IDEA GROUP PUBLISHING

701 E. Chocolate Avenue, Suite 200, Hershey PA 17033, USA
Tel: 717/533-8845; Fax 717/533-8661; URL-http://www.idea-group.com

ITP4769

Implementing Software Metrics Programs: A Survey of Lessons and Approaches

Helle Damborg Frederiksen
Aalborg University
Department of Computer Science
Fredrik Bajers Vej 7E
DK-9220 Aalborg East, Denmark
Email: hdf@cs.auc.dk

Jakob Holden Iversen
University of Wisconsin Oshkosh
College of Business Administration
800 Algoma Blvd
Oshkosh, WI 54901
Email: Iversen@uwosh.edu

ABSTRACT

This survey provides an overview of the current research into what determines the success and failure of software metrics programs. We surveyed a selection of the literature, and using a comprehensive framework, analyzed the papers to understand the important factors for implementing a successful metrics program. We determined that the research on implementation of metrics programs is conducted largely by practitioners and generally does not use formalized research methods. The focus is primarily on immediate program implementation and data collection, and much less on the program purpose and the resulting improvements. We conclude that practitioners should consider factors outside of the immediate metrics program implementation, and that academia should conduct more empirical studies of metrics programs using more rigorous research methods.

1 INTRODUCTION: METRICS IN SOFTWARE DEVELOPMENT

Whereas most companies in the software industry recognize the need to measure performance, they find it difficult to implement an effective metrics program to reliably measure their software development effort.

This has given rise to software metrics, which measures the performance of a software developing organization. Research into software metrics has followed two directions: 1) defining how to measure different aspects of the software development effort, e.g., the use of object technology; and 2) implementing a metrics program in an organization. Both have seen many publications address the problems in software metrics (Zuse, 1995). Most of the academic focus has been on defining the metrics themselves, whereas most of the practitioner focus has been on implementation in an organization. A symptom of this disparity is that the metrics that are in most widespread use today were defined in the 1970s (Fenton and Neil, 1999).

There also is research into the process of implementing metrics in software in organizations. In the academic community, the general consensus is that currently, the knowledge of how to define and implement metrics is fairly well developed (Fenton and Neil, 1999). However, it is widely recognized that, for most organizations, implementing a metrics program is difficult, complex, and likely to fail. Recently, a great deal of research has been on factors that cause metrics programs to either succeed or fail (Berry and Jeffery, 2000; Jones, 2001; Iversen and Kautz, 2000; Iversen and Mathiassen, 2000; Philips, 1999; Starrett, 1998; Herbsleb and Grinter, 1998; Hall and Fenton, 1997; Dekkers, 1999). Since there is significant overlap among these factors, but also some contradiction, it is impossible for an organization in a specific situation to use them to implement a metrics program.

This research provides an overview of the current research into implementation of software metrics programs. Section 2 presents the method to select the papers that became our data material, section 3 discusses the framework, section 4 presents the results, and section 5 provides directions for practitioners and researchers.

2 METHOD: SELECTION OF PAPERS

The basis for this research is a selection of papers published in eight important software engineering journals in the period 1990-2002 (to the extent that the journals were published at the time). The following journals and papers were included in the final dataset:

- **IEEE Software:** (Ross, 1990; Pfleeger, 1993; Hall and Fenton, 1997; Offen and Jeffery, 1997; Van Latum et al., 1998; Kautz, 1999; Grable et al., 1999; Rifkin, 2001)
- **Cutter IT Journal/American Programmer:** (Statz, 1999; Dekkers, 1999; Philips, 1999; Mah and Putnam, 1997; Austin, 1997)
- **Software Process - Improvement and Practice:** (Briand et al., 1996; Wieczorek, 1997; Holt, 1997)
- **Journal of Systems and SW:** (Nusendorf and Bunde, 1993; Ebert, 1999; Seddio, 1993)
- **Computer:** (Stark et al., 1994)
- **IEEE Transaction on Software Engineering:** (Daskalantonakis, 1992)
- **Software Engineering Journal:** No relevant papers in the period
- **Communications of the ACM:** No relevant papers in the period

We were interested in papers about successful implementation and operation of metrics programs in software-developing organizations. The criterion to decide whether a paper was inside the scope of this study was whether the authors provided insight into what it would take to implement or operate a successful metrics program. Papers that described cases of metrics programs and provided lessons on this basis were particularly interesting. Outside the scope of this survey were papers whose primary objective was to introduce one or more specific metrics (e.g., a new way of measuring program complexity), and papers that reported on the specific measurement results of a metrics program (e.g. the effectiveness of using function points for estimation purposes).

