

Thomas J. Ostrand

PUBLICATIONS

73. Automated test mapping and coverage for network topologies. *Proc. ACM SIGSOFT Int. Sym. Software Testing and Analysis (ISSTA 2018)*, Amsterdam, June 2018. (with P. Strandberg, D. Sundmark, W. Afzal and E. Weyuker)
72. Automated System-Level Regression Test Prioritization in a Nutshell. *IEEE Software*, Vol. 34, No. 4, pp. 30-37, 2017. (with P. Strandberg, D. Sundmark, W. Afzal and E. Weyuker)
71. Identifying Fault-prone Files in Large Industrial Software Systems. in *Perspectives on Data Science for Software Engineering*, ed. T. Menzies, L. Williams and T. Zimmerman, published by Morgan Kaufmann, Cambridge, MA, 2016. (with E. Weyuker)
70. Experience Report: Automated system level regression test prioritization using multiple factors. Best paper award, *Proc. 27th Int. Symp. on Software Reliability Engineering (ISSRE 2016)*, Ottawa, Canada, October 2016. (with P. Strandberg, D. Sundmark, W. Afzal and E. Weyuker)
69. Transitioning Fault Prediction Models to a New Environment. *Proc. 12th European Dependable Computing Conference (EDCC 2016)*, Gothenburg, Sweden, September 2016. (with J. Derhag, E. Weyuker, and D. Sundmark)
68. Automated test generation using model checking: an industrial evaluation. *Int. Journal on Software Tools for Technology Transfer*, Springer, Nov. 2014. (with E. Enoiu, A. Čaušević, E. Weyuker, D. Sundmark, P. Pettersson)
67. The Limited Impact of Individual Developer Data on Software Defect Prediction. *Empirical Software Eng. Journal*, Vol. 18, No. 3, June 2013. (with E. Weyuker and R. Bell)
66. On the use of calling structure information to help predict software fault proneness. *Empirical Software Eng. Journal*, Vol. 17, No. 4-5, Aug 2012. (with Y. Shin, E. Weyuker and R. Bell)
65. Can File Level Characteristics Help Identify System Level Fault-Proneness? *Proc. Haifa Verification Conference 2011 (HVC11)*, Haifa, Israel, December 2011. (with E. Weyuker)
64. Predicting Bugs in Large Industrial Software Systems, *Lec. Notes in Comp. Sci.*, Vol. 7171, pp 71-93. Springer, 2011 (with E. Weyuker)
63. Software Fault Prediction. *AT&T Software Symposium*, Middletown, NJ, October 2011. (with E. Weyuker, R. Bell, A. Nocera, A. Gauld)
62. Does Measuring Code Change Improve Fault Prediction? *Proc. 7th Int. Conference on Predictive Models in Software Engineering (Promise2011)*, Banff, Canada, September 2011. (with E. Weyuker and R. Bell)
61. Replicate, Replicate, Replicate. *Proc. Workshop on Replication in Empirical Software Engineering Research*, Banff, Canada, September 2011. (with E. Weyuker and R. Bell)
60. Assessing the Impact of Using Fault-Prediction in Industry. *Proc Testing: Academic & Industrial Conference (TAIC 2011)*, Berlin, Germany, March 2011. (with E. Weyuker and R. Bell)
59. Software Testing Research and Software Engineering Education. *Workshop on the Future of Software Engineering and Research*, Santa Fe, NM, November 2010. (with E. Weyuker)
58. Finding Fault: Developing an Automated System for Predicting Which Files Will Contain Defects, book chapter in *Making Software: What Really Works, and Why We Believe It*, O'Reilly Media, Inc., October 2010. (with E. Weyuker)
57. Programmer-based Fault Prediction. *Proc. Int. Conference on Predictive Models in Software Engineering (PROMISE10)*, Timisoara, Romania, September, 2010. (with E. Weyuker and R. Bell)

