

## REFERENCES

1. Aggarwal K.K., Yogesh Singh, Arvinder Kaur, Ruchika Malhotra, “Software Design Metrics for Object-Oriented Software”, *Journal of Object Technology*, Vol. 6, no. 1, pp. 121-138, 2007.
2. Albrecht A. and Gaffney J., “Software Function, Source Lines of Code, and Development Effort Prediction: A Software Science Validation”, *IEEE Transactions on Software Engineering*, Vol. SE-9, No. 6, pp 639-648, 1983.
3. AmjanShaik, Reddy C.R.K. Dr., Damodaram A. Dr., “Object Oriented Software Metrics and Quality Assessment: Current State of the Art”, *International Journal of Computer Applications (0975 – 8887)*, Volume 37 – No.11, Jan 2012.
4. Andreas Menychtas and Kleopatra G. Konstanteli, “Fault Detection and Recovery Mechanisms and Techniques for Service Oriented Infrastructures”, *IGI Global*, DOI: 10.4018/978-1-60960-827-9.ch014, pp. 259-262, 2012.
5. Antoniol G., Fiutem R. and Lokan C. “Object-Oriented Function Points: An Empirical Validation”, In *Kluwer Academic Publishers*, pp. 225-254, 2003.
6. Arnaoudova V., Eshkevari L., Oliveto R. and Guéhéneuc, “Physical and conceptual identifier dispersion: Measures and relation to fault proneness”, Y.-G.; Antoniol G.; SOCCER Lab., Ecole Polytech. de Montreal, Montreal, QC, Canada. *Software Maintenance (ICSM), 2010 IEEE International Conference on 12-18 Sept. 2010*.
7. Arockiam L., Kasmir Raja. S.V and Xavier. C, “Measuring Object Oriented Software by their components”, *Proceedings of the National Convention on Information technology*, 2001, pp. 5-12.
8. Backus, John W., “The Syntax and Semantics of the Proposed International Algebraic Language of Zürich ACM-GAMM Conference”. *Proceedings of the International Conference on Information Processing, UNESCO*, 1959.

9. Baillot P., "Stratified coherence spaces: a denotational semantics in light linear logic", *theoretical computer science*, 318 (1-2), pp 29-55, 2004.
10. Basili V. R., Briand L. C. , and Melo W. L. , "A Validation of Object-Oriented Design Metrics as Quality Indicators", In *IEEE Transactions on Software Engineering*, volume 22, pages 751–761, Oct. 1996.
11. Behoren C.A., "Measuring the productivity of computer system development", *IEEE Transaction on software engineering tol. SE9,M6*,pp 648-657, Nov 1983.
12. Bessani A.N., Fac. of Sci., Univ. of Lisbon, Lisbon, Portugal, "From Byzantine fault tolerance to intrusion tolerance (a position paper)" , *Dependable Systems and Networks Workshops (DSN-W)*, 2011 *IEEE/IFIP 41st International Conference on June 2011*.
13. Boehm B.W., Brown J.R, Kaspar H., lapow M., Mcleod G. and Merritt M., *characteristics of software quality*, north Holland, 1978.
14. Briand L., Wiist J. and Lounis H. "Using coupling measurement for impact analysis in object-oriented systems", *Proceedings of the 19th International Conference on Software Maintenance*, pp. 475-482, 1999.
15. Briand L.C. and Wust J. "Modeling development effort in object oriented system using design properties", *IEEE Transactions on Software Engineering*, Vol. 27, No. 11, pp. 963-986, 2001.
16. Briand L.C., Bunse C. and Daly J.W. "A controlled experiment for evaluating quality guidelines on maintainability of object oriented design", *IEEE Transactions on Software Engineering*, Vol. 2, No. 6, pp. 513-530, 2001.
17. Brito e Abreu, F, "MOOD – Metrics for Object-Oriented Design", *OOPSLA Workshop on Pragmatic and Theoretical Directions in Object-Oriented Software Metrics*, 1994.
18. Card D.N and Agresti W., "Measuring software design complexity", *Journal of system and software DOZ* 10.1016/0164-1212 (88) 50021-0 Jan 1988.

