SPI Challenges Of Small And Medium Sized Software Companies – Problems And Prospects

Saranya A, Research Scholar, Madurai Kamaraj University, Madurai, India.
Dr S Kannan, Associate Professor, Madurai Kamaraj University, Madurai, India.

Abstract

Software Process Improvement has been widely studied and applied in the Software Industry with the hopes of improving the quality of the various engineered software products. The Capability Maturity Model from the SEI is a very good example for SPI attempts in Software Organizations. But the increasing number of small and medium sized software organizations presents new SPI Challenges. Owing to the challenges unique to such organizations, doubts arise about the feasibility of applying SPI methods like CMM. The paper reviews the challenges faced by the small and medium sized organizations and presents 2 attempts in the literature to tackle the problem of SPI in Small and Medium Sized Organizations.

1. Introduction

The drive for improvement in Software Engineering practices lead the Software Engineering Institute (SEI) to come up with Capability Maturity Model (CMM). Over the years, the CMM has come to be accepted as roadmap for Software Process Improvement (SPI) [2]. The Software CMM proposes a five-level framework that can serve as a basis for both assessing the maturity of processes and improvement of the processes sued by the organization.

1.1 An Overview of the Software CMM

The software CMM defines 5 levels and 18 Key Process Areas (KPAs). The validity of the 5 maturity levels for guiding Software Process Improvement has been exemplified by many case studies and surveys [10, 11, 12]. The architecture of the CMM is comprised of a ladder with an initial level and 4 steps.

Level 1

Level 1 characterizes a state of chaos in the environment. The success of an organization at this level is attributable only to the competency of the people involved.

Level 2

Level 2 – called “repeatable” – implies that success can be repeated but only for similar projects. Different projects continue to work differently. The Key Process Areas at this level include:

- Requirements Management
- Software Project Planning
- Software Project Tracking and Oversight
- Software Subcontract Management
- Software Quality Assurance
- Software Configuration Management

Level 3

Level 3 is characterized by the presence of an organizational common process, but one that is tailored for individual projects in a controlled manner. The Key Process Areas are:

- Organizational Process Focus
- Organizational Process Definition
- Training Programme

- Integrated Software Management
- Software Product Engineering
- Inter-group coordination
- Peer Reviews

Level 4

This level called "Managed", is characterized by measurements of the process and the products. Key Process Areas are:

- Quantitative Process Management
- Software Quality Management

Level 5

The "optimizing" Level 5, indicates a culture of continuous process improvement. Key Process Areas include:

- Defect Prevention
- Technology Change Management
- Process Change Management

2. Defining Small Organizations

The task of "Defining" a small organization tends to be highly subjective. The 1998 SEPG Conference Panel defined "small" as 3-4 months duration and 5 or lower staff [4]. Brodman and Johnson define a "small" organization as fewer than 50 developers and a "small" project as fewer than 20 developers [5]. The European Committee (EC) subdivides small organizations as eXtra eXtra Small Organizations (XXS) that have 1 or 2 employees, eXtra Small Organizations (XS) that have 3-15 employees and Small Organizations (S) with 16-50 employees.

3. SPI Challenges faced by Small Organizations

Romana, Ivan and Jozsef list the following as the SPI challenges faced by Small Organizations [3].

a) High Individual Dependence – In Small organizations, owing to the small number of employees, individuals appointed to develop software for a particular problem domain become experts and the success of the project largely rests on the competency of that individual.

b) Overloaded Persons – Owing to the small number of employees, individuals are vested with multiple tasks requiring different skill sets and expertise leading to a sub-optimal performance of some activities.

c) Human Factors – When the growth of a small software organization with 1 or 2 persons to one having 10-15 employees is not supported by establishment of appropriate organizational structures, problems may creep in.

d) Small Number of Projects – Crisis results when small organizations attempt to start new projects to support existing ones.

e) Importance of Customer Communication – In Software Development undertaken by small organizations for known customers, communication with customers becomes intensive.

f) Funding Constraints – Small organizations can seldom bear the additional cost incurred by performing SPI, that requires appointment of new personnel.

4. Dynamic CMM For Small Organizations

Laryd and Orci propose a dynamic CMM for small organizations [1]. They attribute the large number of roles with responsibilities proposed in the CMM as the main reason for the need of a down scaled model suited for small organizations. They propose 3 models for the 3 types of small organizations – XXS, XS and S and restrict the model for Level 2.

4.1 XXS

In the case of a single employee, the person is both the manager in the role SM and a Software Engineer SE. In caseof 2 employees, one is both SM and SE,and the other is SE.

Because generally in such organizations subcontracting is not possible, the SSM KPA becomes irrelevant. The STG group is responsible for testing, verification and validation and because an internal STG is not realistic for a 2-person company, an external STG's services are utilized. Again, because an independent SQA is not feasible, Customer SQA (CSQA) available at the customer site may be used.

4.2 XS

In a XS Organization having 3-15 employees, there is a necessity for a Project Manager PM, and a Marketing & Sales(MS) role. There is a need for an internal STG. SCM for each project can be performed by SE and the Senior Manager SM can also work as SE.

4.3 S

Here the number of employees is 16-50. A Documentation Support Group (DSG) is needed due to the importance of documentation support. Software Subcontracting becomes relevant and there is a need for Software Subcontract Manager. SM may or may not take the MS role. Till subcontracting becomes extensively SM can play the role of SSM. There should be an independent internal SQA and a person in SCM role cannot take the role of SE due to the extent of Configuration Management required.