We looked through the tables of contents of all issues of the eight journals in the period, and made a first selection based on title. If the title indicated that the paper was related to metrics, the abstract was considered, and if the paper was within the scope of interest as described above, it was included in the dataset. Some papers were excluded after reading their text. The final data set included 21 papers. To analyze the papers, we developed a framework (see section 3).

3 FRAMEWORK

The framework includes two types of elements: Elements that concern the paper in general (sections 3.1 and 3.4) and elements that

concern the lessons of the paper (sections 3.2 and 3.3). The elements of the framework are based on works relating to implementation of metrics and measurement. Cameron and Whetten (1983) focus on the effectiveness of the measurement programs. Niessink and Vliet (2001) stress the organizational context of the metrics program. Berry and Jeffery (2000) introduce an instrument for predicting the success of a metrics program, and are concerned with the process and the context of the metrics program. The two latter works are different perspectives on metrics programs. Furthermore, the framework includes general characteristics of the research papers, e.g., research method.

3.1 Cameron and Whetten

To characterize the papers, we build on Cameron and Whetten's (1983) seven guidelines for measuring organizational effectiveness. The guidelines are presented as questions to ask when assessing organizational effectiveness and making organizational improvements (see Table 1). Since one purpose of both the present effort and the guidelines is to help "make studies of effectiveness comparable," we feel that their work is a good match with ours. The questions may be used in a software metrics program to determine what and how to measure, but they may also be used to evaluate the effectiveness of the metrics program itself. We chose the latter as shown in Figure 1.

1. What is the purpose for judging effectiveness?
2. What level of analysis is being used?
3. From whose perspective is effectiveness being judged?
4. On what domain of activity is the judgment focused?
5. What time frame is being employed?
6. What types of data are being used for judgments of effectiveness?
7. What is the referent against which effectiveness is judged?

3.2 Niessink and Vliet

Niessink and Vliet (2001) argue that the success factors for metrics programs presented in the literature typically focus on the 'internals' of the metrics program, e.g. incremental implementation and a well-planned metrics framework. However, to assess the success of the metrics program, the organizational context must be considered, i.e. the measurements should generate value, not just data. In their experience, the main reasons for implementing a metrics program are reporting, monitoring performance, learning, performance improvement, organizational health, and navigation (Niessink and Vliet, 2001).

To facilitate the implementation of a metrics program that supports solving a problem in the organizational context, they introduce a generic process model with four steps, (see Figure 2), explained below:

1. Improvement, analysis: The outset is an organizational problem, which is analyzed.
2. Measurement, implementation: Based on the analysis, the organization must decide on possible causes and solutions to the problem. To find out which of the possible causes is the real cause of the problem, or which of the possible solutions is the best solution, the organization should collect information, which occasionally requires implementation of a metrics program.
3. Measurement, analysis: The data is gathered and analyzed to relate it to the proposed causes and solutions.
4. Improvement, implementation: Finally, the organization solves the problem by implementing the solutions found.

Figure 1. Our work is a literature survey that evaluates the research papers published about implementation of software metrics programs.

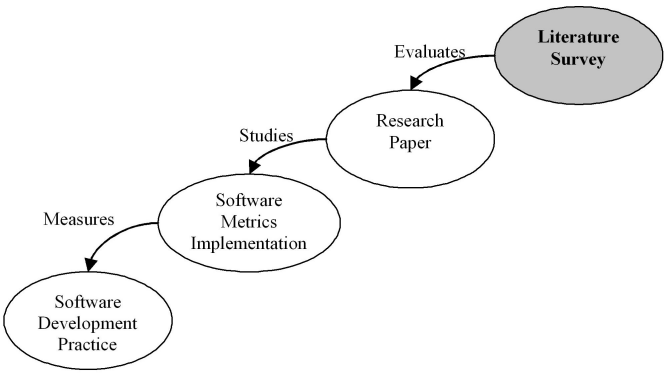

Table 1. Cameron and Whetten's seven questions.

1. **What is the purpose for judging effectiveness?**
2. **What level of analysis is being used?**
3. **From whose perspective is effectiveness being judged?**
4. **On what domain of activity is the judgment focused?**
5. **What time frame is being employed?**
6. **What types of data are being used for judgments of effectiveness?**
7. **What is the referent against which effectiveness is judged?**

Niessink and Vliet (2001) think of the steps labeled "measurement" as internal to the metrics program. We used the four steps in our classification of the lessons presented in the papers. Furthermore, we have added data collection as an element in classifying the lessons. Columns 2 to 5 in Table 2 include data for the part of the framework inspired by Niessink and Vliet.