19. Cartwright M and M. Shepperd. An Empirical Investigation of an Object-Oriented Software System. *IEEE Transactions on Software Engineering*, 26(8):786–796, 2000.
20. Chen Liming Algirdas Avlzlenls, “N-version Programming: A Fault-Tolerance Approach to Reliability of Software Operation”, University of California, Los Angeles, CA 90024 USA, Xerox Corporation Computer Science Department, El Segundo, CA 90245 USA, 1996.
21. Chidamber S.R. and C. F. Kemerer, “Towards a metrics suite for object-oriented design”, in Proc. 6<sup>th</sup> OOPSLA Conference, ACM 91, pp. 197-211, 1991.
22. Chidamber S.R. and Kemerer C. F., “A Metrics Suite for Object Oriented Design”, *IEEE Transactions on Software Engineering*, Vol. 20, No. 6, pp. 476–493, 1994.
23. Christensen K., Fitsos G. P., and Smith C. P. “A Perspective on Software Science”, *IBM Systems J.* 20, 4 (1981), pp. 372-387.
24. Coleman D., B. Lowther, and P. Oman. The Application of Software Maintainability Models in Industrial Software Systems. *The Journal of Systems and Software*, 29(1):3–16, 1995.
25. Curtis B., Sheppard S. B., and Milliman P. “Third Time Charm: Stronger Prediction of Programmer Performance by Software Complexity Metrics ”, Proc. 4th Int. Conf. on Software Engineering. New York: IEEE, Sept. 1979, pp. 356-360.
26. DeMarco T, “Controlling Software Projects: Management, Measurement and Estimation”, Yourdon Press New York, 1982.
27. Demarco T., “An Algorithm for Sizing Software Products”, performance evaluation review, vol.12, No2, pp. b-22, Spring summer 1984.
28. Denney Richard, “Succeeding with Use Cases: Working Smart to Deliver Quality”, Addison-Wesley Professional Publishing, 2005, ISBN 0-321-31643-6.
29. Douglas Schmidt C. and Adam Porter, “Leveraging Open-Source Communities To Improve the Quality and Performance of Open-Source Software”, schmidt@uci.edu aporter@cs.umd.edu, Electrical & Computer Engineering Department Computer Science Department, University of California, Irvine University of Maryland, 2000.

30. Dr. E. Chandra I, P. Edith Linda, "Class Break Point Determination Using CK Metrics Threshold", Vol.10 Issue 14 (Ver.1.0) pp.73-77, November 2010
31. Duffy, David A., "Principles of Automated Theorem Proving", John Wiley & Sons, 1991.
32. Edgar Gabriel, Graham Fagg E., George Bosilca<sup>1</sup>, Thara Angskun, Jack Dongarra J., Jeffrey Squyres M., Visual Sahay, Prabhanjan Kambadur, Brian Barrett, Andrew Lumsdaine, Ralph Castain H., David Daniel J., Richard Graham L. and Timothy Woodall S. "Open MPI: Goals, Concept, and Design of a Next Generation MPI Implementation", 1 Innovative Computing Laboratory, University of Tennessee, {egabriel, fagg, bosilca, anskun, dongarra}@cs.utk.edu 2 Open System Laboratory, Indiana University, 2004.
33. Frederick T. Sheldon, Kshamta Jerath, Hong Chung, Metrics for Maintainability of Class Inheritance Hierarchies, Journal of Software Maintenance and Evolution: Research and Practice, Vol. 14, Issue 3, pp. 147-160, 2002.
34. Gilles Kahn, "Natural Semantics", Proceedings of the 4th Annual Symposium on Theoretical Aspects of Computer Science, Springer-Verlag, London, 1987.
35. Glen Dobson, Computing Department, Lancaster University, "Using WS-BPEL to Implement Software Fault Tolerance for Web Services", g.dobson@lancs.ac.uk, 2003.
36. Halstead M. H., "Elements of Software Science", New York: Elsevier North-Holland, 1977.
37. Hanny J.E., Sjoberg D.I.K, and Dyba T. "A Systematic Review of Theory use in Software Engineering Experiments", IEEE Transaction on Software Engineering, Vol. 33, No. 2, pp. 87-107, 2007.
38. Hector Olague M., Letha Eitzkorn, Sampson Gholston H. and Stephen Quattlebaum "Empirical Validation of Three Software Metrics Suites to Predict Fault-Proneness of Object-Oriented Classes Developed Using Highly Iterative or Agile Software Development Processes", IEEE transaction on software engineering, Vol. 33, No. 6, pp. 402-419, 2007.

