Association for Information Systems
AIS Electronic Library (AISeL)

AMCIS 1996 Proceedings

Americas Conference on Information Systems (AMCIS)

8-16-1996

Process and Context of Software Metrics Deployment

Haidong SOng
University of Southern Queensland, song@usq.edu.au

Arun Rai
Southern Illinois University at Carbondale

Arkalgud Ramaprasad
Southern Illinois University at Carbondale

Process and Context of Software Metrics Deployment

Haidong Song
Department of Information Systems, Faculty of Business
University of Southern Queensland, Toowoomba, QLD 4350, Australia
Tel: 61-76-31-1687, Fax: 61-76-31-1782, Email: song@usq.edu.au
Arun Rai, Arkalgud Ramaprasad
Pontikes Center for Management of Information
Department of Management, Southern Illinois University at Carbondale
Carbondale, IL 62901-4627, U.S.A.

Introduction

Software metrics are used to compare and profile software development. They are known, quantifiable, objective and subjective measures, and are increasingly seen as valuable tools in the development of high quality software products. Although the deployment of software metrics has been deemed a major factor in the transformation of software development from an art to an engineering discipline, the deployment process itself has not been explored in detail.

This paper reports the results of a case study examining metrics deployment, the underlying organizational processes, and the effects of the deployment context. The study included multiple system developmen projects in one organization. Our aim was to examine if the choice and deployment of metrics are influenced by contextual elements which differentiate software development projects and teams. By selecting multiple projects from the same organization, we could focus our attention on contextual aspects that differentiate projects. By design, the larger organizational context is the same for these projects.

Case Study of Software Metrics Deployment

The case study involved three distinctive development projects from a large retail corporation. The projects varied on several key dimensions, namely: business applications, development methodology used, and characteristics of the development team (Table 1). Data were collected through semi-structured interviews with development team members and users. In addition, project documents and follow-up telephone calls were used to resolve inconsistencies and provide data triangulation.

Project SERVICE selected from the product and services division is large and complex. The project involves multiple applications, more than 100 databases, and has a large user base (16,000 service technicians).

Project CREDIT was selected from Credit Information System division. This system was used across 32,000 check-out counters nationwide to keep customer signature for credit card purchases electronically and presently about 85% of receipts are stored electronically.

Project MARKETING was an enhancement project to the assortment planning function in the marketing division. The system is used by over 100 marketing associates, all located at the company headquarters.

The case studies allowed for replication of inquiry and given the different contexts of the cases, additional insights can be garnered [Yin, 1984]. The aim here is to attain analytic generalizability as opposed to statistical generalizability.

Discussion of Case Study Results

Metrics Applied

The diverse set of software metrics used in the three projects can be broadly classified into eight categories: overall quality metrics, software size-based metrics, schedule-based metrics, cost metrics, benefit/value metrics, defects metrics, reliability metrics, and user acceptance metrics. The metrics focus on the output of the project, not on the development process and intermediate deliverables. In addition, the priority of the metrics differed across the three projects.

The overall quality indicator used in a project determined the category of metrics which received the highest priority, and the diligence with which a metric was measured. Further, most metrics used are direct numerical measures or estimates. Time spent, number of calls, number of tickets, number of defects, number of assortment plan processed, etc. - are all numerical measures of product or process attributes which can be directly recorded during the development process. Perceptual characteristics such as user acceptance or user satisfaction were quantified using convenient surrogates. For example, the number of transactions processed through the MARKETING system was treated as a measurement of user acceptance of the system.

Across the three projects, the most visible metrics were used, as suggested by Grady [1992]. The selection of metrics and their definition was at the discretion of each project team.

Deployment Process

The case study shows that the deployment process of software metrics can be decomposed into five phases: definition, data collection, validation, analysis, and use. The meanings of the first three phases are self-evident. Analysis refers to the process of manipulation of metrics data in such ways as tabulation, classification, and statistical calculation. Use refers to the process by which the metrics data are applied to provide insights into the project or to manage or improve the development process.

There are two configurations of the five phases in the deployment process, corresponding to the two essential approaches which drive the overall deployment process -- top-down and bottom-up -- as suggested be Hetzel [1993]. In the top-down approach, high-level goals and needs determine the derivation of metrics and the subsequent use of metrics. In

the bottom-up approach, metrics were derived based on the readily observable aspects of the software and the development process according to subjectively perceived value and usefulness of the data. The metrics were subsequently used for feedback.

