

An Enhanced Action Research Approach for Managing Risks in Software Process Improvement

Faiza Ayub Syed

Department of Software Engineering, University of Engineering & Technology, Taxila, Pakistan
Email: faizaayubsyed@yahoo.com

Kokub Sultan

Department of Software Engineering, University of Engineering & Technology, Taxila, Pakistan
Email: kokub.sultan@yahoo.com

Ali Javed

Department of Software Engineering, University of Engineering & Technology, Taxila, Pakistan
Email: ali.javed@uettaxila.edu.pk

Abstract—Managing risks in Software Process Improvement (SPI) is a key point of software success. A software risk is considered as an essential characteristic of software development process which if ignored will increase the chance of project failure. For this purpose different risk management approaches are developed. These approaches lead to the identification, assessment and control of risk occurrence in software projects. Collaborative Practice Research (CPR) is one of the action research approaches for managing risk in SPI. In this approach the focus is on gathering information regarding SPI and acknowledging risk management in process development by developing risk assessment strategies and models. The main challenge of this action research approach is to validate the developed risk approach. This paper has a critical review on the existing research approach i.e. CPR. It also provides an enhanced form of CPR which modifies the current CPR approach by including a risk validation activity.

Index Terms— Software Process Improvement (SPI), Collaborative Practice Research (CPR), Enhanced Collaborative Practice Research (ECPR).

I. INTRODUCTION

From last few years organizations encircling software industry had flourished well due to competency towards Software Process Improvement (SPI). The reason behind this is that organizations following SPI programs during their project development processes have consistent and good quality products as compared to organizations that are not following SPI.

SPI is a rapid process that helps in improving organizational performance in building quality based products for their clients. This was proposed by McFeeley [1]. He emphasized that organizations should follow some sort of strategic plan or systematic approach for SPI. Different methodologies or models

can be followed for initiating SPI. These models provide guidelines at various levels of process development. SPI plays an essential role in Software Development Life Cycle (SDLC). It helps in increasing product quality and reliability by reducing the risk factors that may cause serious issues related to Return on Investment (ROI) thus gaining client's trust [2]. The success of software project lies in choosing an accurate and efficient software process that helps in risk reduction and boosting the factor of software project success [3]. Organizations should modify their formal development procedures for improving software processes to accelerate the change. For innovative projects with modified development procedures stage gate procedures are followed. Such a procedure helps in optimizing the trade-offs between quality and cost thus improving the software process improvement program [4]. One of the SPI models which focus on continuous process improvement is Capability Maturity Model Integrated (CMMI) which helps in increasing the organization quality and productivity level [5]. Most organizations follow the six sigma framework for process improvement since it helps in the identification of problems in software processes and projects and providing optimized solutions for the identified problems. Thus, an organization can achieve its goals by improving the development process qualitatively and quantitatively [6]. SPI reduces the cost escalation by identifying risks and defects during the project phases and mitigating them through modified formal controlled procedures [7]. For managing risks in SPI, CMMI integrates risk management practices in most of its process areas like Project Management. These risk management process areas helps in identifying, anticipating and mitigating risks, thus, improving software process [8].

In this paper we proposed the inclusion of validation activity for Collaborative Practice Research (CPR), an action research approach. This approach is based on

gathering information regarding SPI and acknowledgement of risk management in process development by developing risk assessment strategies and models. One of the major drawbacks of model proposed by Iverson and Mathiassen [9] is lack of risk validation. They developed a risk framework and design their processes accordingly. To overcome this drawback a validation approach is needed for mitigating risks severity.

This paper consists of six sections. The first section is introduction. The second is about the related work linked with SPI and risk management. The third section is on existing research technique CPR for risk management in SPI. The fourth section is proposed research technique ECPR for managing and validating risks. The fifth section is based on CPR and ECPR comparison. Sixth section is the conclusion.

II. LITERATURE REVIEW

Risk management in software teams is essential factor of Software Process Improvement (SPI). This provides us with an approach that helps in identifying and understanding risky regions in projects and strategies that provides solution to these risks. Also it helps in understanding risk management in large enterprises. It is necessary to recognize and examine the potential causes that oppose to put SPI into practice for projects running in organizations.