39. Henderson-Sellers, B. "Object-Oriented Metrics: Measures of Complexity", Prentice Hall, New Jersey, 1996.
40. Hendry s and Kafura K., Software Structure Matrix Based on Information that IEEE Trarate in Software Company pp 510-510, 1982.
41. Hitz M. and Montazeri B. "Measuring coupling and cohesion in object oriented systems", Proceedings of the International Symposium on Applied Corporate Computing, 1995.
42. Hitz V, and B. Montazeri. Chidamber and Kemerer's Metrics Suite: A Measurement Theory Perspective. IEEE Transactions on Software Engineering, 22(4):267–271, 1996.
43. <http://dl.acm.org/citation.cfm?id=820036>
44. [http://en.wikipedia.org/wiki/Comparison\\_of\\_open-source\\_configuration\\_management\\_software](http://en.wikipedia.org/wiki/Comparison_of_open-source_configuration_management_software)
45. [http://toolbar.netcraft.com/report\\_url](http://toolbar.netcraft.com/report_url)
46. <http://w3techs.com/technologies>
47. <http://w3techs.com/technologies/details/cp-javaruntime/all/all>
48. <http://w3techs.com/technologies/details/pl-php/all/all>
49. <http://w3techs.com/technologies/details/ws-apache/all/all>
50. <http://www.javaworld.com/javaworld/jw-05-2004/jw-0517-java.net.html>
51. IEEE, "IEEE Standard Glossary of Software Engineering Terminology", IEEE Std. 610.12-1990. Institute of Electrical and Electronics Engineers, 1990.
52. International Organization for Standardization/International Electro technical Commission (ISO/IEC), "Information Technology - Software Product Evaluation- Quality Characteristics and Guidelines for their use," 1991.
53. Jelinski Z., and P.B. Moranda,"Software reliability research", in Statistical Computer Performance Evaluation, W. Freiberger, Ed., New York: Academic Press, 1972, pp. 465-484.

54. Jiang Zheng, Nachiappan Nagappan, John P. Hudepohl, "On the value of Static Analysis for Fault Detection in Software", IEEE Transactions on Software Engineering, April 2006.
55. Jie Xu, Danny Ho and Luiz Fernando Capretz "An Empirical Validation of Object-Oriented Design Metrics for Fault Prediction", Journal of Computer Science, Vol. 4, No. 7, pp. 571-577, 2008.
56. John Reppy, "Concurrent ML: Design, Application and Semantics", Springer-Verlag, Lecture Notes in Computer Science, Vol. 693, 1993.
57. Junguo Li, Gang Huang, Jian Zou and Hong Mei, "Failure Analysis of Open Source J2EE Application Servers", Published on 2007.
58. Kafura , Dennis G., Reddy, Geereddy R, "The Use of Software Quality Metrics in Software Maintenance", Technical Report TR-85-33, Computer Science, Virginia Polytechnic Institute and State University, 2005.
59. Kafura D. and Canning J., "A Validation of Software Metrics Using Many Metrics and Two Resources", Proc. 8th Intl. Conf. on Software Engineering. Washington, D. C.: IEEE Computer Society Press, 1985, 378-385.
60. Kafura D. and Reddy G. R., "The Use of Software Complexity Metrics in Software Maintenance " ,IEEE Trans. Software Eng. SE-13, 3 (March 1987), 335-343.
61. Kamaljit Kaur, Kirti Minhas, Neha Mehan and Namita Kakkar, "Static and Dynamic Complexity Analysis of Software Metrics", World Academy of Science, Engineering and Technology, pp. 159-161, 2009.
62. Kemerer C.F., "An empirical validation of software cost estimation models", Communications of the ACM, Vol. 30, No. 5, pp. 416-429, 1987.
63. Kitchen Ham B.A., Pfleeger S.L., Pickard L.M, Jones P.W., Hoaglin D.C., El-Emam K. and Rosenberg J. "Preliminary Guidelines For Empirical Research In Software Engineering", IEEE Transaction on Software Engineering, Vol. 28, No. 8, pp. 721-734, 2002.
64. Korson T. and McGregor J.D. "Understanding Object-Oriented: A unifying Paradigm", Communications of the ACM, Vol. 33, No. 9, pp. 41-60, 1990.

