Comparison of Software Process Improvement

*E.Geetha Rani[#1], k.Swarupa Rani[*2], D.Anusha[#3], j.k.m.gayatri[*4], k.l.Tejaswini[#5]*
#1Infornation Technology, Gudlavalleru Engineering College, Gudlavalleru, jntuk, INDIA

**2 & #3 PVPSIT, IT, Vijayawada, jntuk, INDIA*

*#4 & *5 Gudlavalleru Engineering College, IT, Gudlavalleru, jntuk, INDIA*

Abstract

Software Process Improvement programs are being integrated with the key processes practiced in different software organizations, who really want to improve their processes continuously in order to enhance their quality of the products. Software Process Improvement (SPI) encompasses a set of activities that will lead to a better software process, and as a consequence, higher quality software delivered in a desired time span .Software Process Improvement is an important activity which starts when an organization plans to enhance the capabilities of its ongoing processes. There are many Software Process Models those exist in software industry. There exist a set of key processes those are practiced and are being applied by different organizations to improve quality of these products. This paper summarizes important SPI models those comprise of effective set of practices which can enhance the maturity of software organizations.

Keywords- Software Process improvement models, Software engineering Institute, SPI

1. Introduction

The objectives of software process improvement is to set methods in order to improve the development process including project management, eliciting and managing requirements, decision making, measuring performance, planning the work, handling the risks and many more[6]. It's important that organization should analyze the organizational structure and process and identify the main reasons behind their failed and successful projects and take measures to make improvements wherever needed.

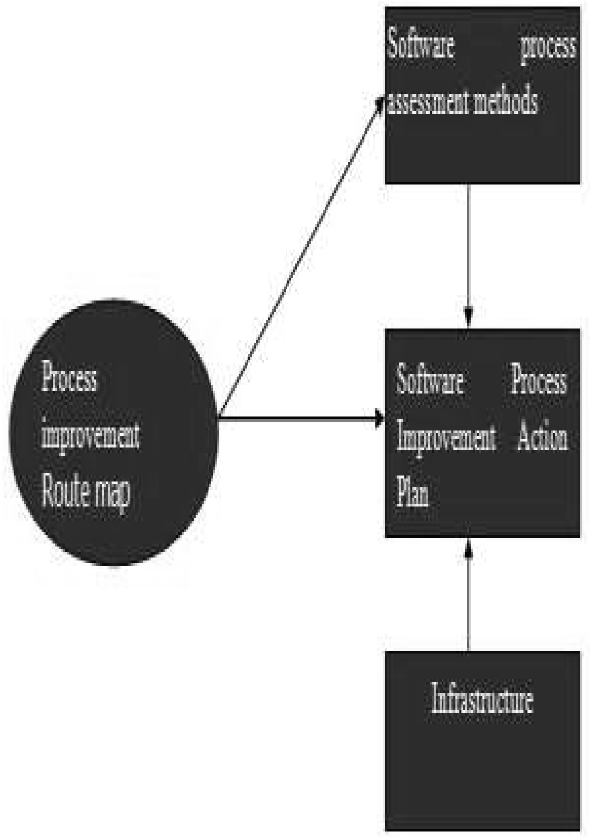
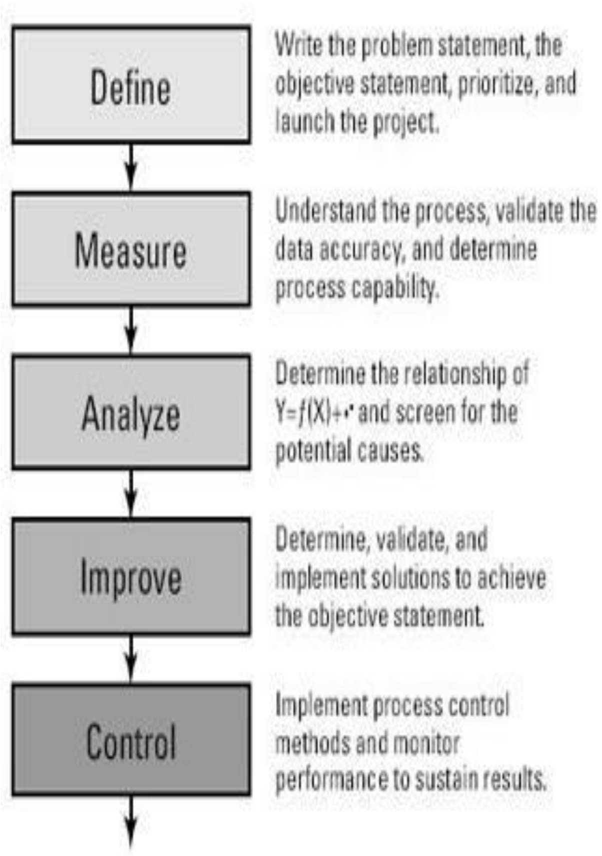

Software process assessment methods
Process improvement Route map
Software Process Improvement Action Plan
Infrastructure

Figure1: Software Process Improvement Framework

2. Six Sigma Six Sigma is a process improvement management framework to achieve bottom-line results, and customer's loyalty. In short the objective of Six Sigma is the implementation of a measurement based strategy that is focused on process improvement and variation reduction.

