Publications

Books

1. Runeson, P., ed. (2000). *Industriell programvaruteknik - forskningsresultat i kortform (Software engineering - research results in brief, in Swedish)*. ISBN B-91-7318-3350. NUTEK B2000:1 (The National Board for Industrial and Technical Development).

2. Wohlin, C., P. Runeson, M. Höst, M. C. Ohlsson, B. Regnell, and A. Wesslén (2000). *Experimentation in Software Engineering: An Introduction*. The Kluwer International Series In Software Engineering. Kluwer.

3. Höst, M., B. Regnell, and P. Runeson (2006). *Att genomföra examensarbete*. Studentlitteratur.

4. Runeson, P., M. Höst, A. Rainer, and B. Regnell (2012). *Case Study Research in Software Engineering – Guidelines and Examples*. Wiley.

5. Wohlin, C., P. Runeson, M. Höst, M. C. Ohlsson, B. Regnell, and A. Wesslén (2012). *Experimentation in Software Engineering*. One of the top 50% most downloaded Springer eBooks 2012-2013. Springer. DOI: 10.1007/978-3-642-29044-2.

6. Larsson, S. and P. Runeson (2013). *Det digitala samhället*. Pufendorfinstitutet vid Lunds universitet. ISBN: 978-91-979893-1-2.

7. Larsson, S. and P. Runeson, eds. (2014). *DigiTrust: Tillit i det digitala – Tvärvetenskapliga perspektiv från ett forskningsprojekt*. Pufendorfinstitutet vid Lunds universitet. ISBN: 978-91-979893-6-7. http://lup.lub.lu.se/record/4461283.

8. Wohlin, C., P. Runeson, M. Höst, M. C. Ohlsson, B. Regnell, and A. Wesslén (2015). *Experimentation in Software Engineering (in Chinese)*. 软:软. China Machine Press. DOI: 10.1007/978-3-642-29044-2.

PhD thesis

1. Runeson, P. (1998). "Models for Estimation of Software Faults and Failures in Inspection and Test". ISSN 1101-3931, ISRN LUTEDX/TETS–1034–SE+222P. Box 118, SE-221 00 Lund, Sweden: Lund University.

Journal papers

1. Wohlin, C. and P. Runeson (1994). Certification of Software Components. *IEEE Transactions on Software Engineering* **20**(6), 494–499.

2. Runeson, P. and C. Wohlin (1995). Statistical Usage Testing for Software Reliability Control. *Informatica* **19**(2), 195–207.

3. Wohlin, C., P. Runeson, and J. Brantestam (1995). An Experimental Evaluation of Capture-Recapture in Software Inspections. *Software Testing, Verification and Reliability* (5), 213–232.

4. Wohlin, C., P. Runeson, and A. Wesslén (1996). Software Reliability Estimations through Usage Analysis of Specifications and Designs. *International Journal of Reliability, Quality and Safety Engineering* **3**(2), 101–117.

5. Runeson, P. and C. Wohlin (1998). An Experimental Evaluation of an Experience-Based Capture-Recapture Method in Software Code Inspections. *Empirical Software Engineering* **3**(4), 381–406.

6. Bratthall, L., P. Runeson, K. Adelswärd-Bruck, and W. Eriksson (2000). A Survey of Lead-Time Challenges in the Development and Evolution of Distributed Real-Time Systems. *Information and Software Technology* **42**(13), 947–958.

7. Regnell, B., P. Runeson, and T. Thelin (2000). Are the perspectives really different? Further experimentation on scenario-based reading of requirements. *Empirical Software Engineering* **5**(4), 331–356.

8. Regnell, B., P. Runeson, and C. Wohlin (2000). Towards integration of use case modelling and usage-based testing. *Journal of Systems and Software* **50**(2), 117–130.

9. Thelin, T. and P. Runeson (2000). Robust Estimators of Fault Content with Capture-Recapture and Detection Profile Estimators. *Journal of Systems and Software* **52**(3-4), 139–148.

10. Thelin, T., P. Runeson, and B. Regnell (2001). Usage-Based Reading – An Experiment to Guide Reviewers with Use Cases. *Information and Software Technology* **43**(15), 925–938.

11. Karlström, D., P. Runeson, and C. Wohlin (2002). Aggregating Viewpoints for Strategic Software Process Improvement. *IEE Proceedings Software* **149**(5), 143–152.

12. Thelin, T., H Petersson, and P. Runeson (2002). Confidence intervals for capture-recapture estimations in software inspections. *Information and Software Technology* **44**(12), 683–702.

13. Berling, T. and P. Runeson (2003). Efficient evaluation of multifactor dependent system performance using fractional factorial design. *IEEE Transactions on Software Engineering* **29**(9), 769–781.

14. Berling, T. and P. Runeson (2003). Evaluation of a Perspective Based Review Method Applied in an Industrial Setting. *IEE proceedings Software* **150**(3), 177–184.

15. Runeson, P., C. Andersson, and M. Höst (2003). Test processes in software product evolution - a qualitative survey on the state of practice. *Journal of Software Maintenance and Evolution: Research and Practice* **15**(1), 41–59.

16. Thelin, T., P. Runeson, and C. Wohlin (2003). An experimental comparison of usage-based and checklist-based reading. *IEEE Transactions on Software Engineering* **29**(8), 687–704.

17. Thelin, T., P. Runeson, and C. Wohlin (2003). Prioritized use cases as a vehicle for software inspections. *IEEE Software* **20**(4), 30–33.

18. Petersson, H., T. Thelin, P. Runeson, and C. Wohlin (2004). Capture-recapture in software inspections after 10 years research - theory, evaluation and application. *Journal of Systems and Software* **72**(2), 249–264.

19. Thelin, T., H. Petersson, P. Runeson, and C. Wohlin (2004). Applying sampling to improve software inspections. *Journal of Systems and Software* **73**(2), 257–269.

20. Thelin, T., P. Runeson, C Wohlin, T Olsson, and C. Andersson (2004). Evaluation of usage-based reading - Conclusions after three experiments. *Empirical Software Engineering* **9**(1-2), 77–110.

21. Andrews, A, E Mancebo, P. Runeson, and R France (2005). A framework for design tradeoffs. *Software Quality Journal* **13**(4), 377–405.