56. Software Fault Prediction Tool. *Proc. Int. Symposium on Software Testing and Analysis (ISSTA2010)*, Trento, Italy, July 2010. (with Elaine Weyuker)
55. Comparing the Effectiveness of Several Modeling Methods for Fault Prediction *Empirical Software Eng. Journal*, June 2010. (with E. Weyuker and R. Bell)
54. We're Finding Most of the Bugs, but What Are We Missing? *Proc. Third Int. Conf on Software Testing, Verification and Validation (ICST2010)*, Paris, France, April 2010. (with E. Weyuker and R. Bell)
53. Does Calling Structure Information Improve the Accuracy of Fault Prediction? *Proc. Sixth Int. Working Conference on Mining Software Repositories(MSR 2009)*. Vancouver, BC, Canada, May 2009. (with Y. Shin, E. Weyuker and R. Bell)
52. Progress in Automated Software Defect Prediction. *Proc. Haifa Verification Conference*, Haifa, Israel, October 2008. (with E. Weyuker)
51. Do Too Many Cooks Spoil the Broth? Using the Number of Developers to Enhance Defect Prediction Models. *Empirical Software Eng. Journal*, October 2008. (with E. Weyuker and R. Bell)
50. Comparing Methods to Identify Defect Reports in a Change Management Database. *Proc. Defects in Large Software Systems Workshop (DEFECTS08)*, Seattle, Washington, July 2008. (with E. Weyuker)
49. Comparing Negative Binomial and Recursive Partitioning Models for Fault Prediction. *Proc. Predictive Models in Software Engineering (PROMISE'08)*, Leipzig, Germany, May 2008. (with E. Weyuker)
48. What Can Fault Prediction Do For YOU? *Proc. Second Int. Conference on Tests and Proofs (TAP08)*, Prato, Italy, April 2008. Published as Lecture Notes in Computer Science 4966, Springer. (with E. Weyuker)
47. How to Measure Success of Software Prediction Models. *Proc. Fourth Int. Workshop on Software Quality Assurance*, Dubrovnik, Croatia, Sept 2007. (with E. Weyuker)
46. Automating Algorithms for the Identification of Fault-Prone Files. *Proc. ACM/Int. Symposium on Software Testing and Analysis (ISSTA07)*, London, England, July 2007. (with E. Weyuker and R. Bell)
45. Using Developer Information as a Factor for Fault Prediction. *Proc. IEEE/Third Int. Promise Workshop*, Minneapolis, May 2007. (with E. Weyuker and R. Bell)
44. An Industrial Research Program in Software Fault Prediction. *Proc. Workshop on Software Testing - From Research to Practice*, Hamburg, Germany, March 2007. (with E. Weyuker)
43. Experience Developing Software Using a Globally Distributed Workforce. *Proc. IEEE/First Int. Conference on Global Software Engineering (ICGSE06)*, Florianopolis, Brazil, October 2006. (with A. Avritzer and E. Weyuker)
42. Adapting a Fault Prediction Model to Allow Widespread Usage *Proc. IEEE/Second Int. Promise Workshop*, Philadelphia, September, 2006. (with E. Weyuker and R. Bell)
41. On the Automation of Software Fault Prediction *Proc. IEEE/Testing: Academic and Industrial Conference - Practice and Research Techniques (TAIC PART)*, Windsor, England, August 2006. (with E. Weyuker)
40. Looking for Bugs in All the Right Places. *Proc. ACM/Int. Symposium on Software Testing and Analysis (ISSTA2006)*, Portland, Maine, July 2006. (with R. Bell and E. Weyuker)
39. Locating Where Faults Will Be. *Proc. ACM/IEEE/CRA Richard Tapia Celebration of Diversity in Computing*, Albuquerque, NM, Oct 2005. (with E. Weyuker and R. Bell)