Mohd and Rodina [10] presented a survey report of 8 Malaysian organizations who have introduced SPI in their projects. According to it the factors affecting SPI are lack of professional expertise, non-technical organizational staff, and lack of know how about SPI activities. Since the survey is based on a set of opinion poll in order to collect the necessary information; the report highlights the limitations by suggesting that the survey is not very useful for organizations who are not participating in the survey. On the other hand the results of the survey are reliable and applicable to organizations who have participated in it. They suggested that more big picture and understanding of SPI in organizational projects can be achieved if this survey is done on large scale covering large number of organizations.

Biffala and Dengerb [11] introduce the ideas for Quality Assurance Tradeoff Analysis Method (QATAM), a procedure that helps in evaluating QA strategies and their tradeoffs from Software Process Improvement (SPI) perspective. QATAM draws and approach to acquire stakeholder suggestions and risks about the project to operationalize quality requirements in scenarios by identifying and mitigating relevant project risks. It is basically a continuing research project which aims at providing support for QA activities in Small and Medium Enterprises (SME's). It provides a framework for identifying and evaluating QA strategies that help sustains quality managers and project leaders. It identifies and estimate possible benefits and risks regarding software engineering and QA policies. Repeatable scenario-based estimation of QA procedures

and exchange between these procedures in a project framework allows measurement of appropriate QA strategy with in the project. Moreover, applying QATAM will provide support in recognizing potential barriers in pragmatic proofs.

Suwanya and Kurutach [12] provide an analysis and comparison of different characteristics for software improvement methods used in Thailand. For this purpose a survey for 1200 software manufacturing companies were conducted. It was found from the results of survey that only 11.8 percent companies have adopted Software Process Improvement (SPI) in software development. Since the software production is time consuming and need for human resource is a major factor so it is difficult for small enterprises to cope with SPI because of low budget and less resources. After the survey a comparative analysis is made on different software standards that can be used as a baseline by an organization for SPI. By implementing SPI for software development it is possible for an organization to achieve quality software products thus increasing robustness factor.

According to Taylor and McGraw [13] one of the major challenges that large organizations are facing is the improvement in security of software. This can be achieved through careful planning and by applying best practices. For this purpose a specific plan is build that insists on prioritizing the requirements that are essential, hiring of professional employees, training of employees, and developing a continuous improvement methodology. This would result in major organizational change resulting in risk reduction.

Miler and Gorski [14] proposed a rapid risk assessment approach for Software Process Improvement (SPI). The series of steps involved in this are: (1) modeling a process (2) mentioning possible risks (3) finding out the root causes of risks and deficiencies (4) improving the modeled process (5) finally updating the modeled process. This approach made risk detection easier thus causing a good impact on SPI.

Iverson and Mathiassen [9] suggested a framework that is based on Collaborative Practice Research (CPR). They studied the Danish SPI research program that was based on CPR. It provides risk strategies and framework for managing risks in SPI. They suggested that there should be a Software Engineering Process Group (SEPG) for SPI project organization as shown in Fig. 1. SEPG along with steering group operate at organizational level and SPI teams (Project Management, Configuration Management and Quality Control) work at project level. These will bring up-to date risk item list for organization.

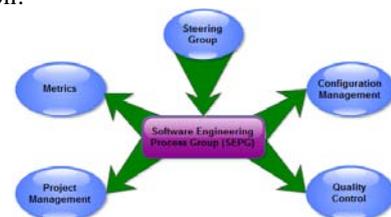


Figure1. SEPG and SPI Organization

Managing risks in software development is a key point of software success. A software risk is considered as an essential characteristic of software development process which if ignored will increase the chance of project failure (NIST) [15]. For this purpose different risk management approaches are developed. These approaches lead to the identification, assessment and control of risk occurrence in software projects. Iverson et al differentiates four different risk management approaches listed in Table 1[9].