22. Karlström, D. and P. Runeson (2005). Combining agile methods with stage-gate project management. *IEEE software* **22**(3), 43–49.

23. Karlström, D., P. Runeson, and S. Nordén (2005). A minimal test practice framework for emerging software organizations. *Software Testing, Verification and Reliability* **15**(3), 145–166.

24. Karlström, D. and P. Runeson (2006). Integrating agile software development into stage-gate managed product development. *Empirical Software Engineering* **11**(2), 203–225.

25. Runeson, P. (2006). A survey of unit testing practices. *IEEE Software* **23**(4), 22–+.

26. Runeson, P., C. Andersson, T. Thelin, A Andrews, and T. Berling (2006). What do we know about defect detection methods. *IEEE Software* **23**(3), 82–+.

27. Andersson, C. and P. Runeson (2007). A replicated quantitative analysis of fault distributions in complex software systems. *IEEE Transactions on Software Engineering* **33**(5), 273–286.

28. Andersson, C. and P. Runeson (2007). A Spiral Process Model for Case Studies on Software Quality Monitoring - Method and Metrics. *Software Process Improvement and Practice* **12**(2), 125–140.

29. Skoglund, M. and P. Runeson (2007). Improving Class Firewall Regression Test Selection by Removing the Class Firewall. *International Journal of Software Engineering and Knowledge Engineering* **17**(3), 359–378.

30. Runeson, P. and M. Höst (2009). Guidelines for conducting and reporting case study research in software engineering. *Empirical Software Engineering* **14**(2). The most cited EMSE paper ever., 131–164.

31. Engström, E., P. Runeson, and M. Skoglund (2010). A systematic review on regression test selection techniques. *Information and Software Technology* **52**(1), 14–30.

32. da Mota Silveira Neto, P., P. Runeson, I. do Carmo Machado, E. de Almeida, S. de Lemos Meira, and E. Engström (2011). Testing Software Product Lines. *IEEE Software*, 16–20.

33. Engström, E. and P. Runeson (2011). Software product line testing – A systematic mapping study. *Information and Software Technology* **53**(1), 2–13.

34. Kitchenham, B. A., D. I. K. Sjøberg, T. Dybå, D. Pfahl, P. Brereton, D. Budgen, M. Höst, and P. Runeson (2012). Three Empirical Studies on the Agreement of Reviewers about the Quality of Software Engineering Experiments. *Information and Software Technology* **54**(8), 804–819.

35. Engström, E. and P. Runeson (2013). Test Overlay in an Emerging Software Product Line – An Industrial Case Study. *Information and Software Technology* **55**, 581–594.

36. Galinac Grbac, T., P. Runeson, and D. Huljenić (2013). A Second Replicated Quantitative Analysis of Fault Distributions in Complex Software Systems. *IEEE Transactions on Software Engineering* **39**(4), 462–476.

37. Hassan, A. E., A. Hindle, P. Runeson, M. Shepperd, P. Devanbu, and S. Kim (2013). Roundtable: What's Next in Software Analytics. *IEEE Software* **30**(4), 53–56.

38. Kitchenham, B. A., D. I. K. Sjøberg, T. Dybå, P. Brereton, D. Budgen, M. Höst, and P. Runeson (2013). Trends in the quality of human-intensive software engineering experiments – A quasi-experiment. *IEEE Transactions on Software Engineering* **39**(7), 1002–1017.

39. Wohlin, C., P. Runeson, P. da Mota Silveira Neto, E. Engström, I. do Carmo Machado, and E. de Almeida (2013). On the Reliability of Mapping Studies in Software Engineering. *Journal of Systems and Software* **86**(10), 2594–2610.

40. Bjarnason, E., P. Runeson, M. Borg, M. Unterkalmsteiner, E. Engström, B. Regnell, G. Sabaliauskaite, A. Loconsole, T. Gorschek, and R. Feldt (2014). Challenges and Practices in Aligning Requirements and Verification and Validation: An Industrial Multi-Unit Case Study. *Empirical Software Engineering* **19**(6), 1809–1855.

41. Borg, M., P. Runeson, and A. Ardö (2014). Recovering from a Decade: A Systematic Map of Information Retrieval Approaches to Software Traceability. *Empirical Software Engineering* **19**(6), 1565–1616.

42. Petersen, K., P. Roos, S. Nyström, and P. Runeson (2014). Early Identification of Bottlenecks in System of Systems Software Development. *Journal of Software: Evolution and Process* **26**(12), 1150–1171.

43. Runeson, P., A. Stefik, and A. Andrews (2014). Variation Factors in the Design and Analysis of Replicated Controlled Experiments – Three (Dis)similar Studies on Inspections versus Unit Testing. *Empirical Software Engineering* **19**(6), 1781–1808.

44. Wnuk, K., P. Runeson, M. Lantz, and O. Weijden (2014). Bridges and Barriers to Software Ecosystem Participation - a Case Study. *Information and Software Technology* **56**(11), 1493–1507.

45. Cruzes, D. S., T. Dybå, P. Runeson, and M. Höst (2015). Case Studies Synthesis: A Thematic, Cross-Case, and Narrative Synthesis Worked Example. *Empirical Software Engineering* **20**(6), 1634–1665.

46. Galinac Grbac, T., P. Runeson, and D. Huljenić (2015). Quantitative analysis of unit verification as predictor in large scale software enginering. *Software Quality Journal* **To appear**.

47. Bjarnason, E., K. Smolander, E. Engström, and P. Runeson (2016). A Theory of Distances in Software Development. *Information and Software Technology* **70**(204-219).

48. Borg, M., K. Wnuk, B. Regnell, and P. Runeson (2016). Supporting Change Impact Analysis Using a Recommendation System: An Industrial Case Study in a Safety-Critical Context. *IEEE Transactions on Software Engineering* **accepted**.

49. Jonsson, L., M. Borg, D. Broman, K. Sandahl, S. Eldh, and P. Runeson (2016). Automated Bug Assignment: Ensemble-based Machine Learning in Large Scale Industrial Contexts. *Empirical Software Engineering* **21**(4), 1579–1585.

50. Munir, H., K. Wnuk, and P. Runeson (2016). Open Innovation in Software Engineering: A Systematic Mapping Study. *Empirical Software Engineering* **21**(2). Invited as journal first presentation at ICSE 2016, 684–723.

