goldberg and D. Robson. *Smalltalk-80: The Language and Its Implementation.* Addison-Wesley Publishing Company, Reading, MA, 1983.

[124] G. Goos, W.A. Wulf, A. Evans Jr., and K.J. Butler, editors. *DIANA An Intermediate Language for ADA.* Number 161 in Lecture Notes in Computer Science. Springer-Verlag, 1983.

[125] J. Gosling. Java intermediate bytecodes. In *Proceedings of the ACM SIGPLAN Workshop on Intermediate Representations (San Francisco, CA, January 22, 1995)*, pages 111–118. Association for Computing Machinery, 1995. Published as *SIGPLAN Notices 30*(3), March 1995.

[126] J. Gosling, B. Joy, and G. Steele. *The Java Language Specification.* The Java Series. Addison-Wesley Publishing Company, Reading, MA, 1996.

[127] P.M.D. Gray, D.S. Moffat, and J.B.H. du Boulay. Persistent prolog: A searching storage manager for prolog. In Atkinson et al. [25], pages 353–368. Edited Papers from the Proceedings of the First Workshop on Persistent Object Systems (Appin, Scotland, August 1985).

[128] O. Gruber, L. Amsaleg, L. Daynès, and P. Valduriez. Eos, an environment for object-based systems. In J. Rosenberg (minitrack coordinator), editor, *Proceedings of the Twenty-Fifth Hawaii International Conference on System Sciences, Volume I, Emerging Technologies, Architectural and Operating System Support for Persistent Object Systems*, pages 757–768, 1992.

[129] O. Gruber and P. Valduriez. An object-oriented foundation for desktop computing. Technical Report FIDE/94/80, ESPRIT Basic Research Action, Project Number 6309—FIDE$_2$, 1994.

[130] M. Hammer and D. McLeod. Database description with SDM: A semantic database model. *ACM Transactions on Database Systems*, 6(3):351–386, September 1981.

[131] R. Hieb, R.K. Dybvig, and C. Bruggeman. Representing control in the presence of first-class continuations. In *Proceedings of the SIGPLAN '90 Conference on Programming Language Design and Implementation (White Plains, NY, June 1990)*, pages 66–77. Association for Computing Machinery, 1990. Published as *SIGPLAN Notices 25*(6), June 1990.

[132] W.W. Ho and R.A. Olsson. An approach to genuine dynamic linking. *Software Practice and Experience*, 21(4):375–390, April 1991.

[133] U. Hölzle, C. Chambers, and D. Ungar. Optimizing dynamic-typed object-oriented languages with polymorphic inline caches. In *Proceedings of the European Conference on Object-oriented Programming (1991)*, Lecture Notes in Computer Science 512. Springer-Verlag, 1991.

[134] U Hölzle and D. Ungar. Optimizing dynamically-dispatched call with run-time type feedback. In PLDI [205], pages 326–336. Published as *SIGPLAN Notices 29*(6), June 1994.

[135] Susan Horwitz, Thomas Reps, and David Binkley. Interprocedural slicing using dependence graphs. *ACM Transactions on Programming Languages and Systems*, 12(1):26–60, January 1990.

[136] A. Hosking, J.E.B Moss, and C. Bliss. Design of an object faulting persistent smalltalk. Technical Report 90-45, University of Massachusetts, Amherst, Massachusetts, May 1990.

[137] R. Hull, R. Morrison, and D. Stemple, editors. *Database Programming Languages*. San Mateo, CA: Morgan Kaufmann Publishers, 1989. Proceedings of the Second International Workshop on Database Programming Languages (Salishan Lodge, Gleneden Beach, Oregon, June 1989).

[138] A.J. Hurst and A.S.M. Sajeev. A capability based language for persistent programming: Implementation issues. In Rosenberg and Koch [214], pages 109–125. Proceedings of the Third International Workshop on Persistent Object Systems (10th–13th January 1989, Newcastle, New South Wales, Australia).

[139] IBM. Ibm internal report on the contents of a sample of programs surveyed. Technical report, IBM Research Centre San Jose, California, 1978.