Figure 2: SIX SIGMA


Define
Write the problem statement, the objective statement, prioritize, and launch the project.
Measure
Understand the process, validate the data accuracy, and determine process capability.
Analyze
Determine the relationship of Y=f(X)+ε and screen for the potential causes.
Improve
Determine, validate, and implement solutions to achieve the objective statement.
Control
Implement process control methods and monitor performance to sustain results.

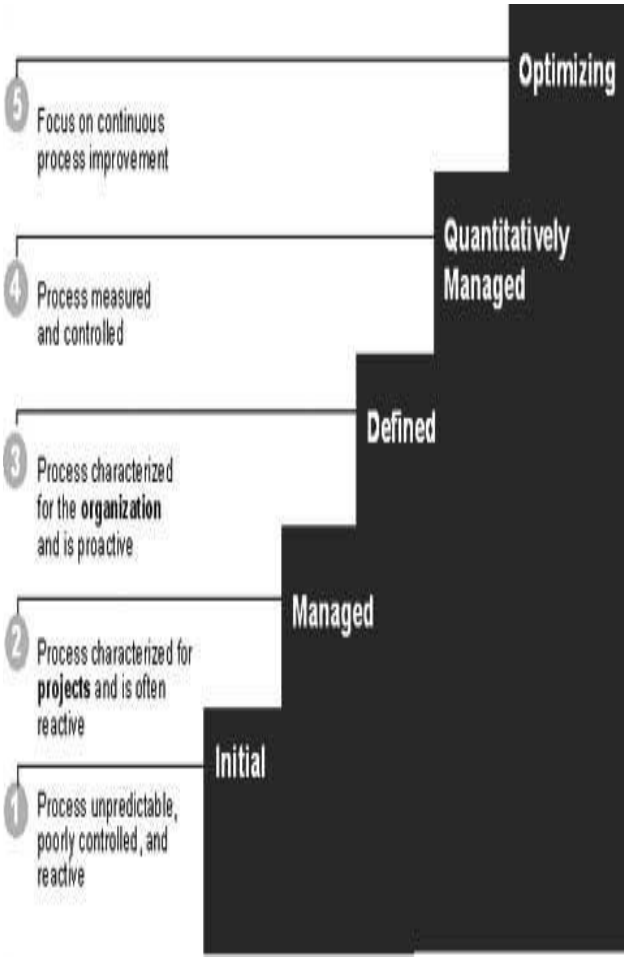
3. Bootstrap BOOTSTRAP is a European method for software process assessment and improvement that was developed to speed up the application of software engineering technology in the European software industry. The BOOTSTRAP methodology is based on the CMM (discussed in the next section). However, it has been extended and adapted to include ISO 9000 guidelines and the European Space Agency software engineering standard (ESA-PSS-05). Unlike the CMM, BOOTSTRAP does not assume strict adherence to a distinct key practice model and allows the use of alternative approaches. This has been a key factor in its success. In addition, BOOTSTRAP has proven suitable for use by all kinds and sizes of software development organizations. The main features of BOOTSTRAP are:

- Questionnaires for both site and project evaluation
- Uniform procedure and mandatory assessor qualification/training
- Constructive instead of a normative approach
- Open questions
- Immediate feedback and action planning

4. CMM The Capability Maturity Model (CMM) plays an important role in the software improvement efforts (SPI) of organizations worldwide. Its goal is to improve, over time, the application of an organization’s software technologies. The model provides a guide for organizations to select software process improvement strategies by facilitating the determination of current capabilities and the identification of critical issues. The CMM process is made up of five well-defined levels of sequential development: initial, repeatable, defined, managed, and optimizing. These maturity levels provide a progressive scale for measuring the maturity of an organization and its ability to use software technologies. Organizations that depend on formal rules, instead of individual performers, to manage software projects are discussed a modified version of the CMM that was more suitable for small organizations and small projects. Problems typically reported with the CCM when used by these organizations were:

- Documentation overload
- Unrelated management structure
- Inapplicable scope of reviews
- High resource requirements
- High training costs
- Lack of need guidance
- Unrelated practices considered to be more “mature.”