Book chapters

1. Broomé, M. and P. Runeson (2000). "Technical requirements for the implementation of an experience base". In: *Learning Software Organizations*. Ed. by G. Ruhe and F. Bomarius. Vol. 1756. Lecture Notes in Computer Science. 10.1007/BFb0101415. Springer Berlin / Heidelberg, pp.87–102. ISBN: 978-3-540-41430-8. `http://dx.doi.org/10.1007/BFb0101415`.

2. Wohlin, C., M. Höst, A. Wesslén, and P. Runeson (2002). "Software Reliability". In: *Encyclopedia of Physical Science and Technology*. Vol. 15. Elsevier Science Ltd., pp.25–39.

3. Lüders, F., I. Crnkovic, and P. Runeson (2005). "Adopting a Component-Based Software Architecture for an Industrial Control System – A Case Study". In: *Component-Based Software Development for Embedded Systems*. Ed. by C. Atkinson, C. Bunse, H.-G. Gross, and C. Peper. Vol. 3778. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, pp.232–248. ISBN: 978-3-540-30644-3.

4. Runeson, P. (2007). "Sampling in Software Development". In: *Encyclopedia of Statistics in Quality and Reliability*. Ed. by F. Ruggeri, R. Kenett, and F. Faltinf. John Wiley and Sons Ltd, pp.1744–1747.

5. Weyns, K., M. Höst, Y. L. Helgesson, and P. Runeson (2010). "Pålitliga IT-system i krishantering". In: *FRIVA – risk, sårbarhet och förmåga. Samverkan inom krishantering*. Ed. by M.Höst et al. Lunds universitet.

6. Runeson, P. and E. Engström (2012). "Regression Testing in Software Product Line Engineering". In: *Advances in Computers*. Ed. by A. Menon. Vol. 86. Elsevier, pp.223–263.

7. Borg, M. and P. Runeson (2014). "Changes, Evolution and Bugs – Recommendation Systems for Issue Management". In: *Recommendation Systems in Software Engineering*. Ed. by M. P. Robillard, W. Maalej, R. J. Walker, and T. Zimmermann. Springer.

8. Olsson, T., C. Rosengren, P. Runeson, S. Bill, and S. Larsson (2014). "Tillit i det digitala samhället – en kartläggning". In: *DigiTrust: Tillit i det digitala – Tvärvetenskapliga perspektiv från ett forskningsprojekt*. Ed. by S. Larsson and P. Runeson. Pufendorfinstitutet vid Lunds universitet, pp.21–40. ISBN: 978-91-979893-6-7. `http://lup.lub.lu.se/record/4464066`.

Papers in conference proceedings

1. Runeson, P. and C. Wohlin (1992). Usage Modelling: The Basis for Statistical Quality Contro. In: *Proceedings 10th Annual Software Reliability Symposium*. Denver, Colorado, pp.77–84.

2. Wohlin, C. and P. Runeson (1992). A Method Proposal for Early Software Reliability Estimation. In: *Proceedings 3rd International Symposium on Software Reliability Engineering (ISSRE)*. North Carolina, USA, pp.156–163.

3. Cosmo, H., E. Johansson, P. Runeson, A. Sixtensson, and C. Wohlin (1993). Cleanroom Software Engineering in Telecommunication Applications. In: *Proceedings Software Engineering and its Applications in Telecommunications*. Paris, France, pp.369–378.

4. Runeson, P., A. Wesslén, J. Brantestam, and S. Sjöstedt (1995). Statistical Usage Testing using SDL. In: *SDL'95 with MSC in CASE*. Ed. by R. Braek and A Sarma. Elsevier Science, pp.323–336.

5. Regnell, B. and P. Runeson (1998). Combining Scenario-based Requirements with Static Verification and Dynamic Testing. In: *Proceedings 4th International Working Conference Requirements Engineering: Foundation for Software Quality*, pp.195–206.

6. Runeson, P. and P. Isacsson (1998). Software Quality Assurance - Concepts and Misconceptions. In: *Proceeding 24th EUROMICRO Conference - Engineering Systems and Software for the Next Decade*, pp.20853–20859.

7. Runeson, P. and B. Regnell (1998). Derivation of an Integrated Operational Profile and Use Case Model. In: *Proceedings 9th International Symposium on Software Reliability Engineering (ISSRE)*. Paderborn, Germany, pp.70–79.

8. Wohlin, C. and P. Runeson (1998). Defect Content Estimations from Review Data. In: *Proceedings International Conference on Software Engineering*, pp.400–409.

9. Bratthall, L. and P. Runeson (1999). Architecture Design Recovery of a Family of Embedded Software Systems. In: *Proceedings Working IEEE/IFIP Conference on Software Architecture*. San Antonio, Texas, pp.3–14.

10. Broomé, M. and P. Runeson (1999). Technical Requirements for the implementation of an Experience Base. In: *Proceedings 11th International Conference on Software Engineering and Knowledge Engineering (SEKE)*. Kaiserslautern, Germany, pp.1–9.

11. Runeson, P., B. Regnell, and T. Thelin (1999). Experimental and Simulation Analysis of Perspective Difference in Scenario-based Requirements Inspection. In: *Proceedings 5th International Workshop on Requirements Engineering: Foundations of Software Quality (REFSQ)*. Heidelberg, Germany, pp.71–82.

12. Thelin, T. and P. Runeson (1999). Capture-Recapture Estimations for Perspective-Based Reading - A Simulated Experiment. In: *Proceedings 1st International Conference on Product Focused Software Process Improvement (PROFES)*. Oulu, Finland, pp.182–200.

13. Thelin, T. and P. Runeson (1999). Robust Estimators of Fault Content with Capture-Recapture and Detection Profile Estimators. In: *Proceedings 3rd International Conference on Empirical Assessment in Software Engineering (EASE)*. Staffordshire, UK: Keele University.

14. Berling, T. and P. Runeson (2000). Application of Factorial Design to Validation of System Performance. In: *Proceedings 7th IEEE International Conference and Workshop on the Engineering of Computer Based Systems*. IEEE, pp.318–326.

