Software Process Improvement in Small Scale Organizations: An Empirical Study

Lakhwinder Sharma
Assistant Professor, Department computer Science, Swami Parmanand College of Engg. & Technology, Mohali.

Neeraj Sharma
Associate Professor, Department of Computer Science, Punjabi University, Patiala.

ABSTRACT
Though, small organizations do not have the same funding opportunities as that of large organizations yet they too need of software process improvement programs. It is better to initiate a software process improvement program as early as possible, irrespective of the organization size. The small organization grows over the time and become a large organization and at that time an idea of incorporating software process improvement may cause economic feasibility for it. In this paper an attempt is made to analyze statistically various performance parameters involved in software process improvement for small scale organizations.

General Terms
Software Engineering, Statistical Hypothesis Testing.

Keywords
Software Process Improvement (SPI), Performance Parameters, Small Scale Organizations.

1. INTRODUCTION
Software process improvement is required to increase the productivity of software companies. In SPI primary goal is to increase the quality of the produced software and to keep budget and time under control. In literature a number of models such as Capability Maturity Model, Capability Maturity Model Integrated and ISO 9001:2000 have been suggested for large scale organizations. Process improvement has been widely and continuously used in manufacturing and software industry. Once the deficiencies are identified in the current processes, SPI starts and ends when a certain satisfactory or predefined level is attained [1]. Therefore, technological improvement is necessary to improve the quality of the product within time in order to compete in market. SPI framework for small scale organization has been reported previously, [8]. The organization with fewer resources need better process improvement and profitability measures [3]. Once process improvement is initiated properly, it results in improved processes and better profit margin. Organizational structure, road maps, assessed methods and a good plan lead towards successful results of process improvement [2]. Customer satisfaction, continuous improvement and less staff turnover show the strong business profitability aspects of an organization [6]. One major characteristic of process improvement is to emphasize the continuous improvement of products as well as of organizational processes in terms of performance, stability, compatibility [7]. After the application of process improvement the organization feels the competitiveness, increase in performance, profitability and innovations in the processes which show the benefits. Process improvement focuses towards the development of the practices, improved quality of the products, reliability, productivity, and customer and employee satisfaction. The change occurs in the shape of good staff, improved technical system, organized structure and better management practices [9]. SPI is an effective way to improve product quality, meet market needs, reliability and return on investment [14]. Managers also facilitate those activities which can also be helpful in successful and effective change. Defining the software process is the first step in improving the software process. It is unusual that a software process is defined during a start-of a business. Defining software process means documenting and understanding all parts of it. And if there is already software process used, it is necessary to capture and understand it.

In this paper an attempt is made to analyze statistically various performance parameters involved in SPI for small scale organizations by collecting field data using research methods and advanced statistical techniques.

1.1 Problem Definition
Evaluating a project is a fundamental step towards improving the organization. Today we have the Capability Maturity Model (CMM) and Software Process Improvement for Capability Determination (SPICE) that are based on best practice and give a method to improve the organization. However, these models are not so appropriate for small or medium sized organizations. The CMM and SPICE are very effort intense and costly to perform and measured benefit is often not seen in a short time perspective. One natural step would be to evaluate each specific project after it has been finished and then use the outcome to improve the performance of project(s) in an organization. This evaluation will not only give feedback for improvement but also more specific for the organization using it.

1.2 Notations and Abbreviations

n: Sample size
µ: Population mean
$\bar{x}$: Sample mean
s^2: Sample variance
t_{cal} : Value of t-test statistics
t_{tab} : Tabulated value of t-test statistics
α: Level of significance
SPI: Software Process Improvement
PP: Performance Parameters
CMM: Capability Maturity Model
SPICE: Software Process Improvement for Capability Determination.

2. RESEARCH DESIGN AND SAMPLING FRAME

In this study, small scale software organizations in North India have been assumed as target population. These organizations, have been marked as 1,2…..N. Then simple random sampling scheme (without replacement) has been adopted for drawing a sample of size 10 organizations representative of the entire population in the region.

2.1 Null Hypothesis

A hypothesis of no difference is called null hypothesis and it is denoted by H_0. According to R.A.Fisher, the null hypothesis is the hypothesis which is tested for possible rejection under the assumption that it is true.