Figure 3: CMM


Optimizing
5
Focus on continuous process improvement
Quantitatively Managed
4
Process measured and controlled
Defined
3
Process characterized for the organization and is proactive
Managed
2
Process characterized for projects and is often reactive
Initial
1
Process unpredictable, poorly controlled, and reactive

5. ISO 9001 ISO 9001 is an international standard for quality assurance in design, development, production, installation, and service (Weissfelner, 1999). It is broken down into twenty elements. ISO 9001-3 relates to the development, supply, and maintenance of software. Almost 90 percent of the companies that completed ISO 9001 implementation reported improved internal documentation as one of the most important benefits of registration. Other benefits included higher product quality, greater internal quality awareness, and increased competitive advantage. ISO 9001 is similar to the CMM in the following areas: emphasis on process, documented processes, practiced processes, address the "what" and not the "how" .Differences between the two approaches occur in the areas of focus, dimensions, assessment and certification, coverage, supplier's role, and level of detail. For example, an ISO 9001-compliant software organization would not necessarily satisfy the requirements for a level 2 in the CMM. However, it would satisfy most of the level 2 and some of the level 3 goals. A related article reported the status of implementing an ISO 9001 compliant quality system in a small software organization. Among the challenges encountered during the installation were a lack of guidance, action knowledge, maturity, and quality personnel. In response, ISO procedures were adapted in the following ways: role combination, shorter development cycles, enhanced early communication, simplified procedures, and minimized paperwork

6. Personal Software Process (PSP) The Personal Software Process is a process-based method developed by the SEI for software engineers to use to apply process definition and measurement to their personal tasks. Most important, the PSP shows developers how to manage product quality, meet commitments, and justify their plans with data. In addition, the PSP follows the concepts of the CMM. The key message of the PSP is that developers should use process management concepts to identify the methods most effective for them. A typical PSP course uses ten software development exercises, a structured sequence of defined processes, and five data analysis exercises to demonstrate the process.Examples of improvement made by the process included improved size and time estimating accuracy, reduced project time in the compile and test phases, and improved defect removal yield. The paper also provided a brief introduction to the PSP along with a comparison of the PSP with the CMM.

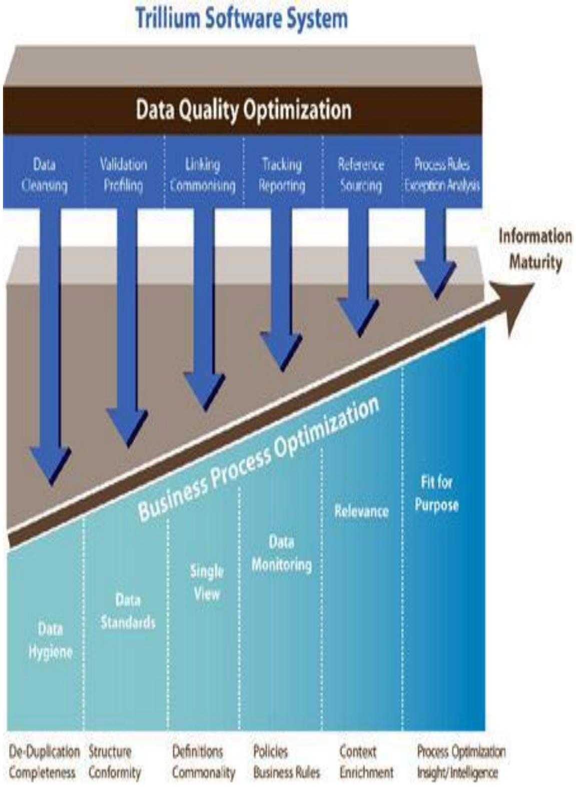
7. Team Software Process (TSP) TSP is a defined method for a group of software developers to create quality software in an efficient manner. It provides process scripts, guidelines, tools, and techniques for a team to develop software applications. The process is based on an incremental model that divides effort into "development cycles." Each cycle involves producing software that satisfies a subset of the total software requirements. In another article discussed how TSP was used during a team software project in a junior level university course. The process provided students with clear, precise guidance and support, good data collection and analysis techniques, and an environment for building a successful software development team. The report concluded that the TSP was an excellent mechanism to emphasize software quality

8. Trillium The Trillium model was initially designed for use with embedded software systems (e.g. telecommunications) and is based on the CMM (Coallier, Mayrand, & Lague, 1999). Its architecture differs from the CMM in the following ways:

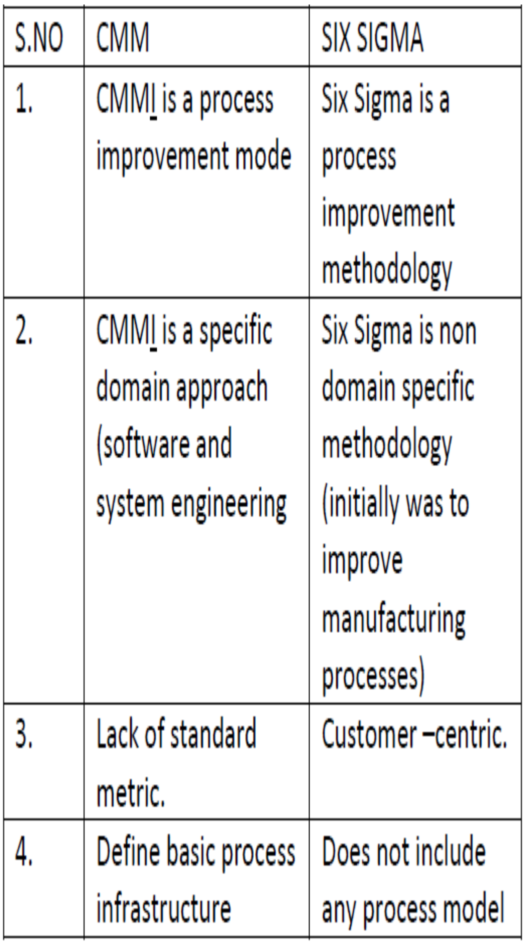
- Architecture is based on roadmaps instead of key process areas
- A product rather than a software perspective
- Wider coverage of capability impacting issues
- Customer focus and a telecommunications orientation

Trillium is comprised of five levels (1-5). These are unstructured, repeatable and project oriented, defined and process oriented, managed and integrated, and fully integrated. Trillium can be used in a number of ways. For example it can be used to benchmark an organization's product development process against industry best practices or to self-assess and identify opportunities for improvement. In addition, it is useful in pre-contractual negotiations to select a supplier.

Figure 4: Trillium



A. CMM VS SIX SIGMA



S.NO	CMM	SIX SIGMA
1.	CMMI is a process improvement mode	Six Sigma is a process improvement methodology
2.	CMMI is a specific domain approach (software and system engineering	Six Sigma is non domain specific methodology (initially was to improve manufacturing processes)
3.	Lack of standard metric.	Customer –centric.
4.	Define basic process infrastructure	Does not include any process model

9. CONCLUSION

The proposed Software process improvement model (SPIM) improves the process in a traditional way. This model is an iterative model. The SPIM model does the process improvement in a stepwise way. It covers user requirements software quality assurance, and organization point of view. Many of the factors can be found in the organization from the SPIM model like management commitment and teamwork were strengthened. SPIM model cover the some limitation of existing model (CMM, SIX SIGMA).For example, the main limitation of CMM is key practice describes. “What to do “but does not prescribe “how to do”. SPIM model describe the Implementation and prescribe how to do. The SPIM model does not necessary to work for the repeatable task .When the new problem come it will work for that also. SPIM is a flexible model. If there is a change in the process, SPIM covers all the aspect of the changing of process due to cyclic model.

10. REFERENCES

[1] Brodman, J., & Johnson, D. (1997). A software process improvement approach for small organizations and small projects. Paper presented at the International Conference on Software Engineering, Boston, MA.

[2] Coallier, F., Mayrand, J., & Lague, B. (1999). Risk Management in Software Product Procurement. In K. Emam. & N. Madhavji (Eds.), Elements of Software Process ssessment and Improvement (pp. 23-44). Washington, DC: IEEE Computer

[3] Dr. Jyotsna Sengupta, Rupinder Kaur, “Software Process Models and Analysis on Failure of Software Development Projects”, International Journal of Scientific & Engineering Research, Feb 2011.

[4] M.P. Thapliyal, Pratibha Dwived, “Software Process Improvement in Small and Medium Software Organizations of India”, International Journal of Computer Applications, 2010

[5] McFeeley, IDEAL SM: “A User's Guide for Software Process Improvement. Handbook” CMU/SEI- 96-HB-001. Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PE, USA.

[6] Donna K. Dunaway, Steve Masters, “CMM SM -Based Appraisal for Internal Process Improvement (CBA IPI): Method Description”, April 1996.

[7] Watts S. Humphrey, “Introduction to Software Process Improvement”, 1992.

[8] B. Curtis, W. Hefley and S. Miller, People Capability Maturity Model. Addison-Wesley: Boston, MA, USA, 2001.

[9] B. Franken, How to Explain Six Sigma by Using the Profit Triangle .

[10] F. Coallier, R. McKenzie, J.F. Wilson and J. Hatz, The Trillium Model, Version 3, Bell Canada, December, 1994.