15. Runeson, P. (2000). A New Software Engineering Program - Structure and Initial Experiences. In: *Proceedings 13th Conference on Software Engineering Education and Training (CSEE&T)*. Austin, Texas: IEEE Computer Society, pp.223–232.

16. Runeson, P., N. Borgquist, M. Landin, and W. Bolanowski (2000). An Evaluation of Functional Size Methods and a Bespoke Estimation Method for Real-Time Systems. In: *Proceedings 2nd International Conference on Product-Focused Software Process Improvement (PROFES)*, pp.339–352.

17. Thelin, T. and P. Runeson (2000). Fault Content Estimations using Extended Curve Fitting Models and Model Selection. In: *Proceedings 4th International Conference on Empirical Assessment & Evaluation in Software Engineering (EASE)*. Staffordshire, UK: Keele University.

18. Wesslén, A., P. Runeson, and B. Regnell (2000). Assessing the Sensitivity to Usage Profile Changes in Test Planning. In: *Proceedings 11th International Symposium on Software Reliability Engineering*. IEEE, pp.317–326.

19. Berling, T. and P. Runeson (2001). Evaluation of a Perspective Based Review Method Performed in an Industrial Setting. In: *Proceedings 5th International Conference on Empirical Assessment & Evaluation in Software Engineering (EASE)*. Keele University.

20. Olsson, T. and P. Runeson (2001). System Level Mutation Analysis Applied to a State-Based Language. In: *Proceedings 8th Annual IEEE International Conference and Workshop on the Engineering of Computer Based Systems (ECBS)*. Washington DC, USA, pp.222–228.

21. Olsson, T. and P. Runeson (2001). V-GQM: a feed-back approach to validation of a GQM study. In: *Proceedings Seventh International Software Metrics Symposium*. Institute of Electrical And Electronics Engineers, pp.236–245. http://dx.doi.org/10.1109/METRIC.2001.915532.

22. Olsson, T., P. Runeson, N. Bauer, and L. Bratthall (2001). An Experiment on Lead-Time Impact in Testing of Distributed Real-Time Systems. In: *Proceedings Seventh International Software Metrics Symposium*. Institute of Electrical And Electronics Engineers, pp.159–168. http://dx.doi.org/10.1109/METRIC.2001.915532.

23. Runeson, P. (2001). Experience from Teaching PSP for Freshmen. In: *Proceedings 14th Conference on Software Engineering Education & Training (CSEE&T)*. Charlotte, NC, USA, pp.98–107.

24. Runeson, P., M. C. Ohlsson, and C. Wohlin (2001). A Classification Scheme for Studies on Fault-Prone Components. In: *Proceedings 23rd International Conference on Product-Focused Software Process Improvement (PROFES)*. Vol. 2188. Lecture Notes in Computer Science. Springer, pp.341–355. http://link.springer.de/link/service/series/0558/bibs/2188/21880341.htm.

25. Wohlin, C., H. Petersson, M. Höst, and P. Runeson (2001). Defect Content Estimation for Two Reviewers. In: *Proceedings 12th International Symposium on Software Reliability Engineering (ISSRE)*. Hong Kong, pp.340–345.

26. Karlström, D. and P. Runeson (2002). Decision Support for Extreme Programming Introduction and Practice Selection. In: *Proceedings Fourteenth International Conference on Software Engineering and Knowledge Engineering*, pp.835–841. http://dx.doi.org/10.1145/568760.568902.

27. Karlström, D., P. Runeson, and C. Wohlin (2002). Aggregating Viewpoints for Strategic Software Process Improvement - a Method and a Case Study. In: *Proceedings 6th International Conference on Empirical Assessment & Evaluation in Software Engineering*. Keele University, Staffordshire, UK.

28. Ohlsson, M. C. and P. Runeson (2002). Experience from replicating empirical studies on prediction models. In: *Proceedings Eighth IEEE Symposium on Software Metrics*. IEEE Comput. Soc, pp.217–226. http://dx.doi.org/10.1109/METRIC.2002.1011340.

29. Thelin, T., P. Runeson, C. Wohlin, T. Olsson, and C. Andersson (2002). How much Information is Needed for Usage-Based Reading? — A Series of Experiments. In: *Proceedings International Symposium on Empirical Software Engineering (ISESE)*. Nara, Japan, pp.127–138.

30. Andersson, C., T. Thelin, P. Runeson, and N Dzamashvili (2003). An experimental evaluation of inspection and testing for detection of design faults. In: *Proceedings International Symposium on Empirical Software Engineering. ISESE 2003*. IEEE Comput. Soc, pp.174–184. http://dx.doi.org/10.1109/ISESE.2003.1237976.

31. Karlström, D. and P. Runeson (2003). Scaling extreme programming in a market driven development context. In: *Extreme Programming and Agile Processes in Software Engineering. 4th International Conference, XP 2003. Proceedings (Lecture Notes in Computer Science Vol.2675)*. Springer-Verlag, pp.363–365. http://www.springerlink.com/content/109ca55222j31ab8.

32. Runeson, P. (2003). Using Students as Experiment Subjects – An Analysis on Graduate and Freshmen Student Data. In: *Proceedings 7th International Conference on Empirical Assessment & Evaluation in Software Engineering*, pp.95–102.

33. Runeson, P. and A. Andrews (2003). Detection or isolation of defects? An experimental comparison of unit testing and code inspection. In: *Proceedings 14th International Symposium on Software Reliability Engineering*. IEEE Comput. Soc, pp.3–13. http://dx.doi.org/10.1109/ISSRE.2003.1251026.

34. Runeson, P. and T. Thelin (2003). A case study using sampling to improve software inspection effectiveness. In: *Proceedings International Symposium on Empirical Software Engineering. ISESE 2003*. IEEE Comput. Soc, pp.252–260. http://dx.doi.org/10.1109/ISESE.2003.1237985.

35. Andrews, A., P. Runeson, and R. France (2004). Requirements Trade-offs During UML Design. In: *Proceedings 11th Annual IEEE International Conference and Workshop on the Engineering of Computer Based Systems*. IEEE, pp.282–291.

36. Runeson, P., M. Holmstedt-Jönsson, and F. Scheja (2004). Are found defects an indicator of software correctness? An investigation in a controlled case study. In: *Proceedings International Symposium on Software Reliability Engineering, ISSRE*. Los Alamitos, CA, USA: IEEE Computer Society, pp.91–100. http://portal.acm.org/citation.cfm?id=1033820.

