THE PDCA-BASED SOFTWARE TESTING IMPROVEMENT FRAMEWORK

LI XU-XIANG[1], ZHANG WEN-NING[2]

[1] Software Technology School, ZhengZhou University,ZhengZhou,450002
[2] Software College, Zhongyuan University of Technology, ZhengZhou, 450007
E-MAIL: lxx@zzu.edu.cn, smile8211@163.com

Abstract:

As one of the most challenging and costly processes, the software testing provides strong support for the high quality software. The popular process improvement models do not adequately address testing application issues, nor has the knowledge management (KM) implementation in testing been well defined. The plan-do-check-action (PDCA)-based testing improvement framework is proposed to address these issues. The framework contains the adaptive processes definition, the processes implementation and measurement analysis, and benchmarking-based assessment approach. Experiments show that: the framework can stimulate team member's subjective initiative, promote the software testing process and improve the testing service quality.

Keywords:

Testing process; knowledge management; PDCA; benchmarking

1. Introduction

With more dependence on information technology and improvement of software complexity, the testing service and authoritative accreditation provided by independent third parties has becoming an important control means of software quality. As these third party testing centers become increasingly aware of service quality's importance, many have been temped to concentrate on KM and invest in improving corresponding testing processes. Unfortunately, many research achievements on software process improvement (SPI) and management methodology don't meet the needs of the third party testing centers and are often hard to apply.

The PDCA-based software testing improvement framework is put forward according to the analysis of heavyweight process approaches, agile methods, KM, benchmarking and accreditation requirement of the China National Accreditation Service for Conformity Assessment (CNAS) to improve service quality and performance at lower cost for independent third party testing centers.

2. Agile Process

In order to solve software crisis, SPI was brought out in the end of 80's in 20 centuries with the intent of improving software quality by controlling and improving process ability of software organizations. Among these achievements, the Capability Maturity Model Integration (CMMI), ISO9000、Software Process Improvement and Capability Determination (SPICE) , known as heavyweight models, are popular and have been widely used all over the world. The credible evidence showed that the heavy-weight based process improvement can result in better project performance and higher quality products[1].But there are organization, especially the small-to-media sized organizations, do not adopt the heavyweight based approach because of the high costly service, intensive management etc[2].

To solve the problems caused by heavyweight models, the Agile Alliance declared the "Agile Manifesto": Valuing individuals and interactions over processes and tools, working software over comprehensive documentation, Customer collaboration over contract negotiation, responding to change over following a plan. The processes most commonly considered agile include Extreme Programming (XP), Lean Development, Crystal, and Scrum. From the manifesto and the application of agile processes, we can see that the practical activities and deliverable products but not processes are the key emphasis in projects. The agile approaches, which have the ability to keep balance between flexibility and stability [3], can respond the market rapidly and improve the service quality. But the advantage of highly dynamic and flexibility limited its scope of application.

3. Benchmarking Process

Benchmarking, comparing the best practices within and across organizations to improve performance, was called one of the three most important innovations in management methodologies in 90's in 20 centuries[4]. Under the goal of continuous pursuit of excellence, the software benchmarking process improvement model (as shown in Figure 1) has been developed and widely used in American, Australia, England, Japan etc and extensively studied by ISBSG, SPG、PBC etc. Also, the independent third party testing centers can adopt the method to identify the competitive advantage and gain competition in global information-intensive economies.

The software benchmarking model is the process of measuring the organization's performance against best practices, acquiring or transferring good practices that can be adopted. It is divided into several key phases: selecting the most valuable improvement needs; conducting assessment and benchmarking to identify performance gaps; setting the adaptive improvement objective; learning from the software best practices; assessing and analyzing the improvement effect to serve the business targets [5]. However, the software testing relative data in public repositories is relative small and needed to be further collected and analyzed to support the comparison of software testing activities.

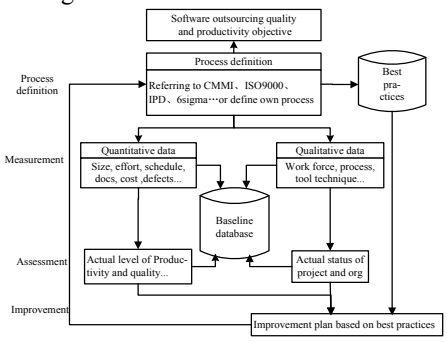

Figure 1: Benchmarking Process

According to the essential features of benchmark and the implementation effect for recent years, the benchmarking process is helpful to cultivate the spirit of cooperation, collaboration, trust and communication, conducive to the construction of organization knowledge-oriented culture.

4. Knowledge Management

According to the process analysis above, the KM ideas exists in most improvement methods but not state clearly. Furthermore, the promoting effect of KM on the SPI activities and core-competence has drawn great attentions from industries and experts. For example, Prof. Ikujiro Nonaka and Hirotaka Takeuchi believe that the key to the success of most organizations is the knowledge sharing, innovation and utilization, especially the tacit knowledge; In India lots of organizations established the meaningful KM systems to sustained viability [6].

The KM comprises a range of strategies and activities adopted in organizations to identify, create, present, distribute and capitalize on knowledge which embodied in the set of work practices, skills, equipment, processes and heuristics of individuals. It ensures that the knowledge is conveyed to the individuals who need it the most at the right time. In this way, the organization can sustain its competitive advantages, enhance learning and innovative capabilities and keep sustainable development.

5. The PDCA-Based Testing Improvement Framework

With the purpose of performance improvement, the PDCA-based testing improvement framework presented in the paper focuses not only on standard and agile processes but also on individual's subjective initiative and competitive advantage. It evokes subjective initiative well through KM, keeps the balance of processes standardization and flexibility, supports benchmarking and manages the core intangible asset for sustainable improvement.