[140] M.A. Jackson. *System Development*. Prentice-Hall, 1983.

[141] K. Jensen and N. Wirth. *PASCAL User Manual and Report*. Springer-Verlag, Berlin, Germany, second edition, 1975.

[142] J.B. Johnston. The contour model of block structure processes. *ACM SIGPLAN Notices*, 6(2):56–82, 1971.

[143] C.B. Jones. *Systematic Software Development Using VDM*. Prentice-Hall, Englewood Cliffs, NJ, second edition, 1990. ISBN 0-13-880733-7.

[144] K. Kato and A. Ohori. An approach to multilanguage persistent type system. In Morrison and Atkinson (minitrack coordinators) [178], pages 810–819.

[145] B.W. Kernighan and D.M. Ritchie. *The C Programming Language*. Prentice-Hall, Englewood Cliffs, NJ, second edition, 1988.

[146] W. Kim, N. Ballou, J. Banerjee, H. Chou, J. Garza, and D. Woelk. Integrating an object-oriented programming system with a database system. In *Proceedings of the Conference on Object-Oriented Programming Systems, Languages and Applications (San Diego, CA, 25th–30th September, 1988)*, 1988.

[147] G.N.C. Kirby. *Reflection and Hyper-Programming in Persistent Programming Systems*. PhD thesis, University of St Andrews, 1993.

[148] G.N.C. Kirby, A.L. Brown, R.C.H. Connor, Q.I. Cutts, A. Dearle, V.S. Moore, R. Morrison, and D.S. Munro. The Napier88 standard library reference manual (version 2.2). Technical Report FIDE/94/105, ESPRIT Basic Research Action, Project Number 6309—$FIDE_2$, 1994.

[149] G.L. Krablin. Using abstract data type techniques in a concurrent persistent programming system. In Carrick and Cooper [70]. Proceedings of the Second International Workshop on Persistent Object Systems (Appin, Scotland, 25th–28th August 1987).

[150] G.L. Krablin. Building flexible multilevel transactions in a distributed persistent environment. In Atkinson et al. [25], chapter 14, pages 213–234. Edited Papers from the Proceedings of the First Workshop on Persistent Object Systems (Appin, Scotland, August 1985).

[151] D. Kranz, R. Kelsey, J. Rees, P. Hudak, J. Philbin, and N. Adams. ORBIT: An optimizing compiler for scheme. In *Proceedings of the ACM SIGPLAN '86 Symposium on Compiler Construction (Palo Alto, CA, June 1986)*, pages 219–233. Association for Computing Machinery, 1987. Published as *SIGPLAN Notices 21*(7), July 1986.

[152] C. Lamb, G. Landis, J. Orenstein, and D. Weinreb. The ObjectStore database system. *Communications of the ACM*, 34(10):50–63, October 1991.

[153] C. Lécluse, P. Richard, and F. Velez. O_2, an object-oriented data model. In Bancilhon and Buneman [40], chapter 15, pages 257–276. Edited Proceedings of the Workshop on Database Programming Languages (Roscoff, Brittany, France, September 1987).

[154] P. Lee and M. Leone. Optimizing ML with run-time code generation. In PLDI [206], pages 137–148. Published as *SIGPLAN Notices 31*(5), May 1996.

[155] X. Leroy. The ZINC experiment: an economical implementation of the ML language. Rapport Techniques 117, INRIA, Rocquencourt, February 1990.

[156] T. Lindholm and F. Yellin. *The Java Virtual Machine Specification*. The Java Series. Addison-Wesley Publishing Company, Reading, MA, 1997.

[157] B. Liskov, D. Curtis, M. Day, S. Ghemawat, P. Gruber, R. Johnson, and A.C. Myers. Theta reference manual. Technical Report Memo 88, Programming Methodology Group, MIT Laboratory for Computer Science, February 1994. Also available at http://www.pmg.lcs.mit.edu/papers/thetaref/.