37. Skoglund, M. and P. Runeson (2004). A Case Study on Regression Test Suite Maintenance in System Evolution. In: *Proceedings 20th IEEE International Conference on Software Maintenance*. IEEE, pp.438–442.

38. Thelin, T., C. Andersson, P. Runeson, and N. Dzamashvili-Fogelstrom (2004). A replicated experiment of usage-based and checklist-based reading. In: *Proceedings International Software Metrics Symposium*. Los Alamitos, CA, USA: IEEE Computer Society, pp.246–256.

39. Runeson, P. and P. Greberg (2005). Extreme Programming and Rational Unified Process – Contrasts or Synonyms? In: *Proceedings European Software Process Improvement Conference (EuroSPI)*. Budapest, Hungary.

40. Skoglund, M. and P. Runeson (2005). A case study of the class firewall regression test selection technique on a large scale distributed software system. In: *Proceedings International Symposium on Empirical Software Engineering*. IEEE, pp.72–81. http://dx.doi.org/10.1109/ISESE.2005.1541816.

41. Runeson, P. and M. Wiberg (2006). Simulation of Experiments for Data Collection – a replicated study. In: *Proceedings 10th International Conference on Empirical Assessment & Evaluation in Software Engineering, (EASE)*. Staffordshire, UK: Keele University.

42. Weyns, K. and P. Runeson (2006). Software Dependability under Emergency Conditions. In: *Government track of 17th IEEE International Symposium on Software Reliability Engineering*. Raleigh, North Carolina.

43. Andersson, C. and P. Runeson (2007). Investigating Test Teams' Defect Detection in Function Test. In: *Proceedings First International Symposium on Empirical Software Engineering and Measurement (ESEM)*. Madrid, Spain: IEEE, pp.458–460. http://dx.doi.org/10.1109/ESEM.2007.60.

44. Höst, M. and P. Runeson (2007). Checklists for Software Engineering Case Study Research. In: *Proceedings First International Symposium on Empirical Software Engineering and Measurement (ESEM)*. Madrid, Spain: IEEE, pp.479–481. http://dx.doi.org/10.1109/ESEM.2007.29.

45. Runeson, P., M. Alexandersson, and O. Nyholm (2007). Detection of duplicate defect reports using natural language processing. In: *Proceedings International Conference on Software Engineering*. Piscataway, NJ, USA: Institute of Electrical and Electronics Engineers Computer Society, pp.499–508. http://dx.doi.org/10.1109/ICSE.2007.32.

46. Engström, E., M. Skoglund, and P. Runeson (2008). Empirical evaluations of regression test selection techniques: a systematic review. In: *Proceedings Second ACM-IEEE international symposium on Empirical Software Engineering and Measurements*. ACM, pp.22–31. http://dx.doi.org/10.1145/1414004.1414011.

47. Runeson, P. and M. Skoglund (2009). Reference-based search strategies in systematic reviews. In: *Proceedings 13th International Conference on Empirical Assessment & Evaluation in Software Engineering (EASE)*. Durham University, UK: British Computer Society.

48. Engström, E. and P. Runeson (2010). A Qualitative Survey of Regression Testing Practices. In: *Proceedings 11th International Conference on Product-Focused Software Process Improvement (PROFES)*. Ed. by M. Ali Babar, M. Vierimaa, and M. Oivo. Vol. 6156. Lecture Notes in Computer Science. (Best paper award). Springer Berlin / Heidelberg, pp.3–16.

49. Engström, E., P. Runeson, and G. Wikstrand (2010). An Empirical Evaluation of Regression Testing Based on Fix-cache Recommendations. In: *Proceedings 3rd International Conference on Software Testing Verification and Validation*. IEEE Computer Society, pp.75–78.

50. Kitchenham, B., D. I. K. Sjøberg, P. Brereton, D. Budgen, T. Dybå, M. Höst, D. Pfahl, and P. Runeson (2010). Can we evaluate the quality of software engineering experiments? In: *Proceedings International Symposium on Empirical Software Engineering and Measurement (ESEM)*. Ed. by G. Succi, M. Morisio, and N. Nagappan. Bolzano/Bozen, Italy: ACM.

51. Sabaliauskaite, G., A. Loconsole, E. Engström, M. Unterkalmsteiner, B. Regnell, P. Runeson, T. Gorschek, and R. Feldt (2010). Challenges in Aligning Requirements Engineering and Verification in a Large-Scale Industrial Context. In: *Proceedings 16th International Working Conference Requirements Engineering: Foundation for Software Quality*. Ed. by R. Wieringa and A. Persson. Vol. 6182. Lecture Notes in Computer Science. Essen, Germany: Springer, pp.128–142.

52. Cruzes, D. S., T. Dybå, P. Runeson, and M. Höst (2011). Case Studies Synthesis: Brief Experience and Challenges for the Future. In: *Proceedings International Symposium on Empirical Software Engineering and Measurement (ESEM)*. Banff, Canada.

53. Engström, E., P. Runeson, and A. Ljung (2011). Improving Regression Testing Transparency and Efficiency with History Based Prioritization – an Industrial Case Study. In: *Proceedings 4th International Conference on Software Testing Verification and Validation*. IEEE Computer Society, pp.367–376. DOI: 10.1109/ICST.2011.27.

54. Höst, M., A. Orucevic-Alagic, and P. Runeson (2011). Usage of Open Source in Commercial Software Product Development - Findings from a Focus Group Meeting. In: *Proceedings 12th International Conference on Product-Focused Software Process Improvement (PROFES)*. Ed. by D. Caivano, M. Oivo, M. T. Baldassarre, and G. Visaggio. Vol. 6759. Lecture Notes in Business Information Processing. Springer, pp.143–155.

55. Kasurinen, J., P. Runeson, L. Riungu, and K. Smolander (2011). A Self-Assessment Framework for Finding Improvement Objectives with ISO/IEC 29119 Test Standard. In: *Proceedings European Software Process Improvement Conference (EuroSPI)*. Roskilde, Denmark.