5. Software CMM for Small Organizations

Paulik suggests developing a mapping between CMM terminology and language used by organizations [2]. Organizational Terminology for roles such as PM should be specified. He does not discount the possibility of a person filling multiple roles. After the resolution of terminology issues consideration can be given to the context dependency. For example, SSM may be irrelevant when there is no subcontracting. But peer reviews are essential for any project. Even though small projects may not need an SCM group or a Change Control Board, change control is required. Even if there may not be an internal STG group, testing is required.

Paulik also highlights the importance of Senior Management Sponsorship for SPI in small organizations. He cautions that ignorance of this may lead to islands of excellence. Even though small organizations may not have full time SEPG staff, the responsibility for SEPG must be explicitly assigned and monitored. Paulik also envisages a staged approach where the "as is" processes are considered first before "should be" processes. A top[down approach that everyone follow the "should be" process is likely to lead to chaos. Paulik also highlights the importance of process documentation. Such documents need not be complex or lengthy.

Scope should be given for tailoring processes to a project as different projects may have different needs. Deployment of documented processes can be accomplished via training. Alternative training mechanisms need to be considered. Management training is also required as technically competent people who are promoted to managerial positions may not possess all the required management skills.

[6] identifies six keys for SPI in small organizations.

- Senior Management Support
- Adequate Staffing
- Applying Project Management principles to Process Improvement
- Integration with ISO 9001
- Assistance from Process Management Consultants
- Focus on providing value to the business

Brodman and Johnson have developed a tailored version of CMM for Small Busineeses, Organizations and Projects [7,8,9]. According to Paulik, most changes are:

- Clarification of existing Practices
- Exageration of the obvious
- Introduction of alternative Practices
- Alignment of Practices with Small Business/ Small Organization/ Small Project Structure and resources

Paulik cautions that focusing on achieving a desired level of maturity without understanding the underlying importance of practices is likely to yield unexpected results. He suggests that Maturity Levels are measures and not goals of improvement.

6. Conclusion and Future Work

The paper highlighted the challenges faced by small and medium sized organizations with regards to SPI and presented2solutions proposed in the literature. Considering the increasing number of such small and medium sized software organizations, there is an imperative necessity to tackle the SPI challenges faced by such organizations.

As a part of future work, the authors plan to conduct case studies and surveys to analyze the impact of the various alternatives proposed in the literature for SPI in Small and Medium Sized Software Organizations.

7. References

[1] Orci, Terttu, Laryd, Atrid, “Dynamic CMM for Small Organizations”, Retrieved from: uml.org.cn/cmm/pdf/1116/laryd00dynamic.pdf

[2] Paulk, Mark C., "Using the Software CMM in Small Organizations" (1998). Retrieved from: repository.cmu.edu/cgi/viewcontent.cgi?article=1010&context=isr

[3] Horvat, Romana Vajde, Rozman, Ivan and Gyorkos, Jozsef, “Managing the Complexity of SPI in Small Companies”, Retrieved from: negro.iing.mxl.uabc.mx/~bflores/archivos/spismall.pdf

[4] Rita Hadden, “How Scalable are CMM Key Practices?” Crosstalk: The Journal of Defense Software Engineering, Vol. 11, No. 4, April 1998, pp. 18-20, 23.

[5] Donna L. Johnson and Judith G. Brodman, “Applying the CMM to Small Organizations and Small Projects,” Proceedings of the 1998 Software Engineering Process Group Conference, Chicago, IL, 9-12 March 1998.

[6] John J. Abbott, “Software Process Improvement in a Small Commercial Software Company,” , Proceedings of the 1997 Software Engineering Process Group Conference, San Jose, CA, 17-20 March 1997.

[7] J.G. Brodman and D.L. Johnson, "What Small Businesses and Small Organizations Say About the CMM," Proceedings of the 16th International Conference on Software Engineering, IEEE Computer Society Press, Sorrento, Italy, 16-21 May 1994, pp. 331-340.

[8] Donna L. Johnson and Judith G. Brodman, The LOGOS Tailored Version of the CMM for Small Businesses, Small Organizations, and Small Projects, Version 1.0, August 1996.

[9] Donna L. Johnson and Judith G. Brodman, "Tailoring the CMM for Small Businesses, Small Organizations, and Small Projects," Software Process Newsletter, IEEE Computer Society Technical Council on Software Engineering, No. 8, Winter 1997, p. 1-6.

[10] James Herbsleb, David Zubrow, Dennis Goldenson, Will Hayes, and Mark Paulk, "Software Quality and the Capability Maturity Model,” Communications of the ACM, Vol. 40, No. 6, June 1997, pp. 30-40.

[11] Patricia K. Lawlis, Robert M. Flowe, and James B. Thordahl, "A Correlational Study of the CMM and Software Development Performance," Crosstalk: The Journal of Defense Software Engineering, Vol. 8, No. 9, September 1995, pp. 21-25. Reprinted in Software Process Newsletter, IEEE Computer Society Technical Council on Software Engineering, No. 7, Fall 1996, pp. 1-5.

[12] Bradford K. Clark, "The Effects of Software Process Maturity on Software Development Effort," PhD Dissertation, Computer Science Department, University of Southern California, August 1997.