[158] J.C. Lopes. High performance target language for persistent systems. Department of Computing Science, University of Glasgow, First Year Report, June 1992.

[159] J.C. Lopes. ShTh–Show Thesaurus user interface. Technical Report FIDE/93/76, ESPRIT Basic Research Action, Project Number 6309—FIDE$_2$, 1993.

[160] S. Macrakis. Delivering applications to multiple platforms using ANDF. *AIXpert*, August 1993.

[161] Patricia Maes. *Computational Reflection*. PhD thesis, Universiteit Brussel, 1987.

[162] D. Maier and J. Stein. Development and implementation of an object-oriented DBMS. In B.S. Shriver and P. Wegner, editors, *Research Directions in Object Oriented Programming*, Computer Systems, pages 355–392. MIT Press, Cambridge, MA, 1987.

[163] Marian Makins, editor. *Collins English Dictionary*. HarperCollins Publishers, Glasgow, UK, third edition, 1991.

[164] H.H. Mashburn and Satyanarayanan. RVM recoverable virtual memory. RVM Release 1.3, January 1994.

[165] F. Matthes. P-Quest: Installation and user manual. DBIS Tycoon Report 101-91, Universität Hamburg, Germany, October 1991.

[166] F. Matthes, R. Mueller, and J.W. Schmidt. Object stores as servers in persistent programming environments—the P-Quest experience. Technical Report FIDE/92/48, ESPRIT Basic Research Action, Project Number 3070—FIDE, 1992.

[167] F. Matthes, S. Müßig, and J.W. Schmidt. Persistent polymorphic programming in Tycoon: An introduction. Technical Report FIDE/94/106, ESPRIT Basic Research Action, Project Number 6309—FIDE$_2$, 1994.

[168] F. Matthes and J.W. Schmidt. The type system of DBPL. In Hull et al. [137], pages 219–225. Proceedings of the Second International Workshop on Database Programming Languages (Salishan Lodge, Gleneden Beach, Oregon, June 1989).

[169] D.C.J. Matthews. Poly manual. *ACM SIGPLAN Notices*, 20(9):52–76, September 1985.

[170] J. McCarthy et al. *LISP 1.5 Programmer's Manual*. MIT Press, Cambridge, Massachusetts, 1962.

[171] D.J. McNally. *Models of Persistence in Lazy Functional Programming Systems.* PhD thesis, Department of Computational Science, University of St Andrews, 1993.

[172] Bertrand Meyer. *Object-Oriented Software Construction.* Prentice-Hall International, Hemel Hempstead, UK, 1988.

[173] Microsoft Corporation. *Microsoft Access Language Reference*, 1992.

[174] R. Milner. A proposal for standard ML. *Polymorphism*, 1(3), December 1983.

[175] R. Morrison. *On the development of algol.* PhD thesis, University of St Andrews, 1979.

[176] R. Morrison. Low cost computer graphics for micro computers. *Software Practice and Experience*, 12(8):767–776, 1982.

[177] R. Morrison and M.P. Atkinson. Persistent languages and architectures. In Rosenberg and Keedy [213], pages 9–28. Invited paper.

[178] R. Morrison and M.P. Atkinson (minitrack coordinators), editors. *Proceedings of the Twenty-Fifth Hawaii International Conference on System Sciences, Volume II, Software Technology, Persistent Object Systems*, 1992.

[179] R. Morrison, A.L. Brown, R.C.H. Connor, Q.I. Cutts, A. Dearle, G.N.C. Kirby, and D.S. Munro. The Napier88 reference manual (release 2.0). Technical Report FIDE/94/104, ESPRIT Basic Research Action, Project Number 6309—FIDE$_2$, 1994.

[180] R. Morrison, A.L. Brown, R.C.H. Connor, Q.I. Cutts, A. Dearle, G.N.C. Kirby, J. Rosenberg, and D. Stemple. Protection in persistent object systems. In Rosenberg and Keedy [213], pages 48–66.