56. Runeson, P., P. Heed, and A. Westrup (2011). A Factorial Experimental Evaluation of Automated Test Input Generation – Java Platform Testing in Embedded Devices. In: *Proceedings 12th International Conference on Product-Focused Software Process Improvement (PROFES)*. Ed. by D. Caivano, M. Oivo, M. T. Baldassarre, and G. Visaggio. Vol. 6759. Lecture Notes in Business Information Processing. Springer, pp.217–231.

57. Borg, M., P. Runeson, and L. Brodén (2012). Evaluation of Traceability Recovery in Context: A Taxonomy for Information Retrieval Tools. In: *16th International Conference on Evaluation & Assessment in Software Engineering*. Ciudad Real, Spain.

58. Borg, M., D. Pfahl, and P. Runeson (2013). Analysing Networks of Issue Reports. In: *Proceedings 17th European Conference on Software Maintenance and Reengineering*. Genova, Italy, pp.79–88.

59. Borg, M. and P. Runeson (2013). IR in Software Traceability: From a Bird's Eye View. In: *In Proceedings Empirical Software Engineering and Measurements*, pp.243–246.

60. Pedersen Notander, J., M. Höst, and P. Runeson (2013). Challenges in Flexible Safety-Critical Software Development – An Industrial Qualitative Survey. In: *Proceedings 14th International Conference on Product-Focused Software Process Improvement (PROFES)*. Ed. by J. Heidrich, M. Oivo, A. Jedlitschka, and M. T. Baldassarre. Vol. 7983. Lecture Notes in Computer Science. Paphos, Cyprus: Springer, pp.283–297.

61. Pederson Notander, J., P. Runeson, and M. Höst (2013). A Model-Based Framework for Flexible Safety-Critical Software Development – A Design Study. In: *Proceedings 28th Symposium On Applied Computing*. Coimbra, Portugal, pp.1137–1144.

62. Wnuk, K. and P. Runeson (2013). "Engineering Open Innovation – a Framework for Fostering Open Innovation". In: *Software Business. From Physical Products to Software Services and Solutions (ICSOB13)*. Ed. by G. Herzwurm and T. Margaria. Vol. 150. Lecture Notes in Business Information Processing. Springer, pp.48–59. DOI: 10.1007/978-3-642-39336-5_6.

63. Borg, M., P. Runeson, J. Johansson, and M. Mäntylä (2014). A replicated study on duplicate detection: Using Apache Lucene to search among Android defects. In: *Proceedings Empirical Software Engineering and Measurements*. Torino, Italy. DOI: 10.1145/2652524.2652556.

64. de Mello, R. M., P. Silva, P. Runeson, and G. H. Travassos (2014). Towards a Framework to Support Large Scale Sampling in Software Engineering Surveys. In: *Proceedings Empirical Software Engineering and Measurements*. (Best short paper award (tied)). Torino, Italy. DOI: 10.1145/2652524.2652567.

65. Engström, E., M. Mäntylä, P. Runeson, and M. Borg (2014). Supporting Regression Test Scoping with Visual Analytics. In: *Proceedings of the 2014 IEEE International Conference on Software Testing*, *Verification*, *and Validation*, pp.283–292. DOI: 10.1109/ICST.2014.41.

66. Wnuk, K., K. Manikas, P. Runeson, M. Lantz, O. Weijden, and H. Munir (2014). Evaluating the Governance Model of Hardware-Dependent Software Ecosystems - A Case Study of the Axis Ecosystem. In: *International Conference on Software Business*.

67. Erman, N., V. Tufvesson, M. Borg, P. Runeson, and A. Ardö (2015). Navigating Information Overload Caused by Automated Testing - A Clustering Approach in Multi-Branch Development. In: *Proceedings 8th IEEE International Conference on Software Testing*, *Verification and Validation*.

68. Linåker, J., H. Munir, P. Runeson, B. Regnell, and C. Schrewelius (2015). A Survey on the Perception of Innovation in a Large Software Organization. In: *International Conference on Software Business*. Ed. by K. Wnuk and R. J. Machado.

Papers in workshop proceedings

1. Cosmo, H., G. Fagerhus, E. Johansson, E.-A. Karlsson, P. Runeson, C. Sandahl, A. Sixtensson, and C. Wohlin (1992). Cleanroom Software Engineering Applied to Telecommunications. In: *Proceedings Nordic Seminar on Dependable Computing Systems*, Trondheim, Norway.

2. Runeson, P. and C. Wohlin (1993). Statistical Usage Testing for Software Reliability Certification and Control. In: *Proceedings 1st European International Conference on Software Testing*, *Analysis and Review (EuroSTAR)*. London, UK, pp.309–323.

3. Wohlin, C., P. Runeson, and J.-E. Johansson (1996). When is the Software Reliability Estimate Reliable? In: *Workshop on Issues in Software Reliability*. Leidschendam, The Netherlands.

4. Wohlin, C., P. Runeson, and A. Wesslén (1996). Software Reliability Control using Statistical Usage Testing. In: *Encress seminar*. Norway.

5. Berling, T. and P. Runeson (1998). A Systematic Improvement Approach of System Verification and Validation. In: *Proceedings 6th European International Conference on Software Testing, Analysis and Review (EuroSTAR)*. Munich, Germany.

6. Bratthall, L. and P. Runeson (1998). Architectural Design Recovery of a Family of Embedded Software Systems – An Experience Report. In: *Proceedings Nordic Workshop on Software Architectures (NOSA)*. Ronneby, Sweden.

7. Bratthall, L. and P. Runeson (1999). A Taxonomy of Orthogonal Properties of Software Architectures. In: *Proceedings Nordic Workshop on Software Architectures (NOSA)*. Ronneby, Sweden.

8. Runeson, P., C. Wohlin, and M. Ohlsson (1999). A Proposal for Comparison of Models for Identification of Fault-Proneness. In: *Proceedings 5th Workshop on Empirical Studies of Software Maintenance (WESS)*. Oxford, UK.

9. Bylund, J., N. Lundin, P. Waldemarsson, T. Eriksson, and P. Runeson (2000). Key Success Factors in Software Development. In: *FESMA-AEMES Software Measurement Conference*. Madrid, Spain.

10. Olsson, T. and P. Runeson (2001). Baselining Software Processes as a Starting Point for Research and Improvement. In: *Swedish Conference on Software Engineering Research and Practice (SERPS)*. Ronneby, Sweden.