65. Kristof De Buysere, Oliver Gajda, Ronald Kleverlaan and Dan Marom A., “Framework For European Crowd funding”, Attribution-Non-Commercial-Share Alike, ISBN 978-3-00-040193-0, 2012.
66. Kuhn D.R. and Reilly M.J., “An Investigation of the Applicability of Design of Experiments to Software Testing”, Proc. 27<sup>th</sup> NASA/IEEE Software Eng. Workshop, 2002.
67. Li H. F. and Cheung W. K., “An Empirical Study of Software Metrics.” IEEE Trans. Software Eng. SE- 13, 6 (June 1987), 697-708.
68. Lorenz M and Kidd J, “Object-Oriented Software Metrics”, Prentice Hall, 1994.
69. Maurice Hayward Halsted(1977) Elements of software science Amsterdam Elsevier North-Holland, inc ISBN 0-444-00206-7.
70. Mc Cabe T.J. and Watson A.H., — “Software Complexity, Crosstalk, J. Defense Software Eng., vol. 7, no. 12, pp. 5-9, Dec. 1994.
71. McCabe T., “A Software Complexity Measure”, IEEE Transactions on Software Engineering, Vol. 2, No.4, 1976, pp. 308-320.
72. McCall S.A, Richards P.K, and Waltor G.F, “Factors in Software Quality”, Naril tech information service, no Vol 1, 2, and 3, 1977.
73. Michael Holloway C., “Why Engineers Should Consider Formal Methods”, 16th Digital Avionics Systems Conference, 1997.
74. Mohammad Alshayeb and Wei Li, “An empirical study of system design instability metric and design evolution in an agile software process”, Journal of Systems and Software, Vol. 74, No. 3, pp. 269 - 274, 2005.
75. Mohammad Alshayeb and Wei Li, “An Empirical Validation of Object-Oriented Metrics in Two Different Iterative Software Processes”, IEEE transaction on software engineering, Vol. 12, No. 11, pp. 1043-1049, 2003.
76. Nair V.N., James D.A., Erlich W.K. and Zevallos J. “A Statistical Assessment of Some Software Testing Strategies and Application of Experimental Design Techniques”, Statistica Sinica, Vol. 8, No. 1, pp. 165- 184, 1998.

77. Naughton, T.J. ; Agarwal, P. et. al, "Realization of user level fault tolerant policy management through a holistic approach for fault correlation", Policies for Distributed Systems and Networks (POLICY), 2011 IEEE International Symposium June 2011, Page(s):17 - 24 ,ISBN: 978-1-4244-9879-6 .
78. Oviedo E.I. "Control flow, Data Flow and Program Complexity", Software Engineering Metrics I: measures and validations, McGraw-Hill, Inc., pp. 52- 65, 1993.
79. Pandey R. K. and Vinay Tiwari, "Reliability Issues in Open Source Software" in 2011.
80. Paulson J.W., Succi G. and Eberlein A. "An empirical study of open-source and closed-source software products", IEEE Transactions on Software Engineering, 2004.
81. Pearse T. and Omen P. "Maintainability Measurements on Industrial Source Code Maintenance Activities", Proceedings of the International Conference on Software Maintenance, pp. 295-303, 1995.
82. Pfleeger S. L., "Software Engineering - Theory and Practise", Prentice-Hall, 1998.
83. Potier D., Albin J. L., Ferreol R., and Bilodeau A. "Experiments with Computer Software Complexity and Reliability", Proc. 6th Intl. Conf. on Software Engineering. New York: IEEE, Sept. 1982, 94-103.
84. Ramanath Subramanyam and Krishnan, M.S. "Empirical Analysis of CK Metrics for Object-Oriented Design Complexity: Implications for Software Defects", IEEE transaction on software engineering, Vol. 29, No. 4, pp. 297-310, 2003.
85. Reema ajmera, Ripu R. Sinha, Lamba C.S. Dr, "Comparative analysis of Software testing measurement technique", International Journal of Engineering and Innovative Technology (IJEIT)Volume 1, Issue 2, ISSN: 2277-3754, pp. 70-80, 2012.
86. Richard Kuhn D., Dolores Wallace R. and Albert Gallo M. "Software Fault Interactions and Implications for Software Testing", in June, 2004.