[181] R. Morrison, A.L. Brown, R.C.H. Connor, and A. Dearle. The Napier88 reference manual. Technical Report PPRR-77-89, Universities of Glasgow and St Andrews, 1989.

[182] R. Morrison, A. Dearle, A.L. Brown, and M.P. Atkinson. An integrated graphics programming environment. *Computer Graphics Forum*, 5(2):147–157, June 1986. Also available as PPRR-14-86.

[183] R. Morrison, A. Dearle, R.C.H. Connor, and A.L. Brown. An ad hoc approach to the implementation of polymorphism. *ACM Transactions on Programming Languages and Systems*, 13(3):342–371, July 1991.

[184] J.E.B. Moss. Managing stack frames in Smalltalk. In *Proceedings of the ACM SIGPLAN '87 Symposium on Interpreters and Interpretative Techniques (St. Paul, Minnesota, June 1987)*, pages 229–240. Association for Computing Machinery, 1987. Published as *SIGPLAN Notices 22*(7), July 1987.

[185] J.E.B. Moss. Addressing large distributed collections of persistent objects: The Mneme project's approach. In Hull et al. [137], pages 358–374. Proceedings of the Second International Workshop on Database Programming Languages (Salishan Lodge, Gleneden Beach, Oregon, June 1989).

[186] J.E.B. Moss. Design of the Mneme persistent object store. *ACM Transactions on Information Systems*, 8(2):103–139, April 1990.

[187] J.E.B. Moss. Private communication, March 1993.

[188] Atkinson. M.P. and M.J. Jordan, editors. *Proceedings of the First International Workshop on Persistence and Java (September 1996, Drymen, Scotland).* Sunlabs Technical Report, September 1996.

[189] R. Mueller, F. Matthes, and J.W. Schmidt. Towards a unified model of untyped object stores: Experience with the Tycoon store protocol. In Atkinson [20], chapter 2.2.4. To be published.

[190] D.S. Munro. *On the Integration of Concurrency, Distribution and Persistence.* PhD thesis, University of St Andrews, 1993.

[191] J. Mylopoulos, P.A. Bernstein, and H.K.T. Wong. A language facility for designing database intensive applications. *ACM Transactions on Database Systems*, 5(2):185–207, June 1980.

[192] K.V. Nori, U. Ammann, K. Jensen, H.H. Nageli, and C. Jacobi. Pascal-P implementation notes. In D.W. Barron, editor, *Pascal – The Language and its Implementation*, pages 125–170. Wiley, 1981.

[193] C. Norris and L. Pollock. Register allocation over the program dependence graph. In PLDI [205], pages 266–277. Published as *SIGPLAN Notices 29*(6), June 1994.

[194] M. Odersky and P. Wadler. Pizza into Java: Translating theory into practice. In *Proc. 24th ACM Symposium on Principles of Programming Languages*, January 1997.

[195] The common object request broker: Architecture and specification. Published jointly by Object Management Group and X/Open, 1991.

[196] Ontologic Inc. *ONTOS Reference Manual.* Ontologic Inc., Billerica, Massachusetts, USA, 1991.

[197] M.T. Özsu, U. Dayal, and P. Valduriez, editors. *Proceedings of the International Workshop on Distributed Object Management (Edmonton, Canada, 18th–21st August 1992).* Morgan Kaufmann Publishers, 1992.

[198] D.A. Patterson and J. Hennessy. *Computer Architecture, a Quantitative Approach.* Morgan Kaufmann Publishers, San Mateo, CA, 1990.

[199] Persistent Programming Research Group. PS-algol abstract machine manual. Technical Report PPRR-11-85, Universities of Glasgow and St Andrews, 1985.

[200] Persistent Programming Research Group. PS-algol reference manual — fourth edition. Technical Report PPRR-12-87, Universities of Glasgow and St Andrews, 1987.

[201] S.L. Peyton-Jones. *The Implementation of Functional Programming Languages.* Series in Computer Science. Prentice-Hall International, Hemel Hempstead, UK, 1987.

