Report from the 2nd International Workshop on Replication in Empirical Software Engineering Research (RESER 2011)

Jonathan L. Krein[1], Charles D. Knutson[1], Lutz Prechelt[2], Natalia Juristo[3]
[1]Brigham Young University, USA
[2]Freie Universität Berlin, Germany
[3]Universidad Politécnica de Madrid, Spain
jonathankrein@byu.net, knutson@cs.byu.edu, prechelt@inf.fu-berlin.de, natalia@fi.upm.es

http://sequoia.cs.byu.edu/reser2011

DOI: 10.1145/2088883.2088889 http://doi.acm.org/10.1145/2088883.2088889

ABSTRACT

The RESER workshop provides a venue in which empirical software engineering researchers can discuss the theoretical foundations and methods of replication, as well as present the results of specific replicated studies. In 2011, the workshop co-located with the International Symposium on Empirical Software Engineering and Measurement (ESEM) in Banff, Alberta, Canada. In addition to several outstanding paper sessions, highlights of the 2011 workshop included a keynote address by Dr. Victor R. Basili, in which he addressed the question, "What's so hard about replication of software engineering experiments?" The workshop also featured a joint replication panel session discussing the first cooperative joint replication ever conducted in empirical software engineering research and a planning session for next year's joint replication project addressing Conway's Law.

Categories and Subject Descriptors

D.2.8 [**Software Engineering**]: Metrics—*Performance measures, Process metrics, Product metrics*; G.3 [**Probability and Statistics**]: *Experimental design*; K.2 [**History of Computing**]: *Software, Theory*; K.6.3 [**Management of Computing and Information Systems**]: Software Management—*Software process*

General Terms

Experimentation, Measurement, Theory

Keywords

Experimentation, Methods, Replication, Reporting, Validity, Validation, Software Engineering

1. INTRODUCTION

Many results in Software Engineering suffer from threats to validity that can be addressed by the replication of previous empirical studies. These threats include: 1) Lack of independent validation of empirical results; 2) Contextual shifts in Software Engineering practices or environments since the time of the original research studies; and 3) Limited data sets at the time of the original research studies [2, 3, 19].

However, certain factors discourage replication studies: 1) A perception persists that replication studies are less valuable than the presentation of original studies; 2) Data sets are often not made publicly available; 3) Reports of empirical studies are often not sufficiently detailed to foster replication [5, 10]; and 4) Research tools are either not available or not usable, so precise replication is impractical [2, 3, 9, 19, 20].

Thus the primary goal of the RESER workshop is to raise the quality and amount of replication work performed in software engineering research. In particular this means:

- Collecting replications, whether confirming or contradictory, of previous studies, whether recent or old, on important questions.
- Collecting and packaging advice, tools, and experience regarding replication.
- Forming full-scale replications, perhaps going much beyond the original study, from multiple small-scale replications performed in a coordinated manner.

The workshop is also a forum for small-scale, "useless" replications that are otherwise hard to publish [7]. As part of the workshop, each year we collect results for one specific joint replication—soliciting small-scale replications, from which we form large-scale studies by meta-analysis. Through this process, the workshop is able to produce valuable insights for both specific research topics and regarding practical issues of replication. In addition, the workshop seeks to identify solutions for recurring practical problems in selecting, designing, performing, reporting, and publishing replication studies by furthering appropriate methods, tools, and standards.

2. KEYNOTE ADDRESS

This year's keynote address—*What's so Hard about Replication of Software Engineering Experiments?*—was given by Dr. Victor Basili[1]. During his talk, Vic addressed several important questions regarding replication: 1) Why do we replicate experiments?—to verify the results from the first experiment, to expand our knowledge of the discipline, and to build models that can be used to predict and to be challenged. 2) In that case, what does it mean to replicate an experiment, and 3) what are the criteria for a replication? Physicists and sociologists experiment and replicate experiments, explained Vic, but their expectations are different with respect to verifiability, expansion of knowledge, and

[1]Victor Basili is Professor Emeritus of Computer Science at the University of Maryland and holds a Ph.D. in Computer Science from the University of Texas. He was Director of the Maryland Fraunhofer Center and a director of the Software Engineering Laboratory at NASA/GSFC. He has worked on measuring, evaluating, and improving the software development process and product for over 35 years with numerous companies and government agencies. Methods include Iterative Enhancement, the Goal Question Metric Approach (GQM), the Quality Improvement Paradigm (QIP), and the Experience Factory (EF). He is Co-EIC of the Springer Empirical Software Engineering Journal and an IEEE and ACM Fellow. For more information, visit his website [1].

precision in prediction. The difference has to do with the nature of the domain they are studying. The domain affects their ability to generate relevant and testable hypotheses, identify, control, and manipulate the context variables, supply the appropriate documentation. Knowledge building through replication requires the support of a sufficiently large community of researchers who think empirically.

