

BIBLIOGRAPHY

- [1] Robert C. Bogdan and Sari Knopp Biklen. *Qualitative Research For Education*. Allyn and Bacon, Boston, MA, second edition, 1992.
- [2] Paolo Bucci, Joseph E. Hollingsworth, Joan Krone, and Bruce W. Weide. Implementing components in RESOLVE. *ACM SIGSOFT Software Engineering Notes*, 19(4):40–52, October 1994.
- [3] R. M. Burstall and J. A. Goguen. The semantics of Clear, a specification language. In *Proceedings of 1979 Copenhagen Winter School on Abstract Software Specification*, volume 86 of *Lecture Notes in Computer Science*, pages 292–332, New York, NY, 1980. Springer-Verlag.
- [4] R. A. DeMillo, R. J. Lipton, and A. J. Perlis. Social processes and proofs of theorems and programs. *Communications of the ACM*, 22(5):271–280, 1979.
- [5] Stephen H. Edwards. An approach for constructing reusable software components in Ada. IDA Paper P-2378, Institute for Defense Analyses, Alexandria, VA, September 1990.
- [6] Stephen H. Edwards. Common interface models are necessary to support composability. In Larry Latour, Steve Philbrick, and Chandu Bhavsar, editors, *Proceedings of the Fourth Annual Workshop on Software Reuse*, November 1991.
- [7] Stephen H. Edwards. Common interface models for reusable software. *International Journal of Software Engineering and Knowledge Engineering*, 3(2):193–206, 1993.
- [8] Stephen H. Edwards. Inheritance: One mechanism, many conflicting uses. In Larry Latour, editor, *Proceedings of the Sixth Annual Workshop on Software Reuse*, November 1993.
- [9] Stephen H. Edwards, Wayne D. Heym, Timothy J. Long, Murali Sitaraman, and Bruce W. Weide. Specifying components in RESOLVE. *ACM SIGSOFT Software Engineering Notes*, 19(4):29–39, October 1994.

- [10] George W. Ernst, Raymond J. Hookway, James A. Menegay, and William F. Ogden. Modular verification of Ada generics. *Computer Languages*, 16(3/4):259–280, 1991.
- [11] George W. Ernst, Raymond J. Hookway, and William F. Ogden. Modular verification of data abstractions with shared realizations. *IEEE Transactions on Software Engineering*, 20(4):288–307, April 1994.
- [12] George W. Ernst, James A. Menegay, Raymond J. Hookway, and William F. Ogden. Semantics of programming languages for modular verification. Technical Report CES-85-4, Computer Engineering and Science Department, Case Western Reserve University, October 1985.
- [13] George W. Ernst, Jainendra K. Navlakha, and William F. Ogden. Verification of programs with procedure-type parameters. *Acta Informatica*, 18:149–169, 1982.
- [14] Robert L. Glass. The cognitive view: A different look at software design. In *Software Conflict: Essays on the Art and Science of Software Engineering*, pages 25–31. Yourdon Press, Englewood Cliffs, NJ, 1991.
- [15] J. A. Goguen. LIL: A library interconnect language. In *Report On Program Libraries Workshop*, pages 12–51, Menlo Park, CA, October 1983. SRI International.
- [16] J. A. Goguen and J. Meseguer. Rapid prototyping in the OBJ executable specification language. *Software Engineering Notes*, 7(5):75–84, December 1982.
- [17] J. A. Goguen and J. Tardo. An introduction to OBJ: A language for writing and testing software specifications. *Specification of Reliable Software*, pages 170–189, 1979.
- [18] Joseph Goguen. Abstract errors for abstract data types. In Peter Neuhold, editor, *Proceedings of First IFIP Working Conference on Formal Description of Programming Concepts*, pages 21.1–21.32, Cambridge, MA, 1977. MIT Press.
- [19] Joseph A. Goguen. Parameterized programming. *IEEE Transactions on Software Engineering*, SE-10(5):528–543, September 1984.
- [20] Joseph A. Goguen. Reusing and interconnecting software components. *IEEE Computer*, 19(2):16–28, February 1986.