Table 1: Four Types of Approaches to Software Risk Management

Type of Approach	Characteristics
Risk list	List of risk items arranged according to severity level.
Risk action list	List of risk items arranged according to severity level along with possible resolutions.
Risk strategy model	A contingency model that recapitulate risk sources and ways of dealing with the risk.
Risk strategy analysis	A stepwise approach for detailed understanding of risks.

These approaches along with their strengths and weaknesses are shown in Fig 2. 1. Risk Items: used for detecting risk occurrences 2. Risk Resolution: helps in identification of actions for mitigating risk 3. Heuristics: establish link between risks and possible actions [9].



Figure2. Strengths and weaknesses of Risk Management Approaches

CPR is done to gather information related to SPI and understanding of risk management in order to fulfill the requirements of SPI teams. It also defines the procedures that helps identify roles, documentation and policies related to research process.

III. EXISTING RESEARCH TECHNIQUE

During 1980's and 1990's Scandinavian information research systems come up with a new research practice CPR that owns the following characteristics: 1. To understand and improve the professional practices that exists within organizations participating in SPI program. 2. Activities are conducted by practitioners and researchers in teamwork. 3. Additional approaches like case studies and field experiments are conducted. 4. Finally leading to the selection of those project areas which need improvement.

The research was developed from Danish program for SPI based on CPR as a collaborative effort of three years. The research team was organized as follows: four researchers three of them were authors, four SPI team members, SEPG, and a research group that will guide throughout the research process [9].

A. Research Process

The research process was carried out in an iterative manner focusing on discovering problems and changes that take place during the process. The process involved repetitive set of activities.

B. Research Criteria

Lack of researcher interest, misguidance and lack of knowledge and discipline unfortunately lead to number of pitfalls. For this purpose a research criteria in the form of questions was designed for CPR process. Questions included in the criteria were about: 1. Roles: What will be the roles of practitioners and researchers? 2. Documentation: what data will be gathered to satisfy the research objective and how it will be gathered? 3. Control: how to establish the relationship between researchers and clients? 4. Usefulness: how to determine the usefulness of the process? 5. Theory: how the existing frameworks will be used and how the results of the study will be mapped to those frameworks? 6. Transfer: Under what conditions the result can be adapted in some other problem situation. Following the above criteria, clarify the roles of researchers and practitioners; describe the data collection approach; establish client researcher relationship through contracts and agreements; establish usefulness of results; relating the results to already existing frameworks and transferring the results to other situations to keep the research process general rather than being specific.

C. Research Practice

The research was practiced on Danske Bank's IT department which was basically a system development organization. Danske Bank joined the CPR program and founded SEPG for carrying out SPI activities. The research team included a project manager, two information system managers and four researchers. Initially the SEPG assessed the software process maturity of IT department projects. The assessment report highlighted seven improvement areas, some of which were successfully addressed by SPI teams. Later,

CPR was performed for appropriate risk management.

CPR based process consists of three phases as shown in Fig.3 [9]. These phases are: 1. Initiating. 2. Iterating. 3. Closing. Further these three phases contains different activities.

C. 1. Initiating

During the initiating phase a team of practitioners and researchers held a workshop on SPI to identify the major sources of risks and possible ways of resolving those risks. After the workshop the team presented a report to research authors, which contain information on risky items in SPI and a standard risk management approach for software development. Following the report presented, the whole team conduct brainstorming sessions to identify the pertinent risky items for SPI based on the knowledge and literature for the required area of concern. During the second brainstorming session, possible ways of resolving and addressing the risks identified in first session were produced. The research authors after studying the literature identified four different risk approaches (see Table 1). They decided to adopt risk strategy analysis approach because they need an approach which would help them in gaining a tactical understanding of the SPI project. Risk strategy analysis being a stepwise approach helped the SPI team to better understand the risk management plan.

C. 2. Iterating

After the selection of risk approach the researchers move towards the iterating phase. The goal of iteration phase was to develop a risk framework for the problem situation. In iterating phase risk assessment of four different improvement projects was done. In iteration phase the action research process underwent four iterations. Risk analysis was carried out for quality assurance, project management, metrics program and diffusion improvement projects in first, second, third and fourth iterations respectively. A brief overview of these iterations is given below.

