

# A Study of Software Process Improvement Approaches Proposed for SMEs

Bashiram Basumatary<sup>1</sup>, Nomi Baruah<sup>2</sup> and Himanshu Borgohain<sup>3</sup>

<sup>1,2,3</sup>Dibrugarh University, Department of Computer Science and Engineering

**Abstract:** Today the improvement of software development processes is certainly one of the top priorities of the IT industry. The paper was developed to support those software developers of software enterprises and software researchers who are interested in the software process improvement of small scale and medium scale software enterprises.

**Keywords:** CMMI, K-Model, PSP, SQIMSO.

## 1. Introduction

A brief description of some better known approaches has been described in this paper. In order to describe the variety of software process improvement approaches which has been proposed for SMEs, brief summaries of the main features of software approaches and its software processes are presented in context, developmentally as well as historically.

## 2. Overview of Software Process Improvement Approaches

### 2.1 CMMI

The Carnegie Mellon University's Software Engineering Institute (SEI) has proposed a wide range of Capability Maturity Model (CMM). They are: software engineering (SW-CMM), systems engineering (SE-CMM), software acquisition (SA-CMM), human resources management (P-CMM), integrated product and process development (IPPD-CMM) and so on [1]. The models have helped the software organizations to improve their software processes and to produce quality product. However, the coexistence of the models showed the defects gradually. Therefore, a new model is proposed Capability Maturity Model Integration (CMMI). The purpose of this integration is to reduce the multi-disciplinary model-based process improvement costs [2]. It has been observed that, when CMMI is viewed holistically, CMMI's ultimate goal (i.e. continuous process improvement) is to cause an organization to become less wasteful, leaner and more in touch with actual development progress. The process areas along with capability levels are given below:

**Table 1:** Software Process Areas of CMMI [3]

Maturity Levels	Key Process Areas
Level 5-Optimizing	Organizational Innovation and Deployment
	Causal Analysis and Resolution
Level 4-Quantitatively Managed	Organizational Process Performance
	Quantitative Project Management
Level 3-Defined	Requirements Development
	Technical Solution
	Product Integration
	Verification
	Validation
	Organizational Process Focus
	Organizational Process Definition
	Organizational Training
	Integrated Project Management
	Integrated Supplier Management
	Risk Management
	Decision Analysis and Resolution
Level 2-Managed	Requirements Management
	Project Planning
	Project Monitoring and Control
	Supplier Agreement Management
	Measurement and Analysis
	Process and Product Quality Assurance
Level 1-Performed	Configuration Management

### 2.2 PSP

PSP stands for Personal Software Process. It is a technique to improve the predictability, quality and productivity of the software engineers in their work. The main objective of PSP is to help the software engineers how a defined and measured process can help them to improve them to improve their personal performance irrespective of the engineering methods [4]. The PSP is similar to CMM except that it focuses on the personal process. The PSP involves the following personal costs like the time required by software engineers to learn and use it, the emotional cost of maintaining the needed discipline and the potential risk to their ego [5]. A typical PSP course uses ten software development exercises, a structured sequence of defined processes, and five data analysis exercises to demonstrate the process. The process areas of PSP along with the capability levels are

**Table 2:** The Process Areas of PSP [4]

Maturity Levels	Key Process Areas
Level 5-Optimizing	Process Change Management
	Technology Change Management
	Defect Prevention
Level 4-Managed	Quality Management
	Quantitative Process Management
Level 3-Defined	Peer Reviews
	Software Product Engineering
	Integrated Software Management
	Software Process Definition
	Software Process Focus
Level 2-Repeatable	Software Project Tracking and Oversight
	Software Project Planning
Level 1-Initial	.....

**2.3 BOOTSTRAP**

BOOTSTRAP is a European Software Process Assessment and Improvement Methodology which was initially developed in an ESPRIT project starting from lean and kaizen philosophy. The first version of BOOTSTRAP methodology(version 1.0) was developed in the BOOTSTRAP project taking the CMM(version 1.0) as the basic reference(capability and maturity levels and process definitions) and extending it with the features of ISO 9000 quality standards and the European Space Agency lifecycle model [6].Unlike the CMM,BOOTSTRAP does not assume strict adherence to a distinct key practice model and allows the use of alternative approaches[7].The main characteristics of the method are the reference framework, the assessment procedure, the structure of questionnaires, and the rating and scoring mechanisms employed [8].

**Table 3:** Process Areas in BOOTSTRAP Architecture [6]

Process Category	Process Areas	
Organization	Management Practices	
	Quality Management	
	Resource Management	
Methodology	Product Engineering	Requirements Analysis
		Definition
		Detailed Design and Implementation
		Testing and Integration
		Acceptance & Transfer
		Operational Support and Maintenance
		Process Engineering
	Process Control	
	Life Cycle Independent (Support) Functions	Process Measurement
		Project Management
		Quality Assurance
	Technology	Risk Management
		Technology Management
		Product Engineering Technology

**2.4 SQIMSO**

SQIMSO stands for Software Quality Model for Small Organizations. The SQIMSO is motivated by three main critical issues.First, every improvement program should be wide enough to include three main factors which are process quality, product quality and human resources management.Second, any process quality model should answer the question “How to do?” . Third, any suggested quality model should be practical enough to be implemented by small software organizations for saving costs and time without decreasing the quality turnover level [9].