87. Richard Kuhn Michael D. and Reilly J. "An Investigation of the Applicability of Design of Experiments to Software Testing", National Institute of Standards and Technology, Gaithersburg, MD 20899. kuhn@nist.gov michael.reilly@nist.gov, 2006.
88. Robert C. Martin, Object Oriented Design Quality Metrics an Analysis of Dependencies, <http://www.objectmentor.com/resources/articles/oodmetric.pdf>, 2004-06-04.
89. Robyn R. Lutz, "Software Engineering for Safety: A Roadmap", Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA 91109-8099, 2000.
90. Rodriguez V. and Tsai W.-T., "Software Metrics Interpretation Through Experimentation", Proc. COMPSAC 86. Washington, D. C.: IEEE Computer Society Press, Oct. 1986, 368-374.
91. Rudolf Ferenc, Istvan Siket and Tibor Gyimothy, "Extracting Facts from Open Source Software", University of Szeged, Department of Software Engineering, fferenc|siket|gyimig@inf.u-szeged.hu, in 2004.
92. Samuel B. Lyerly, "The average spearman rank correlation coefficient", Springer, Volume 17, Issue 4, pp 421-428, 1952
93. Shen V. Y., Yu T. J., Thebaut S. M., and Paulsen L. R., "Identifying Error-Prone Software—An Empirical Study." IEEE Trans. Software Eng. SE-11, 4 (April 1985), 317-324.
94. Shooman M.L., "Probabilistic models for software reliability prediction", in Statistical Computer Performance Evaluation, W. Freidberger, Ed., New York: Academic Press, 1972, pp. 485-502.
95. Sten Agerholm and Peter G. Larsen, "A Lightweight Approach to Formal Methods", Proceedings of the International Workshop on Current Trends in Applied Formal Methods, Springer-Verlag, 1998.
96. Subhachandra Chandra and Peter Chen M. "Whither Generic Recovery from Application Faults? A Fault Study using Open-Source Software", Computer Science and Engineering Division, Department of Electrical Engineering and Computer Science, University of Michigan, {schandra,pmchen} @eecs.umich.edu, <http://www.eecs.umich.edu/Rio>, 2000.

97. Tegarden D. P, Sheetz S. D and Monarchi D. E, "Measuring Object-Oriented System Complexity", Proceedings of the First Workshop on Information Technologies and Systems, 1991, pp. 14-15.
98. Tibor Gyimothy, Rudolf Ferenc and Istvan Siket, "Empirical Validation of Object-Oriented Metrics on Open Source Software for Fault Prediction", IEEE Transactions on Software Engineering, Vol. 31, No. 10, 2005.
99. Tsai W.T, Lopex M.A, Rodrigues V. and Volovik D., "An Approach Measuring Data Structure Complexity", IEEE Computer Society Press, Washington, 1986, pp. 240-246.
100. Victor R. Basili, Lionel Briand and Walcelio Melo L., "A Validation of Object Oriented Design Metrics as Quality Indicators", Technical Report, April 1995.
101. Vim George and Rayford Vaughn, "Application of Light Weight Normal Method in Requirement Engineering", Gromstalk, the journal of Defense software security, Jan 2003.
102. Wallace D.R., Kuhn D.R., "Failure Modes in Medical Device Software: an Analysis of 15 Years of Recall Data", International Journal of Reliability, Quality and Safety Engineering, Vol. 8, no. 4, 2001.
103. Weyuker E. J, "Evaluating Software Complexity Measures", IEEE Transactions on Software Engineering, Vol. 21, 1988, pp. 1357-1365.
104. Wos Larry, Overbeek Ross, Lusk Ewing and Boyle Jim, "Automated Reasoning: Introduction and Applications", (2nd ed.), McGraw-Hill, 1992.
105. Yau S. S. and Collofello J. S., "Some Stability Measures for Software Maintenance." IEEE Trans. Software Eng. SE-6, 6 (Nov. 1980), 545-552.
106. Yuming Zhou and Hareton Leung "Empirical Analysis of Object-Oriented Design Metrics for Predicting High and Low Severity Faults", IEEE transaction on software engineering, Vol. 32, No. 10, pp. 771-789, 2006.

107. Zhenmin Li, Lin Tan, Xuanhui Wang, Shan Lu, Yuanyuan Zhou and Chengxiang Zhai, “Have Things Changed Now? – An Empirical Study of Bug Characteristics in Modern Open Source Software”, Department of Computer Science, University of Illinois at UrbanaChampaign Urbana, IL 61801 , fzli4, lintan2, xwang20, shanlu, yyzhou, czhaig@cs.uiuc.edu, 1998.
108. Zuse Ho., “Software Complexity: Measures and Methods”, Walter de Gruyter, Publisher, 1990.