3.3 Berry and Jeffery

Since metrics programs seem to have high failure rates, Berry and Jeffery (2000) identify the variables that may lead to success or failure of a metrics program. To construct a framework for evaluating and predicting the success of a particular program, they introduce an instrument, which is a structured set of questions to collect data on successful and less successful cases from the people implementing and running the programs in the companies. The instrument includes questions based on advice from experienced practitioners and from theory, on the status, context, inputs, processes, and products of the program. The term context is used in our classification of the lessons, whereas the latter three categories are concerned with the internals of the metrics program and are labeled process in our framework. Columns 6 and 7 in Table 2 include data for the part of the framework inspired by Berry and Jeffery.

3.4 General Elements

The papers are also classified according to general elements. The first is measurement purpose as described by Niessink and Vliet (2001), i.e. what the paper states as the main purpose for implementing a metrics program. The second is for classifying the research method used in the paper, e.g. case study or action research. Furthermore, the level of theory building in the paper is classified, and the researcher's position is classified as being inside or outside the organization with the metrics program. We also record the number and types of companies the paper is based on.

Figure 2. Generic process model.

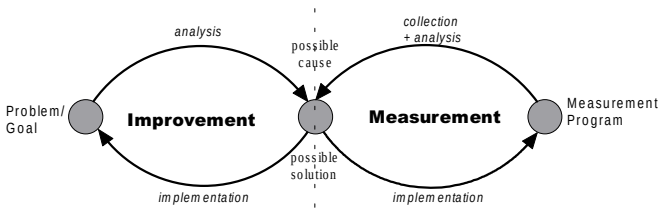

Table 2. Overview of the number of lessons for each paper in the different categories.

Reference	Measurement		Data Collection	Improvement		Context	Process	Total
	Implemen–tation	Analysis		Analysis	Implemen–tation			
(Ross, 1990)	8	2	2	0	0	0	11	11
(Daskalantonakis, 1992)	5	4	0	0	2	1	7	9
(Nusendorf and Bunde, 1993)	12	2	1	0	0	3	12	14
(Pfleeger, 1993)	9	3	4	0	0	0	10	10
(Seddio, 1993)	3	1	1	0	1	1	4	5
(Stark et al., 1994)	4	5	2	0	2	1	9	9
(Briand et al., 1996)	5	0	0	0	0	3	2	6
(Austin, 1997)	2	2	2	0	0	2	5	6
(Hall and Fenton, 1997)	12	4	2	0	1	0	15	15
(Holt, 1997)	5	2	6	0	0	3	9	14
(Mah and Putnam, 1997)	2	2	0	0	1	0	5	5
(Offen and Jeffery, 1997)	5	0	0	1	0	2	5	6
(Wieczorek, 1997)	6	2	0	0	0	3	6	9
(Van Latum et al., 1998)	5	3	3	2	1	0	6	8
(Dekkers, 1999)	6	2	0	1	2	2	6	7
(Ebert, 1999)	8	4	0	0	4	2	12	14
(Grable et al., 1999)	7	2	0	0	0	0	8	8
(Kautz, 1999)	6	1	0	1	2	1	6	8
(Philips, 1999)	1	1	1	0	3	3	3	7
(Statz, 1999)	5	2	2	0	0	0	5	5
(Rifkin, 2001)	0	0	0	1	1	1	0	2
Total	**116**	**44**	**26**	**6**	**20**	**28**	**146**	**178**

4 RESULTS

In the resulting 21 papers we identified 178 lessons that could be classified according to the framework (Table 2). Based on the literature review we developed a set of findings, which are presented with the patterns that emerged from the lessons. Each lesson may appear in several categories.

Finding 1: *Research into implementation of metrics programs focuses narrowly on the installation of the program.*

When metrics programs are introduced, there is little or no consideration of factors that are outside of a narrow scope of determining what to measure, introducing the specific metrics in the organization, and systematically collecting the data. Metrics implementations thus tend to exclude such factors as the purpose of implementing the program, how to analyze the data once collected, and how and what to improve based on the measurements.

Using the labels measurement and improvement from Niessink and Vliet (2001) the following pattern emerges:

- There are 186 lessons on measurement and data collection.
- There are 26 lessons on improvement.
- Nine papers have no lessons on improvement, i.e. these papers focus only on measurement. Another five have only vague lessons on improvement.

Although our data collection may be biased since we focused on studies with relevance to implementing metrics programs, and excluded papers that assumed a metrics program as the foundation for making improvements, consideration of improvement should be a substantial theme during implementation of metrics programs (Niessink and Vliet, 2001).