- [21] Joseph A. Goguen. Principles of parameterized programming. In Ted J. Biggerstaff and Alan J. Perlis, editors, *Software Reusability, Volume I: Concepts and Models*, pages 159–225. ACM Press, New York, NY, 1989.
- [22] D. Gries and F. B. Schneider. *A Logical Approach to Discrete Math*. Springer-Verlag, New York, NY, 1993.
- [23] Raymonde Guindon, Herb Krasner, and Bill Curtis. Breakdowns and processes during the early activities of software design by professionals. In E. Soloway and S. Iyengar, editors, *Empirical Studies of Programmers*, pages 65–82. Ablex, 1985.
- [24] Douglas E. Harms. *The Influence of Software Reuse on Programming Language Design*. PhD thesis, Dept. of Computer and Information Science, The Ohio State University, Columbus, OH, 1990.
- [25] Douglas E. Harms and Bruce W. Weide. Copying and swapping: Influences on the design of reusable software components. *IEEE Transactions on Software Engineering*, 17(5):424–435, May 1991.
- [26] R. Harper, J. Mitchell, and E. Moggi. Higher-order modules and the phase distinction. In *Proceedings of the 17th Annual ACM Symposium on Principles of Programming Languages*, New York, NY, 1990. ACM Press.
- [27] Wael A. Hegazy. *The Requirements of Testing a Class of Reusable Software Components*. PhD thesis, Dept. of Computer and Information Science, The Ohio State University, Columbus, OH, 1989.
- [28] Wayne D. Heym, Timothy J. Long, William F. Ogden, and Bruce W. Weide. Mathematical foundations and notation of RESOLVE. Technical Report OSU-CISRC-8/94-TR45, The Ohio State University, Columbus, OH, August 1994.
- [29] Wayne David Heym. *Computer Program Verification: Improvements for Human-Directed Proof Construction*. PhD thesis, Dept. of Computer and Information Science, The Ohio State University, Columbus, OH, 1995 (to appear).
- [30] Paul N. Hilfinger. *Abstraction Mechanisms and Language Design*. The MIT Press, Cambridge, MA, 1983.
- [31] Joseph Hollingsworth. *Software Component Design-for-Reuse: A Language Independent Discipline Applied to Ada*. PhD thesis, Dept. of Computer and Information Science, The Ohio State University, Columbus, OH, 1992.

- [32] Robert W. Holt, Deborah A. Boehm-Davis, and Alan C. Schultz. Mental representations of programs for student and professional programmers. In E. Soloway and S. Iyengar, editors, *Empirical Studies of Programmers*, pages 33–46. Ablex, 1985.
- [33] Joan Krone. *The Role of Verification in Software Reusability*. PhD thesis, Dept. of Computer and Information Science, The Ohio State University, Columbus, OH, 1988.
- [34] Joan Krone and Murali Sitaraman. A modular system for verification of functionality and performance correctness of software components. Technical Report TR 93-5, Dept. of Statistics and Computer Science, West Virginia University, Morgantown, WV, May 1993.
- [35] Joan Krone and Murali Sitaraman. On modularity and tightness of real-time verification. *Real-Time Newsletter*, 9(1/2):109–115, Spring/Summer 1993.
- [36] Wilf R. LaLonde. Designing families of data types using exemplars. *ACM Transactions on Programming Languages and Systems*, 11(2):212–248, April 1989.
- [37] Larry Latour, Don Batory, Stephen Edwards, Haim Kilov, Trudy Levine, Haikuan Li, Luqi, Ben Sijsma, Will Tracz, and Martin Wirsing. Theoretical foundations working group report. Available by Anonymous FTP from the WISR Software Repository at gandalf.umcs.maine.edu, 1992.
- [38] Larry Latour and Curtis Meadow. Scaling up the 3Cs model—a schema for extensible generic architectures. In Larry Latour, Steve Philbrick, and Chandu Bhavsar, editors, *Proceedings of the Fourth Annual Workshop on Software Reuse*, November 1991.
- [39] Harry R. Lewis and Christos H. Papadimitriou. *Elements of the Theory of Computation*. Prentice-Hall, Inc., Englewood Cliffs, NJ, 1981.
- [40] Barbara Liskov and John Guttag. *Abstraction and Specification in Program Development*. The MIT Electrical Engineering and Computer Science Series. MIT Press, Cambridge, MA, 1986.
- [41] D. C. Littman, J. Pinto, S. Letovsky, and E. Soloway. Mental models and software maintenance. In E. Soloway and S. Iyengar, editors, *Empirical Studies of Programmers*, pages 80–98. Ablex, 1986.

