Software Process Improvement for Small Organizations Based on CMMI/TSP/PSP

ZHANG Lina, LI Ya
School of Computer Science, Henan Polytechnic University, P.R.China, 454000

Abstract Software process improvement is the key issue of the software development technology at present, especially for small Organizations. According to small organizations' characteristic, the paper introduces the Capability Maturity Model Integration (CMMI), Personal Software Process (PSP) and Team Software Process (TSP), and shows the supplementary relationship among them and software process improvement in a small organization based on three of them.

Key words Capability Maturity Model Integration (CMMI), Personal Software Process (PSP), Team Software Process (TSP), Small Organizations

Nowadays, there are many small start up companies. Their core business is outsourcing software development. They are committed to quality software development and focus on creating an organizational culture based on quality. They have strong interest on improvement not just for a CMMI rating. A frequent misconception about adopting Capability Maturity Model Integration (CMMI) is that it works only for large organizations—its cost and complexity appear to make it impractical for smaller organizations to implement. Actually, CMMI implementation complemented with the Personal Software Process (PSP) and The Team Software Process (TSP) might even be more beneficial to smaller businesses because it allows them to grow more consistently and to make changes when they are less costly, that is, “before growth demands them.”

1.CMMI

The CMMI framework is a reference model consisting of best practice descriptions for a broad range of engineering activities, covering the entire product life cycle from requirements definition through delivery and maintenance. It succeeds the Systems Engineering Capability Model (SECM) from the Electronics Industries Alliance, the Integrated Product Development Capability Maturity Model (IPD-CMM), and the SW-CMM, which was originated by the Carnegie Mellon Software Engineering Institute (SEI). The CMMI framework builds on the SW-CMM concepts to provide a mechanism for process improvement that helps organizations to avoid or eliminate these barriers by integrating models that transcend disciplines. As a descriptive model, CMMI is well suited for organizations that are seeking to quantify their capabilities within the scope of software, systems, or product engineering by participating in an appraisal. It is also instrumental in guiding the broad direction of process improvement efforts in each area of expertise.

In CMMI models with a staged representation, there are five maturity levels, each a layer in the foundation for ongoing process improvement.

(1)Maturity Level 1: Initial
At maturity level 1, processes are usually ad hoc and chaotic. The organization usually does not provide a stable environment. Success in these organizations depends on the competence and heroics of the people in the organization and not on the use of proven processes. In spite of this ad hoc, chaotic environment, maturity level 1 organizations often produce products and services that work; however, they frequently exceed the budget and schedule of their projects.

(2)Maturity Level 2: Managed
At maturity level 2, an organization has achieved all the specific and generic goals of the maturity level 2 process areas. In other words, the projects of the organization have ensured that requirements are managed and that processes are planned, performed, measured, and controlled.
(3) Maturity Level 3: Defined
At maturity level 3, an organization has achieved all the specific and generic goals of the process areas assigned to maturity levels 2 and 3. At maturity level 3, processes are well characterized and understood, and are described in standards, procedures, tools, and methods.

(4) Maturity Level 4: Quantitatively Managed
At maturity level 4, an organization has achieved all the specific goals of the process areas assigned to maturity levels 2, 3, and 4 and the generic goals assigned to maturity levels 2 and 3. Subprocesses are selected that significantly contribute to overall process performance. These selected subprocesses are controlled using statistical and other quantitative techniques.

(5) Maturity Level 5: Optimizing
At maturity level 5, an organization has achieved all the specific goals of the process areas assigned to maturity levels 2, 3, 4, and 5 and the generic goals assigned to maturity levels 2 and 3. Processes are continually improved based on a quantitative understanding of the common causes of variation inherent in processes.

Capability Maturity Model Integration (CMMI) is a process improvement approach that provides organizations with the essential elements of effective processes. It can be used to guide process improvement across a project, a division, or an entire organization. CMMI helps integrate traditionally separate organizational functions, set process improvement goals and priorities, provide guidance for quality processes, and provide a point of reference for appraising current processes.

2. PSP
Historically, CMMI principles were used primarily by large organizations. Because of these successful implementations, small organizations wanted to know how they could tailor the CMMI for use in their environments. Could software development teams and individuals apply similar principles to improve their work? Improvement requires change, and changing the behavior of software engineers is a nontrivial problem. The reasons for this explain why process improvement is difficult and illustrate the logic behind the Personal Software Process (PSP).