According to Berry and Jeffery (2000), context is important to the success of a metrics program, however, the lessons from the papers represent a different picture. Only 28 lessons are on context whereas 146 are on process. In addition, about half of the papers had no or vague lessons on context, e.g. a lesson could be stated as "beware of context."

Furthermore, of 186 lessons on measurement, 116 are on implementation, 26 on data collection, and 44 on analysis. There is much advice on how to get to the data but less on how to interpret the data for further use.

Finding 2: *Improvements focus more on how to implement the solutions and less on how to identify the problems in the organization.*

Of 26 lessons on improvement, 20 are on implementation and 6 on analysis. This pattern might have emerged because the papers assume that the problems are known prior to implementing a metrics program. However, it is important to link the metrics program to the actual problems of the organization to obtain success. Hence, linking from the problems to the measurement should be emphasized in the literature. Furthermore, the measurements could be used to identify potential problems or possible causes of problems.

Finding 3: *The main purpose of metrics programs is to improve performance.*

To evaluate the success or failure of a metrics program, the purpose of the program should be determined a priori. Of 21 papers, 14 list performance improvement as a major reason for implementing a metrics program, six list monitoring performance, and only one lists learning as the main reason. Four papers list no real purpose of the metrics program. From the survey we could conclude that to consider a metrics program successful, it should contribute to the improvement of organizational performance. However, this is not necessarily reflected in the success criteria for the programs (see Finding 4).

Finding 4: *Success criteria for the implementation of the metrics program are not declared explicitly or are not well defined.*

Most of the papers stated purposes and/or success criteria for the metrics program. However, often they were broad statements that would be difficult to confirm. Examples include "Improve software development," "Meet the needs of those providing the data," "Metrics considered useful by participants," and "Widespread corporate use." Far more operational were goals such as "A program that survives more than 2 years," and "Supporting business goals."

Only two papers report any attempt to validate the success of the metrics program. One uses a questionnaire; the other includes a description of three projects where metrics were applied successfully. Although several papers claim that the metrics program was successful, no attempt was made to support such claims with empirical evidence. The perspective used to determine whether a program is successful is partly the software people and partly the group of metrics and software process improvement people.

Finding 5: *The studies of the metrics programs are not based on a rigorous research method, and the level of theory building in the studies is very low.*

None of the papers specify the research method used. Eleven papers are case studies where one or more of the authors were active participants on the metrics program. Many papers draw on the authors' experience in the field in general, but do not explicitly apply any empirical methods. Consequently, most of the empirical papers studied one organization. The time frame of the studies is often not explicit, but could be deduced to be long term due to the extensive experience and involvement of the authors.

5 CONCLUSION

We presented a general framework for evaluating papers on implementing software metrics programs. Researchers can use this framework to provide an overview of the current research on implementing metrics programs. Practitioners can use it to provide a comprehensive view of the determinants that are instrumental in implementing and operating successful metrics programs.

The findings have implications for research:

- More rigorous and empirical research into implementation of metrics programs is needed.
- Involvement of practitioners in the research process is good, as it provides for rich case studies, but may limit the critical voice of the researcher.
- There is a need for more theory in studying and understanding metrics programs in order to broaden the understanding of the metrics programs and their success or failure.

Likewise, the findings have implications for practice:

- There is a need to broaden the focus when implementing metrics programs. The role of the measurements in the broader context of improvement should be clarified in the organization.
- Explicit, attainable, and measurable success criteria should be specified in advance.

Our work is based on 21 papers, and may not be representative of all the ways in which metrics programs are implemented. However, the patterns that emerge from the lessons cannot be ignored, and are confirmed by our experience with metrics programs during the last five years. Hopefully, the implications of this survey will be considered in future research and practice of metrics program implementation.

6 BIBLIOGRAPHY

Austin, R. D. (1997), "Measurement and Behavior," American Programmer (X:11), pp. 12-16.

Berry, M. and Jeffery, R. (2000), "An Instrument for Assessing Software Measurement Programs", in Empirical Software Engineering Conference, Staffordshire, UK.

Briand, L. C., Differing, C. M. and Rombach, H. D. (1996), "Practical Guidelines for Measurement-Based Process Improvement," Software Process - Improvement and Practice (2), pp. 253-280.

Cameron, K. S. and Whetten, D. A. (1983), "Some Conclusions about Organizational Effectiveness", in Cameron, K. S. and Whetten, D. A. (Eds), Organizational Effectiveness: A Comparison of Multiple Models, Academic Press, New York.

Daskalantonakis, M. K. (1992), "A Practical View of Software Measurement and Implementation Experience Within Motorola," IEEE Transactions on Software Engineering (18:11), pp. 998-1010.