Furthermore, not all the five phases were present in all projects. In a number of cases, metrics were used without proper validation. In addition, in some cases, the definition of metrics evolved over time without going through a formal definition process. However, in all cases, the definition of metrics had a close association with the overall quality indicator of the project, or the requirements for project management.

Contextual Factors

The case study suggests that the characteristics of the project team, characteristics of the development process, characteristics of the system, characteristics of the software quality assurance function, and management commitment have significant impact on the deployment process of software metrics.

Among the characteristics of the development team, having a champion, the team s receptivity towards quality assurance and metrics, team members' need and ability to learn, and user participation and involvement have significant impact on metrics deployment. These findings confirm that bottom-up commitment from everyone in the project team is the most important factor for the success of system quality improvement efforts, as suggested by Kan [1995].

There is also a strong relationship between development process maturity and deployment process of metrics which has been suggested by a number of researchers such as Pfleeger and McGowan [1990], and the Quantum Report [1992]. However, there is little evidence to support the influence of automated tools on deployment process as claimed by Pfleeger [1993].

The presence of a quality assurance (QA) function facilitates the successful implementation of assurance activities and the deployment of metrics. However, even in the absence of a formal QA function, various quality assurance activities were performed, since some developers implicitly assumed the responsibilities of quality assurance function in different development phases. Therefore, one way to achieve quality is to integrate quality related activities with every developer's effort, and ensure it through team member control. Nevertheless, the presence of formal QA function as a managerial control would also be beneficial.

It was observed that system complexity on the one hand is positively correlated with the introduction of software metrics, but on the other hand has a negative impact on the deployment of the metrics due to the embedded difficulty in applying and verifying metrics for complex systems. In addition, because the due dates in all three projects were set very tight and the project teams were under constant pressure to complete the project within the specified time frame, the measurement of quality, i.e. the deployment of software metrics, was sacrificed for project progress in terms of deadlines and milestones.

Like in any other IS development and application effort, management commitment in the form of quality initiative and sponsorship of quality assurance practices was an important factor in the deployment process of software metrics.

Conclusions and Implications

Management by fact is one the ten core values and concepts of the Baldrige Quality Award, and software metrics are the major means towards this end in the are of software development. Yet there is a paucity of systematic research on the concept. This study explored the type of metrics used in IS development projects, the phases in the deployment process, and the contextual factors which could affect the deployment process. The results show that the definition of software metrics is project dependent, evolves locally in context of the specific project. The deployment process consists of five dsitinctive pahses, and a dynamic relationships exists among the phases. Does project context influence the choice and deployment metrics? The answer from our study is "yes", and five categories of contextual factors are identified. Additional research using multiple methodologies such as survey and longitudinal approaches should be conducted to compare and contrast the deployment process in different organizational contexts.

REFERENCE.

Grady, R. Practical Software Metrics for Project Management and Process Improvement. Englewood Cliffs, NJ: Prentice-Hall, 1992.

Hetzel, B. Making Software Measurement Work: Building an Effective Measurement Program. Boston, MA: QED Publishing Group, 1993.

Kan, S. Metrics and Models in Software Quality Engineering, Reading, MA: Addison-Wesley Publishing Company, 1995.

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

Pfleeger, S.L. & McGowan, C.L. Software Metrics in a Process Maturity Framework, Journal of Systems and Software, 12, July 1990, 255-261.

Quantum Report: A Measurement-Based Framework for the Assurance of Software Quality, The Department of Trade and Industry, London, United Kingdom, 1992.

Yin, R.K. Case Study Research, Applied Social Research Methods Series, 5, Beverly Hills, CA: Sage Publications, 1984.

Characteristics	SERVICE	CREDIT	MARKETING
User Base	16000 technicians	32000 counters	100-200 partners
Functional Area	Product services	Retail & credit	Marketing
Code Complexity	High	Low	Medium
Database	More than 100	1 DB with 7 copies	Many

Methodology	Some SSDM	Not used	SSDM based

Table 1 Characteristics of Three Case Study Projects