Vic then posed a reformulation of his initial question: What's so different/hard about *experimentation* in software engineering?—or in other words, is it possible that the difficult aspects of replication are simply inherent to experimentation in this field? From this point, Vic described research he had completed in collaboration with Filippo Lanubile and Forrest Shull. Their work was an effort to deal with the difficulty of building knowledge in software engineering. As part of that project, they proposed a framework for building a body of software engineering knowledge by identifying key dependent and independent variables and using those variables to integrate collections of experiments with like hypotheses. Thus the framework focused on using context variables to classify and strategize knowledge-production activities.

Interestingly, this point echoes one of the key take-aways from last year's keynote address[2], in which James Herbsleb stated that "replication is always about generalization," suggesting that different types of replication allow for different types of generalization [12]. It seems that Jim and Vic are both suggesting that we cannot effectively define the concept of replication (at least if we are to operationalize it) in isolation from the type of knowledge we intend to build. It follows then that an individual replication is fairly meaningless if not constructed within the broader context of a research strategy. Further, the notion of an *individual* replication is far less meaningful or analytically powerful—with respect to building knowledge—than that of replication as an extended process, a line of inquiry, a protracted conversation with the world.

Vic's keynote incited numerous audience interruptions, questions, comments, debate, etc. In short, it was a great success.

3. JOINT REPLICATION PANEL SESSION

The joint replication panel session featured work by four separate research teams[3] and explored the results and methodology of the first (to our knowledge) cooperative *joint replication*[4] ever conducted in empirical software engineering research. As a result of this project, the target study—at only six replications—is now (arguably [6]) one of the most prolifically replicated studies in the history of software engineering research.

[2]RESER 2010, Cape Town, South Africa.

[3]Universidad Politécnica de Madrid, Spain, Brigham Young University, USA, University of Alabama, USA, and Freie Universität Berlin, Germany [11, 14, 15, 16].

[4]The term *joint replication* refers to a replication "that is not performed by a single researcher or a single closely-knit research group, but rather by a group of researchers who work together loosely" [16]. In a joint replication, participating research groups initially "gather subjects and collect, clean, and analyze data independently," though ideally a combined meta-analysis would be performed following individual analyses [14]. In the case of a controlled experiment, if the research groups share a common definition of the experiment to be replicated, then the process may be termed a *strict* joint replication [14]—meaning that the participants attempt to match their replications to one another as closely as possible. The idea of a joint replication was originally conceived in 2010, following last year's workshop. The concept was inspired by the work of Dieste, Fernandez, Garcia, and Juristo on the potential value of small-scale, "useless" replications [7].

The original study was first performed in 1997 by Prechelt, Unger, Tichy, Brössler, and Votta, investigating the impact that design patterns have on software maintenance. This work, referred to as PatMain, was originally conducted in a paper-based format (no actual programming and testing) with 27 professionals, and published in TSE in 2001 [17]. PatMain was then replicated by researchers at Simula Research Laboratory with 44 paid professional subjects from various consultancy companies. That replication, published in EMSE in 2002 [21], was nearly identical in its setup, except that the subjects worked in a real programming environment. For the RESER 2011 joint replication, the participating research labs shared a common experimental framework which, though it necessitated some changes, mirrored the original experiment as much as possible [8, 11, 14, 15, 16].

Overall, the joint replication panel session was a truly workshop-style event. Lutz Prechelt opened the session by presenting an overview of the original experiment, followed by an explanation of the joint replication concept, as well as the RESER 2011 experiment framework. Each of the four participating research labs then took a few minutes to explain their specific experiments (subjects, expected/actual results, global findings, etc.)—at least that was the intended strategy. It quickly became clear, however, that the session needed a *lot* more time given the amount of resulting discussion. Consequently, the latter half of the session was forced to focus on the global findings[5] and analysis challenges of only one of the four labs.