C. 2.1 First Iteration

During first iteration the research authors integrated the brainstorming results and formulated a prototype of the risk approach. The practitioners responsible for quality then perform a risk analysis for improving the quality assurance of the process. During the risk analysis they analyze that some risk items needed to be reformulated.

C. 2.2 Second Iteration

The second iteration started with risk reformulation. The process is applied for risk assessment of project management improvement project. The synthesized framework was reviewed again for risk items and associated actions by carrying out a detailed risk analysis and adding explicit risk items as the process goes on, thus, producing a revised risk framework.

C. 2.3 Third Iteration

In third iteration the risk approach was applied on the required area of concern. In this session a new scheme of documenting the risk process was introduced. After successful application of approach, the risk analysis was performed on the concerned project which in this case was an organizational metrics program. The risk analysis and documentation helped the team in better understanding of risks and actions.

C. 2.4 Fourth Iteration

The same risk approach was applied again in fourth iteration without any reformulation; on an improvement project responsible for dissemination of software practices. Although the results of risk analysis were useful, however, the team emphasized that risk items and resolution actions should be supplemented with more details for best results.

C. 3 Closing

After applying the risk approach assess the risk occurrence by conducting field experimentation and evaluating results. If results are not suitable, then move back to the first iteration. If evaluation results are fine then close the action part of the process. Finally assess the usefulness of risk process by considering the projects on which it was applied and extract the research results.

D. Research Results

The research resulted in formulation of two approaches. First approach was to manage risks in SPI teams whereas second approach tailored risk management to information systems and software engineering perspectives. Both approaches were used to address the problem situation by providing a framework that would be used for understanding the problem domain and a methodology for solving the problem.

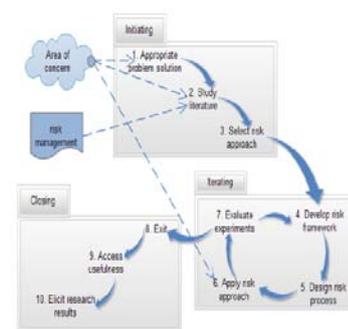


Figure3. CPR Based Process Approach

IV. PROPOSED RESEARCH TECHNIQUE

The framework for CPR based process proposed by Iverson and Mathiassen have a downside since it lack risk validation. To overcome this problem we proposed an ECPR approach that enhances the CPR framework by including the activity of risk validation in the

iterating phase of CPR framework as shown in Fig. 4. The inclusion of risk validation activity helps in validating the risk framework. The major phases of ECPR were similar to CPR approach i.e. Initiating, Iterating and Closing.

A. Initiating

During initiating phase in ECPR, appropriate risk management approach is selected for the concerned improvement project.

B. Iterating

In iterating phase, based on study a risk framework is designed by focusing on the risky points that were identified during the initiating phase. After designing the framework next step was to validate that risk framework. For this purpose a risk framework prototype is developed. Validation can be done on the prototype by inspecting the related documentation, checking the system by examining the operational and technical aspects. Make sure that all system specification should be inspected to get better results. Finally evaluate the prototype through field experiments. If the evaluation results cover up a significant number of risks then we can be sure that the prototype for risk framework can be used in risk design process.

Risk validation was performed on risk framework prototypes developed for quality assurance and project management improvement projects. The methodology we have followed for risk validation in above two improvement projects is Stage-Gate [16]. Stage-Gate or Phase-Gate methodology also referred to as a Phase-Gate process or sometimes Gate-Review is a method in which the project is divided into stages or phases separated by gates. Phases include scope definition, feasibility study, development, testing and validation, and launching phase. Each stage/phase consists of set of activities. These stages are managed by single or cross functional teams. At the end of each stage there is a decision point called gate. Gate reviews are done to review the progress at each stage. They include checklists, forms and guidelines to ensure that critical steps are not omitted at any stage. The steering group conducts gate reviews and decides whether to proceed to next stage with original plan or revised plan.