**Table 4:** SQIMSO Model Process Structure [9]

Process Category	Process Title
Customer-Supplier	Acquisition
	Supplier Selection
	Requirement Elicitation
Engineering	Requirement Analysis & Specification
	Software Planning
	Software Design
	Software Implementation
	Software Testing and Integration
	Software Maintenance
	Documentation
Support	Configuration Management
	Quality Assurance
	Joint Review
	Method Selection
Management	Project Management
	Improvement Program
	Human Resources Management

**2.5 K-Model**

The certification level of K-model is categorized into three levels. They are Initial level (level 1), Good level (level 2) and Very good level (level 3)[10].The certification level is differently applied to valuation factors. Each certification degree acts as an indicator representing the degree of activity capability level related with software development project performance. The initial level is the necessary level of improving the process capability in the situation of the performance level of special project, or quality cost, the appointed date of delivery. In the good level, the process is the capability level to successfully perform the project. In the very good level, the performance of the project is of consistent quality level [10].

**Table 5:** K-Model Processes (Good Level) [10]

Group	Assessment Process
Project Management	Project Planning
	Project Control
	Partnership Management
Develop	Requirement Management
	Analysis
	Design
Support	Implementation
	Test
	Quality Assurance
	Configuration Management
	Measurement and Analysis

**2.6. MPS Model**

The MPS model was proposed to improve the quality of Brazilian software processes and products through the development and dissemination of Brazilian software

process model. The model is based on software engineering best practices and is aligned to Brazilian software industry context. The MPS model is constituted of the MPS Reference Model, the MPS Assessment Method and the MPS Business Model [11].

**Table 6:** MPS Processes [11]

Levels MPS	MPS Processes
G	Project Management Requirement Management
F	Measurement Acquisition Configuration Management
E	Quality Assurance Human Resources Management Process Establishment Process Assessment & Improvement Project Management(New Outcomes) Reuse Management
D	Requirements Development Product Design and Construction Product Integration Verification Validation
C	Decision Analysis and Resolution Risk Management Development for Reuse
B	Project Management(New Outcomes)
A	Causal Analysis and Resolution

### 3. Conclusion

It is hoped that the paper provide support for those researchers who are interested in software process improvement in small scale and medium scale enterprises as it gives references about the software processes which may be used to assist further research in software process improvement. It is also hoped that the paper provides information about the software processes used in small scale and medium scale enterprises. The software processes which are being described here can be used by software enterprises and can tailor them as per the software projects they undertake.

### References

[1] <http://www.tutorialsto.com/software/engineering/cmmi/small-scale-enterprises-to-implement-cmmi-softw-are-process-improvement-research-and-analysis.htm> l,accessed on 3<sup>rd</sup> February,2014

[2] Hillel Glazer, Jeff Dalton, David Anderson, Mike Konrad and Sandy Shrum, "CMMI® or Agile: Why Not Embrace Both", November 2008. TECHNICAL NOTE CMU/SEI-2008-TN-003.

[3] L.G. Jones, Software Process Improvement and Product Line Practice: CMMI and the Framework for Software Product Line Practice, US: SEI, Carnegie Mellon University, 2002.

[4] Watts S. Humphrey, "The Personal Software Process: Status and Trends", IEEE Software, vol.17, no.6, pp.71-75, Dec 2005.

[5] Mike Grasso, "The Personal Software Process", A Software Engineering Seminar, University of Maryland Baltimore County.

[6] Pasi Kuvaja, Software Process Capability and Maturity Determination. BOOTSTRAP methodology and its evolution, University of OULU, Finland.

[7] Zahran, Software Process Improvement: Practical Guidelines for Business Success. Reading, MA: Addison-Wesley, 1998.

[8] Stienen, H., Software Process Assessment and Improvement: Five Years of Experiences with BOOTSTRAP. In K. Emam & N. Madhavji (Eds.), Elements of Software Process Assessment and Improvement (pp. 57-75), Washington, DC: IEEE Computer Society Press, 1999.

[9] Rabih Zeineddine and Nashat Mansour, "SQIMSO: Quality Improvement for Small Software Organizations", Journal of Computer Science, vol.1, no.3, pp.316-322, 2005.

[10] Sun Myung Hwang, "Quality Metrics for Software Process Certification based on K-model", IEEE 24<sup>th</sup> International Conference on Advanced Information Networking and Applications Workshops, 2010.

[11] Gleison Santos, Mariano Montoni, Jucele Vasconcellos, Sávio Figueiredo, Reinaldo Cabral, Cristina Cerdeiral, Anne Elise Katsurayama, Peter Lupo, David Zanetti and Ana Regina Rocha, "Implementing Software Process Improvement Initiatives in Small and Medium-Size Enterprises in Brazil", Sixth International Conference on the Quality of Information and Communications Technology, 2007.

### AUTHOR



Bashiram Basumatary received the Bachelor of Technology degree in Computer Science & Engineering from Dibrugarh University in July, 2014. The work of this paper is from his BTech Project. His area of interest includes Software Process Re-engineering and Unified Modelling Language.



Nomi Baruah is Assistant Professor in Department of Computer Science & Engineering, Dibrugarh University. Her research area includes Software Process Reengineering, Agile Software Software Development, Software Reuse, Software Process Related Customization & Automization, Unified Modelling Language and Software Metrics.



Himanshu Borgohain received the Bachelor of Technology degree in Computer Science & Engineering from Dibrugarh University in July, 2014. The work of this paper is from his BTech Project. His area of interest includes Software Process Re-engineering and Unified Modelling Language.