By the end of the session, it became clear that the joint replication process is generating significant insights on multiple levels. Not only is the project attempting to address a question of practical interest regarding design patterns and software maintenance, but it also pilots a new research methodology, which is generating a whole set of new questions. The workshop discussion also suggested several key take-aways regarding general experimentation in software engineering, including insights on the use of students as subjects in software engineering experiments (made possible by the fact that the PatMain experiment replications now collectively span a diverse set of subjects). Regarding subjects, one workshop attendee commented on the fact that although the subjects look demographically similar across labs, their performance varies far more across replications than within. Further discussion led to an important idea that has inspired us to continue analysis after the workshop—a truly "workshopish" thing to happen.[6]

4. PAPER SESSION 1

The first technical paper session included three paper presentations. The first paper, "A Comparative Analysis of Three Replicated Experiments Comparing Inspection and Unit Testing," [18] was presented by Andreas Stefik. The study explores the relative effectiveness of code inspection versus unit testing utilizing three replications—two strict and one differentiated. The study's results indicate some differences between testing and inspection for some metrics, but more importantly, the synthesis of the three replications reveals that the results are "overshadowed by complex differences in the tasks and experiments." The study concludes that "both the differences in the instrumentation and the between-experiment participants themselves were larger than the differences between inspection versus unit testing." This observation demonstrates that while the task of synthesis is non-trivial, it is extremely important for actual knowledge building if we are to avoid prema-

[5]The results of a combined analysis of all four data sets—i.e., a joint or meta-analysis.

[6]Analysis for the joint replication is still in progress; results and methodology are targeted for publication 2012-13.

ture and *useless* generalization. These conclusions also mirror the synthesis results explored in the Joint Replication Panel Session (see Section 3). It appears that an insufficient understanding of context variables—inherently a search problem—affects more than one research area in software engineering.

The second paper, "A Secondary Data Archive for Code-Level Debian Metrics," [13] was presented by Megan Squire. This paper presents "a new process to collect, calculate, archive, and distribute interesting metrics for all the packages in the standard Debian GNU/Linux Installation." Of particular interest to the RESER workshop, this process is integrated into the automated FLOSSmole data store, facilitating "timely, repeatable, and very easy comparison, replication and analysis by other groups." Thus, Kozak and Squire's work provides significant support for the replication of artifact-based research. As stated in the paper, their goal is "to show researchers how useful it is to collect and update metrics and metadata frequently, then donate the entire corpus of data collected and analyzed to a secondary data archive, where it will be made freely available for anyone to download, test, use, and extend." By facilitating reliable and accessible data stores, these efforts help to lay a more solid foundation for replication and knowledge building in the domain of artifact-based research.

The final paper, "Design Team Perception of Development Team Composition: Implications for Conway's Law," [4] was presented by Charles Knutson. This paper examines "a pilot study intended to foster discussion within the RESER community." In the context of the workshop, the goal of the presentation was to lay groundwork for a later session (see Section 6) by proposing a joint replication of Conway's Law for RESER 2012. In the paper, the authors present a controlled experiment designed to elucidate cognitive nuances of Conway's Law. The study argues that, with respect to a software system, "the designers' perception of the ultimate composition of the development team [may] affect the resultant system architecture more so than [does] the actual composition of the design team." By reporting evidence that Conway's Law is more complex than current formulations express, the study calls for additional, more thorough replications to explore the nuances of Conway's *phenomenon*.

5. PAPER SESSION 2

The second technical session included two paper presentations. The first paper, "Replication of Empirical Studies in Software Engineering: Preliminary Findings From a Systematic Mapping Study," [6] was presented by Fabio da Silva. In this study, the authors attempt to shed light on the amount of empirical replication that has been conducted in software engineering research over the past 17 years. The effort they describe is herculean, involving an analysis of over 16,000 academic articles, of which only 93 report a replication of some type. Those articles document 125 replications of 76 original studies. The authors observe that although the number of replications has risen over the last few years, the absolute number of replications is still very small, especially considering the breadth of topics in software engineering. Their work suggests that as a community we need better incentives to perform external replications as well as improved standards for reporting empirical studies and replications. Incidentally, their comments on reporting standards echo a call made by Jeff Carver at last year's workshop to develop guidelines for reporting replicated experiments [5].