2.2 Alternative Hypothesis

Any hypothesis which is complementary to null hypothesis is called an alternative hypothesis and it is denoted by H_1.

3. TEST STATISTICS

Student t test for significance of mean

Suppose we want to test:

1. If a random sample x_i (i=0, 1……..n) of size n has been drawn from a normal population with specified mean say µ.
2. If the sample mean differs significantly from the hypothetical value µ of the population mean.

Then the t statistic is

$$t_{cal} = \frac{(\bar{x} - \mu)}{\frac{s}{\sqrt{n}}}$$

Where µ denotes the population mean and $\bar{x}$ denotes the sample mean and is given by

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

and s^2 is the sampling variance and is given by

$$s = \frac{1}{n-1}\sqrt{(x-\bar{x})^2}$$

and n is the sample size.

3.1 Level of Significance

We compare the $t_{(cal)}$ with $t_{(tab)}$ at pre-defined (α) level of significance (for e.g. 5% level of significance or 1% level of significance etc.) with (n-1) degree of freedom.

3.2 Test Criteria

1. **If** $t_{cal} < t_{tab\,(\alpha,(n-1))}$ d.f then we accept the null hypothesis at α level of significance.
2. **If** $t_{cal} > t_{tab\,(\alpha,(n-1))}$ d.f then we reject the null hypothesis at α level of significance.

4. THE FIELD SURVEY EXECUTION

The field survey was conducted through interviews with North India organizations that have departments of software development or similar. Organizations that practice some kind of evaluation were included in the survey, just as organizations that have no model for evaluation or have no funds for software process improvement purpose. The organizations that do not have a model for evaluation did perform some kind of evaluation but in a more non-formal way.

Small Scale Organizations that are taken in the survey are following:

Table 1.1: Small Scale Organizations Data

Companies	**Business Segment**	**Interview Person Position**
Company 1 IMPETUS	Software Product Development	Process Manager
Company 2 DATANOVA	Software and Hardware	Process Manager
Company 3 VEYOM TECH	IT-consulting	Process Manager
Company 4 VH TECHNO SOFT	Software and Hardware	Process Manager
Company 5 INFORMATICA	IT-consulting	Process Manager
Company 6 ISHIR INFOTECH	IT-consulting	Process Manager
Company 7 PATH INFOTECH Ltd.	Software and Hardware	Process Manager
Company 8 RAYS SOFT	IT-consulting	Process Manager
Company 9 SKY TECHNOLOGY	IT-consulting	Process Manager
Company 10 SOFTWARE MANAGEMENT SOLUTION	Software and Hardware	Process Manager

5. PROCESS PERFORMANCE MEASURES

5.1 Efficiency

Efficiency is used to refer to a number of related concepts. It is the using of resources in such a way as to maximize the production of goods and services. A software process can be called economically efficient if:

- No one can be made better off without making someone else worse off.
- More output cannot be obtained without increasing the amount of inputs. Production proceeds at the lowest possible per-unit cost

H_0 = Efficiency of the processes of all the companies does not differ significantly.

H_1 = Efficiency of the processes of all the companies differ significantly.

Mean $(\bar{x})$ = 1.6 μ = 1.29…......... (5.1.1)

$s = \frac{1}{n-1}\sqrt{(x-\bar{x})^2}$ = 0.2811…………. (5.1.2)

$t_{cal} = \frac{(\bar{x}-\mu)}{\frac{s}{\sqrt{n}}}$ = 3.48 and t_{tab} = 2.26….. (5.1.3)

Hence $t_{cal} > t_{tab\,(0.05,9)}$ So H_0 rejected and H_1 accepted for α level of significance.

5.2 Reliability

Software Process Reliability is the application of statistical techniques to Data Collected during system development and operation to specify, predict, estimate, and assess the reliability of software-based systems. "Software Reliability Engineering (SRE) is a standard, proven best practice that makes testing more reliable, faster, and cheaper. It can be applied to any system using software and to frequently-used members of software component libraries.

H_0 = Reliability of the processes of all the companies does not differ significantly.

H_1 = Reliability of the processes of all the companies differ significantly.