[202] S.L. Peyton-Jones. Implementing lazy functional languages on stock hardware: the spineless tagless G-machine. *Journal of Functional Programming*, 2(2):127–202, April 1992.

[203] S.L. Peyton-Jones. Private communication, February 1994.

[204] S.L. Peyton-Jones, C. Hall, K. Hammond, W. Partain, and P. Wadler. The Glasgow Haskell compiler: a technical overview. In *Proceedings of the UK Joint Framework for Information Technology (JFIT) Technical Conference (Keele, 1993)*, 1993.

[205] *Proceedings of the SIGPLAN '94 Conference on Programming Language Design and Implementation (Orlando, Florida, June 20-24, 1994)*. Association for Computing Machinery, 1994. Published as *SIGPLAN Notices 29*(6), June 1994.

[206] *Proceedings of the SIGPLAN '96 Conference on Programming Language Design and Implementation (Philadelphia, PA, May 21-24, 1996)*. Association for Computing Machinery, 1996. Published as *SIGPLAN Notices 31*(5), May 1996.

[207] *Conference Record of the Sixteenth Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages—POPL (Austin, Texas, January 1989)*. ACM Press, 1989.

[208] C.V. Ramamoorthy, A. Prakash, W. Tsai, and Y. Usuda. Software engineering: Problems and perspectives. *IEEE Computer*, 17(10), October 1984.

[209] Jonathan Rees and W. Clinger. The revised^3 report on the algorithmic language Scheme. AI Memo 848a, Massachusetts Institute of Technology, Cambridge, Massachusetts, September 1986.

[210] J.E. Richardson. *E: A Persistent Systems Implementation Language*. PhD thesis, Computer Sciences Department, University of Wisconsin, Madison, WI, 1989.

[211] J.E. Richardson, M.J. Carey, and D.T. Schuh. The design of the E programming language. *ACM Transactions on Programming Languages and Systems*, 15(3):494–534, July 1993.

[212] J. Rosenberg and J.L. Keedy. Object management and addressing in the MONADS architecture. In Carrick and Cooper [70], pages 114–133. Proceedings of the Second International Workshop on Persistent Object Systems (Appin, Scotland, 25th–28th August 1987).

[213] J. Rosenberg and J.L. Keedy, editors. *Security and Persistence. Proceedings of the International Workshop on Computer Architectures to Support Security and Persistence of Information (Bremen, West Germany, 8–11 May 1990)*, Workshops in Computing. Springer-Verlag in collaboration with the British Computer Society, 1990.

[214] J. Rosenberg and D. Koch, editors. *Persistent Object Stores*. Workshops in Computing. Springer-Verlag in collaboration with the British Computer Society, 1989. Proceedings of the Third International Workshop on Persistent Object Systems (10th–13th January 1989, Newcastle, New South Wales, Australia).

[215] J. Rumbaugh. *Object-oriented Modeling and Design*. Prentice-Hall, Englewood Cliffs, New Jersey, 1991.

[216] G. Russell, P. Shaw, and W.P. Cockshott. DAIS: An Object-Addressed Processor Cache. In Atkinson et al. [23]. Proceedings of the Sixth International Workshop on Persistent Object Systems (Tarascon, Provence, France, 5th–9th September 1994).

[217] Gordon Russell. *DOLPHIN: Persistent, Object-oriented and Networked*. PhD thesis, Department of Computer Science, University of Strathclyde, Glasgow, Scotland, 1995.

[218] A. Sabry and M. Felleisen. Reasoning about programs in continuation-passing style. In *Proceedings of the 1992 ACM Conference on Lisp and Functional Programming (San Francisco, CA, June 1992)*, pages 288–298. Association for Computing Machinery, 1992.

[219] A.S.M. Sajeev and A.J. Hurst. Programming persistence in χ. *IEEE Computer*, pages 57–66, September 1992.

[220] C. Schaffert. CORBA: OMG's object request broker. In Özsu et al. [197].

[221] J.W. Schmidt. Some high level language constructs for data of type relation. *ACM Transactions on Database Systems*, 2(3):247–261, September 1977.