The second paper, "Replicate, Replicate, Replicate," [22] was presented by Elaine Weyuker. In this paper, authors observe that while empirical replication is standard procedure throughout all fields of scientific experimentation, in software engineering it "is often considered an inferior type of research." The authors also describe four types of replication that they have been performing as part of validating the effectiveness of their core research in software fault prediction. In particular, they discuss replication over time, replication by using different subject systems, replication by changing variables in prediction models, and replication by varying modeling algorithms. Additionally, Elaine pointed out that one important purpose for replication is to aggregate empirical evidence sufficient to convince practitioners to believe and adopt research findings—"it is difficult to encourage practitioners to adopt our research techniques since they have not seen sufficient empirical evidence, and it is likely that whatever evidence they have seen does not sound like their environment."

6. CONWAY SESSION

In the last session of the day, Charles Knutson led the discussion on a proposed joint replication for RESER 2012. Using an earlier paper on Conway's Law [4] (presented in Paper Session 2) as a launch point, workshop participants engaged in a spirited debate concerning the nature and value of joint replications (such as the PatMain study performed for RESER 2011). Also, considering that the 2011 joint replication was (for the most part) a strict replication, and since the pilot study of Conway's Law was clearly differentiated, workshop participants explored the relative merits of each of these two approaches. Judging from the energy level of that discussion, it seems reasonable to conclude (as we did last year [12]) that the definition of replication in our field is still somewhat controversial. Further, it is not clear yet whether these differences reflect positively on the state of our science or whether they are tell-tell signs of methodological immaturity. On the one hand, they may inspire diversity and creativity in the scientific process; on the other, they may degrade communication and collaboration between researchers. Following these discussions, the Conway Session closed with a discussion of the appropriate scope for a joint replication of Conway's Law.

The goal of the Conway Session was to get ideas "on the table" sufficient to launch an online discussion for next year's workshop. To participate in the 2012 joint replication of Conway's Law, visit the RESER website.[7] The latest information (including contact information) and a call for participation will be posted there. The invitation to participate is open to all, whether you have previously been involved with RESER or not.

7. CONCLUSIONS

As organizers, we feel that the second iteration of the RESER workshop has been a tremendous success. The papers and presentations were both relevant and high quality, the speakers were engaging and the atmosphere was true to the spirit of a workshop, with major insights emerging far more from the participants and the discussion than from any prepared material. The workshop dinner at the Maple Leaf Grill was wonderful, with great food and engaging company the night before the workshop. Significantly, the keynote by Vic Basili was the most "workshopish" keynote any of us have ever experienced, with the audience interrupting Vic to pepper him with engaging and thoughtful questions. The setting in Banff was beautiful and inspiring, and we are grateful to our hosts for their hospitality. We look forward to the third iteration of the RESER workshop in 2012.

8. ACKNOWLEDGMENTS

We gratefully acknowledge the feedback and support of the advisory committee members: Joerg Doerr of Fraunhofer IESE, Germany

[7] `http://sequoia.cs.byu.edu/reser`

and Peri Tarr of the IBM T. J. Watson Research Center, USA. Our Program Committee also provided outstanding reviews and feedback to the authors, for which service we are indebted to them. Finally, we acknowledge the generous support of our corporate sponsor, InfoTrax Systems, Inc., who helped fund (among other things) the organization of the workshop and the workshop dinner at the Maple Leaf Grill. We sincerely appreciate their continued interest in advancing Software Engineering research and industry best practices.

9. REFERENCES

[1] V. R. Basili. `http://www.cs.umd.edu/~basili/`, 2011. Accessed, October 2011.

[2] A. Brooks, J. Daly, J. Miller, M. Roper, and M. Wood. Replication of experimental results in software engineering. Technical report, International Software Engineering Research Network, 1996.

[3] A. Brooks, M. Roper, M. Wood, J. Daly, and J. Miller. Replication's role in software engineering. In F. Shull, J. Singer, and D. I. K. S. berg, editors, *Guide to Advanced Empirical Software Engineering*, pages 365–379. Springer, 2008.

[4] S. H. Burton, P. M. Bodily, R. G. Morris, C. D. Knutson, and J. L. Krein. Design team perception of development team composition: Implications for Conway's Law. In *Proceedings of the International Workshop on Replication in Empirical Software Engineering Research*, Banff, AB, Canada, September 2011. IEEE Computer Society.

[5] J. C. Carver. Towards reporting guidelines for experimental replications: A proposal. In *Proceedings of the International Workshop on Replication in Empirical Software Engineering Research*, Cape Town, South Africa, May 2010.

[6] F. Q. B. da Silva, M. Suassuna, R. F. Lopes, T. B. Gouveia, A. C. A. França, J. P. N. de Oliveira, L. F. M. de Oliveira, and A. L. M. Santos. Replication of empirical studies in software engineering: Preliminary findings from a systematic mapping study. In *Proceedings of the International Workshop on Replication in Empirical Software Engineering Research*, Banff, AB, Canada, September 2011. IEEE Computer Society.