Mean $(\bar{x})$ = 1.9 μ = 1.65……….. (5.2.1)

$s = \frac{1}{n-1}\sqrt{(x-\bar{x})^2}$ = 0.3220...……….. (5.2.2)

$t_{cal} = \frac{(\bar{x}-\mu)}{\frac{s}{\sqrt{n}}}$ = 2.45 and t_{tab} =2.26....... (5.2.3)

Hence $t_{cal} > t_{tab\,(0.05,9)}$ So H_0 rejected and H_1 accepted for α level of significance.

5.3 Testability

Testability is a non-functional requirement important to the testing team members and the users who are involved in user acceptance testing. It can be defined as the property that measures the ease of testing a piece of code or functionality, or a provision added in software so that test plans and scripts can be executed systematically.

H_0 = Testability of the processes of all the companies does not differ significantly.

H_1 = Testability of the processes of all the companies differ significantly.

Mean $(\bar{x})$ = 1.7 μ = 1.59…..…… (5.3.1)

$s = \frac{1}{n-1}\sqrt{(x-\bar{x})^2}$ = 0.2844……….... (5.3.2)

$t_{cal} = \frac{(\bar{x}-\mu)}{\frac{s}{\sqrt{n}}}$ = 1.27 and t_{tab} =2.26…... (5.3.3)

Hence $t_{cal} < t_{tab\,(0.05,9)}$ So H_0 accepted and H_1 rejected for α level of significance.

5.4 Usability

Usability is a term used to denote the ease with which people can employ a particular tool or other human-made object in order to achieve a particular goal. Usability can also refer to the methods of measuring usability and the study of the principles behind an object's perceived efficiency or elegance.

H_0 = Usability of the processes of all the companies does not differ significantly.

H_1 = Usability of the processes of all the companies differ significantly.

Mean $(\bar{x})$ = 2 μ = 1.77……….…. (5.4.1)

$s = \frac{1}{n-1}\sqrt{(x-\bar{x})^2}$ = 0.3143............…. (5.4.2)

$t_{cal} = \frac{(\bar{x}-\mu)}{\frac{s}{\sqrt{n}}}$ = 2.31 and t_{tab} = 2.26….. (5.4.3)

Hence $t_{cal} > t_{tab\,(0.05,9)}$ So H_0 rejected and H_1 accepted for α level of significance.

5.5 Expendability

Relative efforts required to expand software capabilities and/or performance by enhancing current functions by adding new functionality.

H_0 = Expandability of the processes of all the companies does not differ significantly.

H_1 = Expandability of the processes of all the companies differ significantly.

Mean $(\bar{x})$ = 1.4 μ = 1.25……....…. (5.5.1)

$s = \frac{1}{n-1}\sqrt{(x-\bar{x})^2}$ = 0.1721……….…. (5.5.2)

$$t_{cal} = \frac{(\bar{x}-\mu)}{\frac{s}{\sqrt{n}}} = 2.75 \text{ and } t_{tab} = 2.26 \ldots\ldots (5.5.3)$$

Hence $t_{cal} > t_{tab\,(0.05,9)}$ So H_0 rejected and H_1 accepted for α level of significance.

5.6 Flexibility

Ease of effort for changing the software's mission or data to meet changing needs and requirements.

H_0 = Flexibility of the processes of all the companies does not differ significantly.

H_1 = Flexibility of the processes of all the companies differ significantly.

Mean $(\bar{x}) = 1.5$ $\mu = 1.36$…………. (5.6.1)

$$s = \frac{1}{n-1}\sqrt{(x-\bar{x})^2} = 0.1757 \ldots\ldots\ldots (5.6.2)$$

$$t_{cal} = \frac{(\bar{x}-\mu)}{\frac{s}{\sqrt{n}}} = 2.51 \text{ and } t_{tab} = 2.26 \ldots (5.6.3)$$

Hence $t_{cal} > t_{tab\,(0.05,9)}$ So H_0 rejected and H_1 accepted for α level of significance.

5.7 Reusability

Reusability is the likelihood segment of source code that can be used again to add new functionalities with slight or no modification. Reusable modules and classes reduce implementation time, increase the likelihood that prior testing and use has eliminated bugs and localizes code modifications when a change in implementation is required.