[222] P. Schwartz, W. Chang, J.C. Freytag, G. Lohman, J. McPherson, C. Mohan, and H. Pirahesh. Extensibility in the STARBURST database system. In K.R. Dittrich and U. Dayal, editors, *Proceedings of the ACM/IEEE International Workshop on Object-Oriented Database Systems (23rd–26th September 1986, Pacific Grove, CA)*, pages 85–92. IEEE Computer Society Press, 1986.

[223] M. Serrano. *Vers une Compilation Portable et Performant des Langages Fonctionnels*. PhD thesis, Université Pierre et Marie Curie (Paris VI), France, December 1994.

[224] Z. Shao and A.W. Appel. Space-efficient closure representations. In *Proceedings of the 1994 ACM Conference on Lisp and Functional Programming (New York, June 1994)*, pages 150–161. Association for Computing Machinery, 1994.

[225] D.W. Shipman. The functional data model and the data language DAPLEX. *ACM Transactions on Database Systems*, 6(1):140–173, March 1981.

[226] O. Shivers. Control-flow analysis in Scheme. In *Proceedings of the SIGPLAN '88 Conference on Programming Language Design and Implementation (Atlanta, Georgia, June 22-24, 1988)*, pages 164–174. Association for Computing Machinery, 1988. Published as *SIGPLAN Notices 23*(7), July 1988.

[227] N.C. Shu, B.C. Housel, R.W. Taylor, S.P. Ghosh, and V.Y. Lum. EXPRESS: A data EXtraction, Processing, and REStructuring System. *ACM Transactions on Database Systems*, 2(2):134–174, June 1977.

[228] V. Singhal, S.V. Kakkad, and P.R. Wilson. Texas: An efficient, portable persistent store. In A. Albano and R. Morrison, editors, *Persistent Object Systems: Implementation and Use*, Workshops in Computing, pages 11–33. Springer-Verlag in collaboration with the British Computer Society, 1992. Proceedings of the Fifth International Workshop on Persistent Object Systems (San Miniato, Italy, 1st–4th September 1992).

[229] R. Sites, A. Chernoff, M.B. Kirk, M.P. Marks, and S.G. Robinson. Binary translation. *Communications of the ACM*, 36(2):69–81, February 1993.

[230] D.I.K. Sjøberg. *Thesaurus-Based Methodologies and Tools for Maintaining Persistent Application Systems*. PhD thesis, University of Glasgow, July 1993.

[231] D.I.K. Sjøberg, M.P. Atkinson, J.C. Lopes, and P.W. Trinder. Building an integrated persistent application. In C. Beeri, A. Ohori, and D.E. Shasha, editors, *Database Programming Languages (DBPL-4)*, Workshops in Computing, pages 359–375. Springer-Verlag in collaboration with the British Computer Society, 1993. Proceedings of the Fourth International Workshop on Database Programming Languages—Object Models and Languages (Manhattan, New York City, USA, 30th August–1st September 1993).

[232] D.I.K. Sjøberg, Q.I. Cutts, R.C. Welland, and M.P. Atkinson. Analysing persistent language applications. Technical Report FIDE/94/109, ESPRIT Basic Research Action, Project Number 6309—FIDE$_2$, 1994.

[233] J.M. Spivey. *The Z Notation*. Prentice-Hall, 1989.

[234] R. Stallman. Using and porting Gnu CC version 2.0. Documentation for the Gnu C compiler, Free Software Foundation, Cambridge, Mass, February 1992.

[235] R. Stallman. Private communication (posted to the USENET newsgroup comp.compilers), January 1993.

[236] G.L. Steele Jr. RABBIT: A compiler for scheme (a study in compiler optimization). Master's thesis, AI Laboratory, MIT, Cambridge, Mass., May 1978. Also as a Technical Report, AI-TR-474.

[237] D. Stemple, R.B. Stanton, T. Sheard, P.C. Philbrow, R. Morrison, G.N.C. Kirby, L. Fegaras, R.L. Cooper, R.C.H. Connor, M.P. Atkinson, and S. Alagic. Type-safe linguistic reflection: A generator technology. Technical Report FIDE/92/49, ESPRIT Basic Research Action, Project Number 3070—FIDE, 1992. 29pp.