38. A Different View of Fault Prediction *Proc. 29th IEEE Annual Int. Computer Software and Applications Conference (COMPSAC 2005)*, Edinburgh, Scotland, July, 2005. (with E. Weyuker, R. Ostrand and R. Bell)
37. Predicting the Location and Number of Faults in Large Software Systems *IEEE Trans. on Software Engineering*, Vol 31, No 4, April 2005. (with E. Weyuker and R. Bell)

36. Difficulties Encountered Doing Empirical Studies in an Industrial Environment *Proc. 15th IEEE Int. Symposium of Software Reliability Engineering* Saint-Malo, France, November 2004. (with E. Weyuker)
35. Challenges in Predicting the Location of Faults in Large Software Systems. *IEEE/Proc. Workshop on Predictive Software Models (PSM04)*, Chicago, September 2004. (with R. Bell and E. Weyuker)
34. Where the Bugs Are. *Proc. ACM/Int. Symposium on Software Testing and Analysis (ISSTA2004)*, Boston, MA, July 2004. (with E. Weyuker and R. Bell)
33. A Tool for Mining Defect-Tracking Systems to Predict Fault-Prone Files. *Proc. IEEE Int. Workshop on Mining Software Repositories (MSR2004)*, Edinburgh, May 2004, <http://msr.uwaterloo.ca/msr2004/>. (with E.J. Weyuker)
32. Using Static Analysis to Determine Where to Focus Dynamic Testing Effort. *Proc. IEEE Second Int. Workshop on Dynamic Analysis (WODA 2004)*, Edinburgh, May 2004, <http://www.cs.virginia.edu/woda/proceedings.pdf>. (with E. Weyuker and R. Bell)
31. Identifying Fault-Prone Files in Large Software Systems. *Proc. 7th IASTED Int. Conf. on Software Engineering and Applications, (SEA2003)*, Marina del Rey, CA, Nov. 2003. (with E.J. Weyuker)
30. Generating Formal Specifications from Test Information. *Proc. Workshop on Formal Approaches to Testing*, Masaryk University, Brno, August 2002.
29. The Distribution of Faults in a Large Industrial Software System. *Proc. ACM/Int. Symposium on Software Testing and Analysis (ISSTA2002)*, Rome, Italy, July 2002, pp. 55-64. (with E. Weyuker)
28. Articles published in *Encyclopedia of Software Engineering, 2nd edition*, ed. J. Marciniak, Wiley-Interscience, 2002.
 - Black-Box Testing
 - White-Box Testing
 - Categories of Testing
27. Confessions of an Empirical Researcher, or What Some People Won't Do. *Proc. 5th Int. Conference on Empirical Assessment and Evaluation of Software Engineering (EASE2001)*, U. Keele, England, April, 2001. (with E.J. Weyuker)
26. A Learning Environment for Software Testers at AT&T. *Proc. Workshop on Learning Software Organizations*, Oulu, Finland, June 2000, pp.47-53. (with E.J. Weyuker)
25. Clearing a Career Path for Software Testers. *IEEE/Software*, March/April 2000, pp. 76-82. (with E. Weyuker, R. Prasad, J. Brophy)
24. How Can We Encourage Practitioners to Respond to Empirical Evidence? *Proc. Fifth IEEE Workshop on Empirical Studies of Software Maintenance (WESS99)*, Oxford, Eng., Sept. 1999. (with E. Weyuker)
23. Test Generation with the Category-Partition Method, *ATTEST99: AT&T Software Testing Conference*, Middletown, NJ, June 1999.
22. A Visual Test Development Environment for GUI Systems. *Proc. Software Quality Week*, May 1998. (with H. Foster, T. Goradia, W. Szermer).
21. A Visual Test Development Environment for GUI Systems. *Proc. ACM Symp. on Software Testing and Analysis*, March 1998. (with A. Anodide, H. Foster, T. Goradia).