[7] O. Dieste, E. Fernandez, R. Garcia, and N. Juristo. Hidden evidence behind useless replications. In *Proceedings of the International Workshop on Replication in Empirical Software Engineering Research*, Cape Town, South Africa, May 2010.

[8] Freie Universität Berlin. Replication of 'PatMain'. `http://www.inf.fu-berlin.de/w/SE/PatmainReplicationInfo`, 2011. Accessed, October 2011.

[9] C. Ghezzi. Reflections on 40+ years of software engineering research and beyond: An insider's view. Keynote address at the International Conference on Software Engineering, Vancouver, BC, Canada, May 2009.

[10] A. Jedlitschka and D. Pfahl. Reporting guidelines for controlled experiments in software engineering. In *Proceedings of the International Symposium on Empirical Software Engineering*, pages 95–104, Noosa Heads, Australia, November 2005. IEEE Computer Society.

[11] N. Juristo and S. Vegas. Design patterns in software maintenance: An experiment replication at UPM. In *Proceedings of the International Workshop on Replication in Empirical Software Engineering Research*, Banff, AB, Canada, September 2011. IEEE Computer Society.

[12] C. D. Knutson, J. L. Krein, L. Prechelt, and N. Juristo. Report from the 1st international workshop on replication in empirical software engineering research (RESER 2010). *SIGSOFT Software Engineering Notes*, 35(5):42–44, 2010.

[13] C. Kozak and M. Squire. A secondary data archive for code-level Debian metrics. In *Proceedings of the International Workshop on Replication in Empirical Software Engineering Research*, Banff, AB, Canada, September 2011. IEEE Computer Society.

[14] J. L. Krein, L. J. Pratt, A. B. Swenson, A. C. MacLean, C. D. Knutson, and D. L. Eggett. Design patterns in software maintenance: An experiment replication at Brigham Young University. In *Proceedings of the International Workshop on Replication in Empirical Software Engineering Research*, Banff, AB, Canada, September 2011. IEEE Computer Society.

[15] A. Nanthaamornphong and J. C. Carver. Design patterns in software maintenance: An experiment replication at University of Alabama. In *Proceedings of the International Workshop on Replication in Empirical Software Engineering Research*, Banff, AB, Canada, September 2011. IEEE Computer Society.

[16] L. Prechelt and M. Liesenberg. Design patterns in software maintenance: An experiment replication at Freie Universität Berlin. In *Proceedings of the International Workshop on Replication in Empirical Software Engineering Research*, Banff, AB, Canada, September 2011. IEEE Computer Society.

[17] L. Prechelt, B. Unger, W. F. Tichy, P. Brössler, and L. G. Votta. A controlled experiment in maintenance comparing design patterns to simpler solutions. *IEEE Transactions on Software Engineering*, 27(12):1134–1144, 2001.

[18] P. Runeson, A. Stefik, A. Andrews, S. Grönblom, I. Porres, and S. Siebert. A comparative analysis of three replicated experiments comparing inspection and unit testing. In *Proceedings of the International Workshop on Replication in Empirical Software Engineering Research*, Banff, AB, Canada, September 2011. IEEE Computer Society.

[19] F. Shull, V. Basili, J. Carver, J. Maldonado, G. Travassos, M. Mendonça, and S. Fabbri. Replicating software engineering experiments: Addressing the tacit knowledge problem. In *Proceedings of the International Symposium on Empirical Software Engineering*, pages 7–16, Nara, Japan, October 2002. IEEE Computer Society.

[20] F. Shull, J. Carver, S. Vegas, and N. Juristo. The role of replications in empirical software engineering. *Empirical Software Engineering*, 13(2):211–218, 2008.

[21] M. Vokáč, W. F. Tichy, D. I. K. Sjøberg, E. Arisholm, and M. Aldrin. A controlled experiment comparing the maintainability of programs designed with and without design patterns: A replication in a real programming environment. *Empirical Software Engineering*, 9(3):149–195, 2004.

[22] E. J. Weyuker, R. M. Bell, and T. J. Ostrand. Replicate, replicate, replicate. In *Proceedings of the International Workshop on Replication in Empirical Software Engineering Research*, Banff, AB, Canada, September 2011. IEEE Computer Society.