The PSP consists of a set of methods, forms, scripts, measures, and standards that show software engineers how to use a disciplined process to plan, measure, and manage their work. By using the PSP concepts and methods in their work, engineers in almost any technical field can improve their estimating and planning skills, make commitments that they can meet, manage the quality of their work, and reduce the number of defects in their products. When engineers use the PSP, the recommended process goal is to produce zero-defect products on schedule and within planned costs. When used with the Team Software Process (TSP), the PSP has been effective in helping engineers achieve these objectives.

3. TSP
Organizations using the CMMI to guide their software process improvement efforts often struggle with implementation details. The Team Software Process (TSP) was designed to implement effective, high-maturity processes for project teams. The TSP is prescriptive, defining a whole product framework of customizable processes and an introduction strategy that includes building management sponsorship, training for managers and engineers, automated tool support, coaching, and mentoring. The TSP is a high-maturity process for project teams. It contains an adaptable set of processes, procedures, guidelines, and tools for project teams to use in the production of high-quality software on time and on budget.

Since the CMMI describes what an organization at a high level of process maturity should be doing, and the TSP describes how high-maturity processes are implemented for project teams. Examining the relationship between the TSP and the CMMI shows the TSP strongly supports the key practices of the CMMI, especially the project-level practices it targets. When adopted across a small organization, the TSP is an instantiation of an effective, high maturity process. Small organizations choosing to use the TSP should have no concerns about compatibility between the TSP and the CMMI. In fact, an
organization can build or tailor its CMMI based process improvement effort around the consistent, high-maturity operational framework provided by the TSP.

The objective of the TSP is to create a team environment that supports disciplined individual work and builds and maintains a self-directed team. The TSP guides self-directed teams in addressing critical business needs of better cost and schedule management, effective quality management, and cycle-time reduction. It can be used for all aspects of software development: requirements elicitation and definition, design, implementation, test, and maintenance. The TSP can support multidisciplinary teams that range in size from two engineers to over a hundred engineers. It can be used to develop various kinds of products, ranging from real-time embedded control systems to commercial desktop client-server applications.

4. CMMI/PSP/TSP for small businesses

The CMMI, as well as many other reference models and methodologies for SPI, have been successfully implemented in many large software development and maintenance companies. However, around the world many small software organizations are struggling to implement those reference models. Among other factors, the reason is the lack of guidance for implementing an SPI initiative in small environments, as well as the cost and time that it takes. Organizations seem to know what they want their teams to be doing, but they struggle with how to do it. The Team Software Process (TSP), coupled with the Personal Software Process (PSP), was designed to provide both a strategy and a set of operational procedures for using disciplined software process methods at the individual and team levels. Organizations that have implemented the TSP and PSP can experience significant improvements in the quality of their software systems and reduce schedule deviation.

The PSP shows engineers how to address their tasks in a professional way. It can improve engineers’ ability to plan and track their work and to produce quality products. Once engineering teams are PSP trained, they generally need help in applying advanced process methods to their projects. The TSP guides these teams in launching their projects and in planning and managing their work. Perhaps most important, the TSP shows managers how to guide and coach their software teams to consistently perform at their best. The software process frame should be the integration of CMMI/PSP/TSP, as shown in figure 1.

![Figure 1 software process frame in small organization](image)

The CMMI is the first step of the software improvement. PSP for engineers training is fundamental for CMMI implementation. The TSP builds on and enables the PSP. The PSP shows engineers how to measure their work and use that data to improve their performance. The PSP guides individual work. The TSP guides teamwork and creates an environment in which individuals can use the PSP to excel. These methods are all related, they address different aspects of organizational capability.

5. Conclusion

The CMMI, PSP, and TSP provide an integrated three-dimensional framework for process improvement.
improvement in a small organization. These methods not only help engineers be more effective but also provide the in-depth understanding needed to accelerate organizational process improvement.

The process of software improvement is a gradual process, not an event. When use CMM, PSP and TSP, small enterprises should rational use contacting with the actual conduct, based on its specific circumstances. In implementing the Improvement process for software process under CMM / PSP / TSP, first in-depth study and understanding of the theory, according to their own actual conditions to develop viable programs, and in practice constantly sum up experience and find its way.

References