H_0 = Reusability of the processes of all the companies does not differ significantly.

H_1 = Reusability of the processes of all the companies differ significantly.

Mean $(\bar{x}) = 1.3$ $\mu = 1.17$…………. (5.7.1)

$$s = \frac{1}{n-1}\sqrt{(x-\bar{x})^2} = 0.1610 \ldots\ldots\ldots (5.7.2)$$

$$t_{cal} = \frac{(\bar{x}-\mu)}{\frac{s}{\sqrt{n}}} = 2.55 \text{ and } t_{tab} = 2.26 \ldots\ldots (5.7.3)$$

Hence $t_{cal} > t_{tab\,(0.05,9)}$ So H_0 rejected and H_1 accepted for α level of significance.

6. CONCLUSION

From equations (5.1.3), (5.2.3), (5.4.3), (5.5.3), (5.6.3) and (5.7.3) it has been observed that calculated values of t-statistics are greater than its tabulated value. Therefore, the corresponding null hypothesizes (H_0's) have been rejected for predefined level of significance whereas from equation (5.3.3) it has been observed that calculated value of t-statistics is less than its tabulated value. Therefore, the corresponding null hypothesizes (H_0) has been accepted for predefined level of significance. Hence we conclude that efficiency, reliability, usability, expandability, flexibility and reusability of all organizations differ significantly and testability of all organizations does not differ significantly. Despite of this little academic attention has been devoted to software process improvement for small software organizations, even though the rules for managing projects in small organizations differs from managing project(s) in large organizations. Due to which Software process improvement efforts which are designed for large organization fails when implemented in small scale organization. Therefore, a proper framework for SPI consisting of all above discussed parameters should be designed and implemented properly for improving the performance of small scale organizations.

7. ACKNOWLEDGMENT

The authors are very thankful to all the small scale organizations who contributed to field survey execution. The authors would also like to thank SPCET and Department of computer science Punjabi University Patiala for providing library and other necessary research facilities.

8. REFERENCES

[1] P.ClementsandL. Northrop. 2002 Software product lines: practices and patterns. Addison-Wesley, Boston.

[2] Green, Stewart., Regev, Gil, Soffer, Pnina, Zdravkovic, Jelena. 2007 Business Processes and Support Systems: Design for Flexibility software process improvement and practice, Wiley Inter Science.

[3] Werth, Laurie Honour 1993 Introduction to Software Process Improvement. Carnegie Mellon University, CMU/SEI-93-EM-8.

[4] CMMI Product Team 2002 Capability maturity model integration version1.1 continuous representation. CMU/SEI-2002-TR-028.

[5] J. Olsson. 1999 Requirements for an ISO 9000 certification on a small company. Master's thesis, Blekinge Institute of Technology, Karlskrona/Ronneby, Sweden.

[6] G. Kotonya and I. Sommerville. 2004 Requirements Engineering: Processes and Techniques. Wiley, Chichester.

[7] S.Talburt.Ethnographic responsibility without the "real´´. Journal of Higher Education, 75(1):80–104, 2004.

[8] D. Vasiljevic and S. Skoog. 2003 A software process improvement framework for small organizations: A research approach. Master's thesis, Blekinge Institute of Technology, Ronneby, Sweden.

[9] S.Zahran. 1998 Software Process Improvement: Practical Guidelines for Business Success. Addison-Wesley, Harlow.

[10] Ferguson, Pat, Leman, Gloria, Perini, Prasad, Renner, Susan, Seshagiri, Girish 1999 Software Process Improvement Works. Technical report. CMU/SCI-99-TR-027, ECS-TR-99-026.

[11] Greene, J.W.E. 1998 Software Process Improvement: management commitment measures and motivation.

[12] Dyba, Tore, "An Empirical Investigation of the Key Factors for Success in Software Process Improvement", vol. 31, no. 5, IEEE, May 2005.

[13] Allison, I., Merali, Y. 2007 Software process improvement as emergent change: A structurational analysis Elsevier.

[14] Varkoi, Timo, "Management of Continuous Software Process Improvement", IEEE 2002.