11. Olsson, T. and P. Runeson (2001). Information Refinement Using Hierarchical Structures. In: *ICSE'01 - Workshop on Living with Inconsistency at the International Conference on Software Engineering*.

12. Petersson, H., T. Thelin, P. Runeson, and C. Wohlin (2001). Capture-recapture in software inspections after 10 years research - theory, evaluation and application. In: *Proceedings Swedish Conference on Software Engineering Research and Practice (SERPS)*. Ronneby, Sweden, pp.37–48.

13. Thelin, T., P. Runeson, and C. Wohlin (2001). An Experimental Comparison of Usage-Based and Checklist-Based Reading. In: *Proceeding 1st Workshop on Inspections in Software Engineering*. Paris, France, pp.136–144.

14. Berling, T. and P. Runeson (2002). Efficient Evaluation of Multi-Factor Dependent System Performance using Fractional Factorial Design. In: *Proceedings Conference on Software Engineering Research and Practice in Sweden*. Karlskrona, Sweden.

15. Olsson, T. and P. Runeson (2002). Document Use in Software Development: A Qualitative Survey. In: *Proceedings Conference on Software Engineering Research and Practice in Sweden*. Karlskrona, Sweden.

16. Runeson, P. and T. Thelin (2003). Addressing Attitudes Explicitly in Engineering Education- An Exercise to Stimulate Reflection through Pictures. In: *LTH pedagogisk inspirationskonferens*, pp.40–41.

17. Runeson, P. and T. Thelin (2003). Prospects and Limitations for Cross-Study Analyses – A Study on an Experiment Series. In: *Proceedings 2nd Workshop in Workshop Series on Empirical Software Engineering*, pp.133–142.

18. Andersson, C., P. Runeson, and T. Thelin (2004). Evaluating the impact of software process simulations - a case study. In: *Proceedings 5th International Workshop on Software Process Simulation and Modeling (ProSim 2004)*. 26th International Conference on Software Engineering. IEEE, pp.217–221.

19. Regnell, B. and P. Runeson (2004). Model-Based Course Assessment – Principles and Practice. In: *LTH pedagogisk inspirationskonferens*.

20. Runeson, P. and P. Greberg (2004). Extreme Programming and Rational Unified Process — Contrasts or Synonyms? In: *Proceedings Swedish Conference on Software Engineering Research and Practice (SERPS)*. Linköping, Sweden.

21. Andersson, C. and P. Runeson (2005). A Case Study on Quality Monitoring in a Highly Iterative Software Development Process. In: *Proceedings Fifth Conference on Software Engineering Research and Practice in Sweden*.

22. Runeson, P., T. Thelin, and C. Andersson (2005). Advanced industry-academy learning on software testing. In: *EuroSTAR*. Copenhagen, Denmark.

23. Runeson, P. and M. Wiberg (2005). Simulation of Experiments for Data Collection – a replicated study. In: *Proceedings 5th Conference on Software Engineering Research and Practice in Sweden, 2005*.

24. Weyns, K. and P. Runeson (2005). Sensitivity of System Reliability to Usage Profile Changes. In: *Proceedings 5th Conference on Software Engineering Research and Practice in Sweden*.

25. Runeson, P. (2006). A Survey of Unit Testing Practices. In: *EuroSTAR*. Manchester, UK.

26. Runeson, P., P. Beremark, B. Larsson, and E. Lundh (2006). SPIN-syd – a non-profit exchange network. In: *Proceedings 1st International Workshop on Software Engineering Networking Experiences*. Joensuu, Finland.

27. Regnell, B., M. Höst, and P. Runeson (2008). En beprövad process som stöd till examensarbetare och deras handledare. In: *LTH pedagogisk inspirationskonferens*.

28. Runeson, P., M. Skoglund, and E. Engström (2008). Test Benchmarks – what is the question? In: *Proceedings First Software Testing Benchmark Workshop*. Ed. by M. Roper.

29. Sabaliauskaite, G. and P. Runeson (2009). Early Indicators of Software Quality – how far can we go? In: *Proceedings Third International Workshop on Software Quality and Maintainability*. Kaiserslautern, Germany.

30. Engström, E. and P. Runeson (2011). Decision Support for Test Management and Scope Selection in a Software Product Line Context. In: *Proceedings 1st Int'l Workshop on Variability-intensive Systems Testing, Validation & Verification*. Berlin, Germany.

31. Runeson, P., A. Stefik, A. Andrews, S. Grönblom, I. Porres, and S. Siebert (2011). A Comparative Analysis of Three Replicated Experiments Comparing Inspection and Unit Testing. In: *Proceedings 2nd International Workshop on Replication in Empirical Software Engineering*. Banff, Canada, pp.35–42. DOI: 10.1109/RESER.2011.13.

32. Runeson, P. (2012). It Takes Two to Tango – An Experience Report on Industry–Academia Collaboration. In: *Testing: Academic and Industrial Conference - Practice and Research Techniques (TAIC-PART)*. (Best paper award), pp.872–877. DOI: 10.1109/ICST.2012.190.

33. Runeson, P. (2012). Theory Building Attempts in Software Engineering. In: *The Semat Workshop on a General Theory of Software Engineering*. Ed. by I. Jacobson, M. Goedicke, and P. Johnson. Stockholm, Sweden.

34. Runeson, P. and E. Engström (2012). Software Product Line Testing - A 3D Regression Testing Problem. In: *Proceedings of 2nd International Workshop on Regression Testing*, pp.742–746. DOI: 10.1109/ICST.2012.167.

35. Bjarnason, E., K. Smolander, E. Engström, and P. Runeson (2014). Alignment Practices Affect Distances in Software Development: A Theory and a Model. In: *3rd SEMAT Workshop on General Theories of Software Engineering (ICSE 2014 workshop)*, pp.21–31. DOI: 10.1145/2593752.2593757.

36. Runeson, P. and S. Minör (2014). The 4+1 View Model of Industry–Academia Collaboration. In: *International Workshop on Long-term Industrial Collaboration on Software Engineering (WISE)*. ACM. DOI: 10.1145/2647648.2647651.