- [42] Jacques Loeckx and Kurt Sieber. *The Foundations of Program Verification, Second Edition*. John Wiley and Sons, New York, NY, 1987.
- [43] D. Luckham, F. W. von Henke, B. Krieg-Brückner, and O. Owe. *ANNA: A Language for Annotating Ada Programs*. Springer-Verlag, New York, NY, 1987.
- [44] David B. MacQueen and Mads Tofte. A semantics for higher-order functors. In *5th European Symposium on Programming*, volume 788 of *Lecture Notes in Computer Science*, pages 409–423. Springer-Verlag, New York, NY, April 1994.
- [45] Bertrand Meyer. Genericity versus inheritance. In *OOPSLA'86 Proceedings*, pages 391–405, New York, NY, 1986. ACM Press.
- [46] Bertrand Meyer. *Object-Oriented Software Construction*. Prentice Hall, New York, NY, 1988.
- [47] Robin Milner and Mads Tofte. *Commentary of Standard ML*. MIT Press, Cambridge, MA, 1991.
- [48] Robin Milner, Mads Tofte, and Robert Harper. *The Definition of Standard ML*. MIT Press, Cambridge, MA, 1990.
- [49] Donald A. Norman. Some observations on mental models. In Ronald M. Baecker and William A. S. Buxton, editors, *Readings In Human-Computer Interaction: A Multidisciplinary Approach*, pages 241–244. Morgan Kaufmann Publishers, Inc., San Mateo, CA, 1987.
- [50] Donald A. Norman. *The Design of Everyday Things*. Doubleday/Currency, New York, NY, 1990.
- [51] William F. Ogden, Murali Sitaraman, Bruce W. Weide, and Stuart H. Zweben. The RESOLVE framework and discipline—a research synopsis. *ACM SIGSOFT Software Engineering Notes*, 19(4):23–28, October 1994.
- [52] Frank G. Pagan. *Formal Specification of Programming Languages: A Panoramic Primer*. Prentice-Hall, Inc., Englewood Cliffs, NJ, 1981.
- [53] D. L. Parnas. On the criteria to be used in decomposing systems into modules. *Communications of the ACM*, 15(12):1053–1058, December 1972.
- [54] Laurence C. Paulson. *ML for the Working Programmer*. Cambridge University Press, Cambridge, 1991.

- [55] E. Robinson. Logical aspects of denotational semantics. In D. H. Pitt, A. Poigné, and D. E. Rydeheard, editors, *Category Theory and Computer Science*, volume 283 of *Lecture Notes in Computer Science*, pages 238–253. Springer-Verlag, New York, NY, 1987.
- [56] John G. Sanderson. *A Relational Theory of Computing*, volume 82 of *Lecture Notes in Computer Science*. Springer-Verlag, New York, NY, 1980.
- [57] Murali Sitaraman. *Mechanisms and Methods for Performance Tuning of Reusable Software Components*. PhD thesis, Dept. of Computer and Information Science, The Ohio State University, Columbus, OH, 1990.
- [58] Murali Sitaraman. On modular and tight performance specification and verification of object-oriented software components. In *Proceedings of the 1994 International Conference on Software Reuse*, Washington, D.C., to appear, November 1994. IEEE Computer Society Press.
- [59] Ian Sommerville. *Software Engineering*. Addison-Wesley Publishing Company, Reading, MA, third edition, 1989.
- [60] Allen Stoughton. *Fully Abstract Models of Programming Languages*. Research Notes in Theoretical Computer Science. Pitman, London, 1988.
- [61] Mary E. Swigonski. Feminist standpoint theory and the questions of social work research. *Affilia*, 8(2):171–183, Summer 1993.
- [62] Will Tracz. Implementation working group summary. In James Baldo, Jr., editor, *Reuse in Practice Workshop Summary*, pages 10–19, Alexandria, VA, April 1990. IDA Document D-754, Institute for Defense Analyses.
- [63] William Tracz. *Formal Specification of Parameterized Programs in LILEANNA*. PhD thesis, Dept. of Electrical Engineering, Stanford University, Stanford, CA, to appear 1993.
- [64] Diane B. Walz, Joyce J. Elam, Herb Krasner, and Bill Curtis. A methodology for studying software design teams: An investigation of conflict behaviors in the requirements definition phase. In E. Soloway and S. Iyengar, editors, *Empirical Studies of Programmers*, pages 83–99. Ablex, 1985.
- [65] Bruce Weide. The Ohio State University, personal communication, 1993.

- [66] Bruce W. Weide et al. Special feature: Component-based software using RESOLVE. *ACM SIGSOFT Software Engineering Notes*, 19(4):21–67, October 1994.
- [67] Bruce W. Weide, William F. Ogden, and Stuart H. Zweben. Reusable software components. In M. C. Yovits, editor, *Advances in Computers*, volume 33, pages 1–65. Academic Press, 1991.