Following the above methodology we divide our improvement projects into different stages. Since our goal was to analyze and validate the risks; so after defining the stages, gate reviews were performed for reviewing the projects risks. The identified risks were of budget, time/schedule, resources and quality. Validation policy for identified risks is given below.

1. Budget and Schedule Risk: Validate the schedule and budget risks by matching the current estimates with the final spending.
2. Resource Risk: Validate the resources required to complete the project. Since the type of resources needed in project change from phase to phase so the availability of resources should be validated.

3. Quality Risk: Validate the quality risks by holding a quality event. Quality events are used for reviewing the overall quality of the deliverables. A quality event can be an expert review, multi-person review, formal inspection or process review.
 - a) Expert Review: In this type of review a senior analyst reviews the data model of the process where major focus is on the accuracy of content.
 - b) Multi-person Review: An independent review which involves several team members and stakeholders. It allows the members to share their viewpoints. Sometimes it is difficult to handle conflicting viewpoints but still it appears as a best quality event in case where agreement between different stakeholders has to be established.
 - c) Formal Inspection: This quality event helps in capturing the statistics of expected risks.
 - d) Process Review: This type of review ensures that all actions are taken in place and proper change control procedures are followed for modifying or improving an existing project.

Once completed with the risk validation process, a formal approval is made by the steering group to move to the next stage in which we will be applying the risk approach to the concerned project. Based on the results of validation process, a validated risk process is designed before moving on to next stage. After designing and re-approval from the steering group the process is applied on the concerned improvement project. The project is then rated by several teams that have participated in the improvement program. The evaluation results based on ratings show signs of positivity for the improved project.

C. Closing

After ending up with all stages of the improvement project, close the action part of ECPR. Finally assess the usefulness of approach applied for improvement purpose and extract the research results.

D. Research Results

The research resulted in the development of an approach that would not only be used for managing risks in SPI but also for validating the risk approaches developed for SPI.

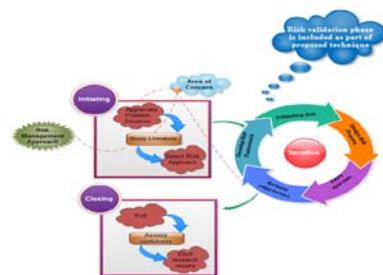


Figure4. ECPR Based Process Approach

V. CONCLUSION

The CPR based action research approach combines information from SPI as well as risk management for responding to the practical needs of SPI teams. CPR not only aims to provide a research approach but it also disseminates information for practicing the research approach. It also provides risk approaches to manage risks in SPI. Managing risks in SPI is not very expensive. Also it helps in reducing the likelihood of failures in software improvement program.

The objective of this research is to validate the identified risks. In order to achieve our objective, we design an ECPR approach that enhances the existing CPR approach by including the validation activity in it.

In ECPR, we first selected a risk approach for our improvement project. Then we performed a risk analysis. The result of risk analysis led to the development of risk framework prototype. After the development of risk framework prototype, we performed risk validation over it. For this, we divided our improvement project into manageable phases. For reviewing the project risks we used Phase-Gate model. This model helped us in validating the project risks. After validation we designed a revised risk framework. Then we applied this framework on our improvement project.

We believe that ECPR would be useful in validating the risk approaches and helps in improving the quality and cost of projects, thus, leading to SPI.