[238] M. Stonebraker and L. Rowe. The design of Postgres. In *Proceedings of the ACM SIGMOD Conference on the Management of Data*, Washington D. C., May 1986.

[239] J. Strong, A. Wegstein, J. Tritter, O. Olsztyn, O. Mock, and T. Steel. The problem of programming communication with changing machines: a proposed solution. *Communications of the ACM*, 1(8):12–18, August 1958. Part 2: 1(9):9–15, September 1958. Report of the Share Ad-Hoc Committee on Universal Languages.

[240] B. Stroustrup. *The C++ Programming Language*. Addison-Wesley Publishing Company, Reading, MA, 1986.

[241] D. Tarditi, G. Morrisett, P. Cheng, C. Stone, R. Harper, and P. Lee. TIL: A type-directed optimizing compiler for ML. In PLDI [206], pages 181–192. Published as *SIGPLAN Notices 31*(5), May 1996.

[242] David Tarditi, Anurag Acharya, and Peter Lee. No assembly required: Compiling Standard ML to C. Technical Report CMU-CS-90-187, School of Computer Science, Carnegie Mellon University, November 1990.

[243] David Tarditi, Peter Lee, and Anurag Acharya. No assembly required: Compiling Standard ML to C. *ACM Letters on Programming Languages and Systems*, 1(2):161–177, June 1992.

[244] R.W. Taylor and R.L. Frank. CODASYL data-base management systems. *ACM Computing Surveys*, 8(1):67–103, March 1976.

[245] R.D. Tennent. Language design methods based on semantic principles. *Acta Informatica*, 8:97–112, 1977.

[246] D. Teodosiu. HARE: An optimizing portable compiler for scheme. *ACM SIGPLAN Notices*, 26(1):109–120, January 1991.

[247] D.C. Tsichritzis and F.H. Lochovsky. *Data Models*. Prentice-Hall, 1982.

[248] S. Tsur and C. Zaniolo. LDL: a logic-based data language. In *Proceedings of the Twelfth International Conference on Very Large Data Bases (Kyoto, Japan, 24th–28th September 1986)*, 1986.

[249] J.D. Ullman. *Principles of Data and Knowledge Bases*, volume 1. Computer Science Press, 1988.

[250] D. Ungar and R. Smith. SELF: The power of simplicity. *Lisp and Symbolic Computation*, 4(3):1–20, 1991. Preliminary version appeared in *Proc. ACM Symp. on Object-Oriented Programming: Systems, Languages, and Applications,* 1987, 227-241.

[251] F. Vaughan. *Implementation of Distributed Orthogonal Persistence Using Virtual Memory*. PhD thesis, University of Adelaide, Faculty of Mathematical and Computer Sciences, December 1994.

[252] F. Velez, G. Bernard, and V. Darnis. The O_2 Object Manager, an Overview. In P.M.G. Apers and G. Wiederhold, editors, *Proceedings of the Fifteenth International Conference on Very Large Data Bases (Amsterdam, Netherlands, 22nd–25th August 1989)*, 1989.

[253] F. Wai. Distributed PS-algol. In Rosenberg and Koch [214], pages 126–140. Proceedings of the Third International Workshop on Persistent Object Systems (10th–13th January 1989, Newcastle, New South Wales, Australia).

[254] P.R. Wilson. Uniprocessor garbage collection techniques. In Y. Bekkers and J. Cohen, editors, *International Workshop on Memory Management (St Malo, France, September 1992)*, number 637 in Lecture Notes in Computer Science. Springer-Verlag, 1992.

[255] N. Wirth. *Programming in Modula-2.* Springer-Verlag, Berlin, Germany, second edition, 1983.

[256] E. Yourdon and L. Constantine. *Structured Design.* Yourdon Press, Englewood Cliffs, New Jersey, 1978.