20. Experiments on the Effectiveness of Dataflow and Controlflow-Based Test Adequacy Criteria. *Proc. IEEE/ACM Int. Conference on Software Engineering*, May 1994. (with M. Hutchins, H. Foster, T. Goradia).
19. Data Flow-Based Test Adequacy Analysis for Languages with Pointers, *Proc. of the Fourth ACM SIGSOFT Symposium on Software Testing, Analysis and Verification (TAV4)*, Victoria, B.C., Canada, Oct 1991, pp. 74-86. (with E. Weyuker)

18. Data-flow Based Testing Techniques. *Proc. Pacific Northwest Software Quality Conf.*, Portland, OR, Oct 1990.
17. Automatic Generation of Test Scripts from Formal Test Specifications. *Proc. ACM Symposium on Testing, Analysis and Verification*, Dec 1989. (with M.J. Balcer and W. Hasling).
16. Using Data Flow Analysis for Regression Testing. *Proc. of the Sixth Annual Pacific Northwest Software Quality Conference*, Portland, Oregon, September 1988, pp. 233-247. (with E. Weyuker).
15. The Category-Partition Method for Specifying and Generating Functional Tests. *Comm. ACM*, June 1988. (with M.J. Balcer).
14. Design for a Tool to Manage Specification-Based Testing. *Proc. of the IEEE Workshop on Software Testing*, Banff, Canada, July 1986, pp. 41-50. (with E. Weyuker and R. Sigal)
13. Collecting and Categorizing Software Error Data in an Industrial Environment. *J. Systems and Software*, Vol.4, 1984, pp. 289-300. (with E.J. Weyuker)
12. Software Error Data Collection and Categorization, *Proc. of the Seventh Annual Software Engineering Workshop*, Goddard Space Flight Center, Greenbelt, Maryland, December 1982. (with E.J. Weyuker)
11. The Use of Revealing Subdomains, *Proc. of the IEEE Workshop on Effectiveness of Testing and Proving Methods*, Catalina Island, May 1982. (with E.J. Weyuker)
10. Experiments With Pseudo-Oracles, *Proc. of the IEEE Workshop on Effectiveness of Testing and Proving Methods*, Catalina Island, May 1982. (with M.D. Davis and E.J. Weyuker)
9. Software Error Data Collection and Analysis, *Proc. of the Conference on Computers and Communications*, Phoenix, May 1982, pp. 209-213. (with E.J. Weyuker)
8. Parsing Regular Grammars with Finite Lookahead, *Acta Informatica*, Volume 16, November 1981, pp. 125-138. (with E.J. Weyuker and M.C. Paull)
7. Current Directions in the Theory of Testing, *Proc. CompSAC 80 Conference*, Chicago, October 1980, pp. 386-389. (with E.J. Weyuker)
6. Theories of Program Testing and the Application of Revealing Subdomains, *IEEE Transactions on Software Engineering*, May, 1980, pp. 236-245. (with E.J. Weyuker)
5. Error Based Program Testing, *Proc. of Thirteenth Conference on Information Sciences and Systems*, The Johns Hopkins University, Baltimore, March 1979, pp. 444-449. (with E.J. Weyuker)
4. Remarks on the Theory of Test Data Selection, *Proc. of the IEEE Workshop on Software Testing and Test Documentation*, Fort Lauderdale, Florida, December 1978, pp. 1-18. (with E.J. Weyuker)
3. Fixed Lookahead Parsing of Regular Grammars by Finite State Machines. *Proc. of the Twelfth Annual Conference on Information Sciences and Systems*, The Johns Hopkins University Baltimore, Maryland, March 1978, pp. 340-345. (with E.J. Weyuker and M.C. Paull)
2. Systematic and Simple Proofs of Loop Programs. *Proceedings of Conference on Information Sciences and Systems*, The Johns Hopkins University, Baltimore, March 1977.
1. Pattern Reproduction in Tessellation Automata of Arbitrary Dimension. *J. Computer and System Sciences* 1971, pp. 623-628.