REFERENCES

- [1] Bob McFeeley, "IDEALSM: A User's Guide for Software Process Improvement", February 1996.
- [2] Rini van Solingen, "Measuring the ROI of Software Process Improvement", IEEE, 2004. DOI:10.1109/MS.2004.1293070.
- [3] Xu Ru-zhi, "Optimizing software process based on risk assessment and control", IEEE, September 2005. DOI:10.1109/CIT.2005.151.
- [4] John E. Etlie, Jorg M. Elsenbach, "Modified Stage-Gate® Regimes in New Product Development", Journal of Project Innovation Management, 2006. DOI:10.1111/j.1540-5885.2006.00230.x.
- [5] Zhedan Pan, Hyuncheol Park, Jongmoon Baik, Hojin Choi, "A Six Sigma Framework for Software Process Improvements and its Implementation", 14th Asia-Pacific Software Engineering Conference, IEEE, 2007. DOI:10.1109/ASPEC.2007.43.
- [6] Yonghui CAO, Xin Xiang, "Software Process Improvement Framework Based on CMMI Continuous Model Using QFD" in proceedings of International Journal of Computer Science Issues, 2013.
- [7] Ray C. Williams, "The CMMI RSKM Process Area as a Risk Management Standard" in proceedings of Sixteenth Annual International Symposium of the International Council On Systems Engineering, July 2006.
- [8] Geir Sigurour Jonsson, "A Case Study into the Effects of Software Process Improvement on Product Quality", June 2004.
- [9] Jakob H. Iversen, Lars Mathiassen, "Managing Risk in Software Process Improvement: An Action Research Approach", September 2004.
- [10] Mohd Hairul Nizam Md. Nasir, Rodina Ahmad, Noor Hafizah Hassan, "Resistance Factors in the Implementation of Software Process Improvement Project", IEEE, ITSIM 2008. DOI:10.1109/ITSIM.2008.4631933.
- [11] Stefan Bifflla, Christian Dengerb, Frank Elberzhagerb, Dietmar Winklera, "Quality Assurance Tradeoff Analysis Method (QATAM), An Empirical Quality Assurance Planning and Evaluation Framework", 2008.
- [12] Suphak Suwanya, Werasak Kurutach, "An Analysis of Software Process Improvement for Sustainable Development in Thailand", IEEE, 2008. DOI:10.1109/CIT.2008.4594764.
- [13] Dan Taylor, Gary McGraw, "Adopting a Software Security Improvement Program", May/June 2005. DOI:10.1109/MSP.2005.60.
- [14] Jakub Miler1, Janusz Gorski, "Risk-driven Software Process Improvement - a Case Study", EuroSPI, 2004.
- [15] Gary Stoneburner, Alice Goguen, Alexis Feringa, "Risk Management Guide for Information Technology Systems", NIST Special Publication 800-30, July 2002.
- [16] Dr. Robert Cooper, Dr. Scott Edgett, "Stage-Gate® - Your Roadmap for New Product Development", <http://www.prod-dev.com/stage-gate.php>, 1996-2013.

Engr. Faiza Ayub Syed is a MS scholar as well as Research Associate in the Department of Software Engineering at University of Engineering and Technology Taxila, Pakistan. She graduated from University of Engineering and Technology Taxila in Software Engineering in July 2011. Her areas of interest are Software Design and Architecture, Software Requirement Engineering, Computer Communication and Networks, Object Oriented Programming, Web Application Development and Desktop Application Development.

Engr. Kokub Sultan is a MS scholar as well as Research Associate in the Department of Software Engineering at University of Engineering and Technology Taxila, Pakistan. She graduated from

Fatima Jinnah Women University Rawalpindi, Pakistan in Software Engineering in July 2011. Her areas of interest are Computer Vision, Data Mining, Software Quality Assurance, Computer Graphics, Object Oriented Programming, Web Application Development and Desktop Application Development.



Engr. Ali Javed is serving as an Assistant Professor in the Department of Software Engineering at UET Taxila, Pakistan. He is also a PhD Scholar in Computer Engineering Department at UET Taxila, Pakistan. He has received his MS degree in

Computer Engineering from UET Taxila, Pakistan in February, 2010. He received Chancellor's Gold Medal in MS Computer Engineering degree and became the first MS student in \ the history of UET Taxila \ to be awarded Chancellor's Gold Medal. He has received B.Sc. degree in Software Engineering from UET Taxila Pakistan in September 2007. He got 3rd position in Software Batch-2003F in BS Software Engineering degree. His areas of interest are Digital Image Processing, Computer vision, Video Summarization, Mobile Application Development, Software Requirements Engineering, Software Quality Assurance and Software testing.