Dekkers, C. A. (1999), "The Secrets of Highly Successful Measurement Programs," Cutter IT Journal (12:4), pp. 29-35.

Ebert, C. (1999), "Technical Controlling and Software Process Improvement," The Journal of Systems and Software (46:1), pp. 25-39.

Fenton, N. E. and Neil, M. (1999), "Software Metrics: Successes, Failures and New Directions," The Journal of Systems and Software (47:2-3), pp. 149-157.

Grable, R., Jernigan, J., Pogue, C. and Divis, D. (1999), "Metrics for Small Projects: Experiences at the SED," IEEE Software (16:2), pp. 21-29.

Hall, T. and Fenton, N. (1997), "Implementing Effective Software Metrics Programs," IEEE Software (14:2), pp. 55-65.

Herbsleb, J. D. and Grinter, R. E. (1998), "Conceptual Simplicity Meets Organizational Complexity: Case Study of a Corporate Metrics Program", in The 20th International Conference on Software Engineering, Kyoto, Japan.

Holt, J. (1997), "Software Metrics - Real World Experiences," Software Process - Improvement and Practice (3:3), pp. 155-163.

Iversen, J. H. and Kautz, K. (2000), "The Challenge of Metrics Implementation", in IRIS 23, Uddevalla, Sweden.

Iversen, J. H. and Mathiassen, L. (2000), "Lessons from Implementing a Software Metrics Program", in Hawaii International Conference on System Sciences (HICSS-33), Wailea, Hawaii.

Jones, C. (2001), "Software Measurement Programs and Industry Leadership," Crosstalk - The Journal of Defense and Software Engineering (14:2), pp. 4-7.

Kautz, K. (1999), "Making Sense of Measurements for Small Organizations," IEEE Software (16:2), pp. 14-20.

Mah, M. C. and Putnam, L. H. (1997), "Software by the Numbers: An Aerial View of the Software Metrics Landscape," American Programmer (X:11), pp. 3-11.

Niessink, F. and Vliet, H. V. (2001), "Measurement Program Success Factors Revisited," Information and Software Technology (43:10), pp. 617-628.

Nusendorf, R. E. and Bunde, D. C. (1993), "A Guidebook and a Spreadsheet Tool for a Corporate Metrics Program," The Journal of Systems and Software (23), pp. 245-255.

Offen, R. J. and Jeffery, R. (1997), "Establishing Software Measurement Programs," IEEE Software (14:2), pp. 45-53.

Pfleeger, S. L. (1993), "Lessons Learned in Building a Corporate Metrics Program," IEEE Software (10:3), pp. 67-74.

Philips, D. (1999), "Back to Basics: Metrics that Work for Software Projects," Cutter IT Journal (12:4), pp. 36-42.

Rifkin, S. (2001), "What Makes Measuring Software So Hard?" IEEE Software (18:3), pp. 41-45.

Ross, N. (1990), "Using Metrics in Quality Management," IEEE Software (7:4), pp. 80-81, 85.

Seddio, C. (1993), "Integrating Test Metrics within a Software Engineering Measurement Program at Eastman Kodak Company: A Follow-up Case Study," The Journal of Systems and Software (20), pp. 227-235.

Stark, G., Durst, R. C. and Vowell, C. W. (1994), "Using Metrics in Management Decision Making," Computer (27:9), pp. 42-48.

Starrett, E. C. L. (1998), "Measurement 101," Crosstalk (11:8), pp. 24-28.

Statz, J. (1999), "What's Practical About Software Measurement?" Cutter IT Journal (12:4), pp. 4-10.

Van Latum, F., Van Solingen, R., Oivo, M., Hoisl, B., Rombach, D. and Ruhe, G. (1998), "Adopting GQM Based Measurement in an Industrial Environment," IEEE Software (15:1), pp. 78-86.

Wieczorek, I. (1997), "On the Establishment of Successful Measurement Programs in Industry," Software Process - Improvement and Practice (3), pp. 191-194.

Zuse, H. (1995), "History of Software Measurement," September 14, 1995. Accessed April 30 2001. Web page. Available from http://irb.cs.tu-berlin.de/~zuse/metrics/History_00.html.

Big Data Summarization Using Novel Clustering Algorithm and Semantic Feature Approach
Shilpa G. Kolteand Jagdish W. Bakal (2017). *International Journal of Rough Sets and Data Analysis (pp. 108-117).*
www.irma-international.org/article/big-data-summarization-using-novel-clustering-algorithm-and-semantic-feature-approach/182295