37. Runeson, P., S. Minör, and J. Svenér (2014). Get the Cogs in Synch – Time Horizon Aspects of Industry–Academia Collaboration. In: *International Workshop on Long-term Industrial Collaboration on Software Engineering (WISE)*. ACM. DOI: 10.1145/2647648.2647652.

38. Munir, H. and P. Runeson (2015). Software Testing in Open Innovation: An Exploratory Case Study of the Acceptance Test Harness for Jenkins. In: *First International Workshop of Open Innovation in Software Engineering*. Ed. by M. Nayebi and K. Wnuk. International Conference Proceedings Series. ACM.

39. Galinac Grbac, T. and P. Runeson (2016). Plug-in Software Engineering Case Studies. In: *Proceedings 4th International Workshop on Conducting Empirical Studies in Industry (CESI@ICSE16)*. Austin, Texas: ACM, pp.21–24. DOI: 10.1145/2896839.2896840.

Editorials and tutorials

1. Messnarz, R., I. Richardson, and P. Runeson (2006). Software process improvement - EuroSPI 2006 Conference. In: *Software Process Improvement, Proceedings (Lecture Notes in Computer Science)*. Vol. 4257. Springer Berlin / Heidelberg, pp.1–4. http://dx.doi.org/10.1007/11908562_1.

2. Wohlin, C., P. Runeson, and M. Höst (2007). Case Study Research in Software Engineering. In: *International Advanced School of Empirical Software Engineering*. Madrid, Spain.

3. Runeson, P. and M. Höst (2009). Tutorial: Case Studies in Software Engineering. In: *Proceedings 10th International Conference on Product-Focused Software Process Improvement (PROFES)*. Ed. by F. Bomarius, M. Oivo, P. Jaring, and P. Abrahamsson. Vol. 32. Lecture Notes in Computer Science. Springer, pp.441–442.

4. Offutt, J. and P. Runeson (2011). ICST 2009 Special Issue. *Software Testing, Verification and Reliability* **21**(3), 153–154.

5. Runeson, P., M. Höst, E. Mendes, A. A. Andrews, and R. Harrison, eds. (2012). *ESEM 12: Proceedings of the ACM-IEEE International Symposium on Empirical Software Engineering and Measurement*. Lund, Sweden: ACM. ISBN: 978-1-4503-1056-7.

6. Yoo, S. and P. Runeson (2014). Guest editorial: special section on regression testing. *Software Quality Journal* **22**(4), 699.

7. Bagheri, E., D. Benavides, K. Schmid, and P. Runeson (2016). Foreword to the special issue on empirical evidence on software product line engineering. *Empirical Software Engineering*, 1–7.

Other publications

1. Runeson, P. and B. Regnell (1999). IT-företagens kaos måste lösas. In: *Sydsvenskan*.

2. Runeson, P. (2004). Hur många fel finns det kvar i programmet. In: *Quartilen*.

3. Runeson, P. (2004). Jobba hårt eller jobba smart. In: *LTH-nytt*. Vol. 4.

4. Runeson (ed.), P. (2005). Vem vill bli rektor för ett universitet i Sverige? In: *Göteborgsposten Debatt*, *Upsala Nya Tidning*.

5. Andersson, T., C. Brogren, I. Crnkovic, H. Eriksson, M. Lindoff, P. Runeson, K. Sandahl, and C. Wohlin (2006). Programvara – Stor i exportindustrin, men liten i forskningen. In: *Ny Teknik*.

6. Runeson, P. and J. Falk (2007). Därför kraschar datorn. *Forskning och framsteg* **5**, 48–51.

7. Runeson, P. (2009). Det allmänna prästadömet – hos Luther, Rosenius och i vår tid. *Begrunda* (1), 11–18.

8. Runeson, P. (2009). Skapa ingen konflikt mellan nyfikenhet och nytta. In: *LTH-nytt*. Vol. 2.

9. Regnell, B., M. Stanford, P. Runeson, D. Sjöberg, K. Scott, E. Nordberg-Karlsson, and M. Gustafsson (2010). "Den osynliga forskargruppen — ledarskap i och kring akademins verksamhetsbärande enheter". In: *AKKA III*. ISBN 978-91-633-6386-3 99-131. Lunds universitet.

10. Runeson, P. (2010). Kyrkan – organism och organisation. *Begrunda* (1), 11–18.

11. Runeson, P. (2013). *Psaltarklanger*. Helsingborg, Sverige: GAudete förlag. ISBN: 978-91-86577-32-2.

12. Runeson, P. and R. B. Svensson (2013). Etisk mjukvara? Ingenjörsprocessen för programvaruutveckling – samhällsaspekter. In: *Lärande i LTH*. Vol. 20. `http://www.lth.se/fileadmin/lth/genombrottet/Laerande_i_LTH/Laerande_i_LTH_blad_20.pdf`.

13. Larsson, S., T. Olsson, C. Rosengren, and P. Runeson (2014). Ta debatt om dataövervakning. In: *Svenska Dagbladet*.

14. Runeson, P. (2014). Utvandraren. *Klassiker online*.

15. Regnell, B., P. Runeson, M. Höst, and J. Linåker (2015). Innovation med öppen källkod ger konkurrensfördelar. In: *Management of Innovation and Technology*. Vol. 2. `http://imit.se`.

16. Runeson, P. (2015). Openness in the Interplay between Technical and Business aspects – a system of systems. In: *1st Scandinavian Workshop on the Engineering of Systems of Systems*. Ed. by J. Axelsson. SICS Technical Report T2015:04. Swedish Institute of Computer Science. `http://soda.swedishict.se/5836/1/SWESoS_2015_proceedings.pdf`.

17. Runeson, P., S. Andersson, F. Heintz, L. Mannila, and L. Rolandsson (2015). *Hur skall det svenska utbildningssystemet möta framtidens utmaningar? Helhetssyn på svensk utbildning om digitalisering och programmering*. White paper. Swedsoft. `http://swedsoft.se/wp-content/uploads/sites/7/2015/05/2015-01-Utbildning-white-paper.pdf`.

18. Runeson, P., H. Munir, and K. Wnuk (2016). It is More Blessed to Give than to Receive – Open Software Tools Enable Open Innovation. *Tiny Transactions on Computer Science (TinyToCS)* **4**.