The framework is divided into phases(as shown in Figure 2): building the learning organization to stimulate subjective initiative through KM; preparing the process-oriented plan[7] which defines the just-enough adaptive testing processes based on the organizational processes; implementing the plan and collecting data; date analyzing and benchmarking; identifying the improvement needs and implementing improvement following the GQM approach.

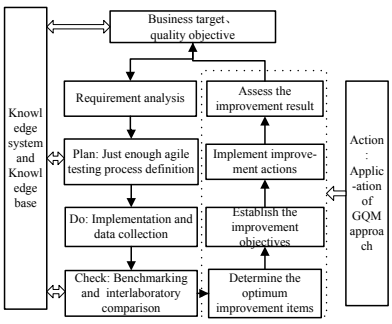

Figure 2: The PDCA-based test improvement framework

5.1. Build Learning Organization through KM

From the famous Cobb-Douglas production function and the research for economic development source in United States by Edward Fulton Denison, we see that knowledge is becoming the core profit source for organizations. As a typical learning organization, the third-party testing center must have a knowledge-oriented culture and establish the learning system to foster innovation and enhance the core competitiveness.

The knowledge can be divided into explicit knowledge and tacit knowledge [6]. According to the analysis of software testing activities, the explicit knowledge related to software testing includes:

(1) Software engineering knowledge: popular SPI models or methods, SPICE, IDEAL, CMMI etc.

(2) Business domain knowledge: business models and procedures in different industries, finance, logistic etc.

(3) Software testing knowledge: testing resource of historical projects, plans, strategies, testing cases etc.

(4)Common knowledge: baseline database, reusable testing cases library, testing tools and techniques related, best practice and failure cases etc.

(5)Personal knowledge: feedback, thought and skills of testing managers, team leader and testing engineers.

The tacit knowledge is the abstract, logical, experiential information which lies at the bottom of testing engineer's heart, psychology and behavior. And it should be mined, transferred, presented and recreated through some good practices, such as shadowing, brainstorming, review, and joint problem-solving etc.

To facilitate learning, the testing centers should set up their knowledge base for future usage and build the effective learning mechanism, such as reading party discussion, expert network and personalized training. The learning culture resulted from KM is helpful to set up a testing team with high technique, high cooperation and high efficiency. The team, also called as flexible team, is the important basis for the framework implementation.

5.2. Plan the Adaptive Testing Processes

Under the just enough strategy in agile process, the flexible testing team prepares the process-oriented plan based on the analysis of requirement/essential features of popular SPI models and its actual resource(s/w, HR, tools, technique, communication ability etc).The plan reviewers pay more attention to the necessity, agility and visibility of each process in the plan. The detailed description of the processes in plan is as followed:

Definition 1: Project Adaptive Process (PAP) is the just enough process set defined for the testing project in the plan. It contains the processes can not be tailed at organizational level (recorded as Pro_i) and the special processes for the project (recorded as $Pro_m^{'}$).

PAP= $\{\ldots Pro_i, \ldots Pro_m^{'} \ldots\}$.

Definition 2: Project Adaptive Process Element (PAPE) is the list of process elements for each process in PAP. So the $Proe_{ij}$ indicates the j^{th} element in the process. To explain the relationship between PAP and PAPE more clearly, we can use the following description:

$Pro_i = \{\ldots Proe_{ij} \ldots\}$.

In the description above, the process element $Proe_{ij}$ is defined using the seven species as shown below:

Proe= {Input-products, Precondition, Activities, Post-condition, Output-products, Environment, Measures}

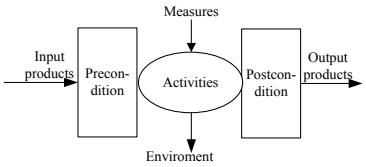

Figure 3: Process element definition

To keep the balance of standardization and flexibility, the testing team can follow the Good Enough Quality (GEQ) approach to discuss the rationality, necessity and existing value of each process and its elements matched.

5.3. Plan Implementation and Data Analysis

At the implementation stage, the software testing team analyzes the collected measurement data and their internal relationship using one or several quality management tools, including checklist, tendency chart, histogram, Control chart, and Cause and Effect diagrams, scatter diagram, Pareto diagram [6]. The team can refer to the measurement and analysis process area in CMMI for more support information.

Based on the measurement analysis in the organization, the flexible testing team compares the data with the baseline data provided by the International Software Benchmarking Standards Group (ISBSG) and China Software Benchmarking Standards Group (CSBSG). Also they can take part in the Proficiency Testing Programs of inter-laboratory comparisons held by CNAS to identify the gaps of process, technique and resource.

5.4. Continuous Improvement

Aligning with the business goal, the organization discusses and determines the priority of improvement needs based on the result of benchmarking and inter-laboratory comparison. The activities at this phase can be conducted by external consultants, similarly the SCAMP B / C appraisal for CMMI.

Based on the analysis and improvement priority, the organization can realize its own advantages/disadvantages and prepare the improvement action plan. At the following phases, the organization sets improvement objectives, implements the action plan step by step and assesses the improvement effect according to the goal-question-metrics (GQM) approach. At last, the practices and achievements proved to be appropriate and effective should be in cooperated into the knowledge base and distributed to the staff to foster innovation.

6. Case Studies

The framework was applied in the following three testing projects: Forewarning and Monitoring Management System for Coal Mine, Fiber Channel Switches System, The Wavelet Denoise System for Retinal fundus Images in the third-party testing center. With the accreditation qualifications of CNAS, the testing center developed a comparatively sound knowledge management system such as regular discussion and expert network and maintained the basic knowledge base referring to the testing processes, reusing testing case libraries. Additionally, the feasible testing team, with good communication skills and strong team spirit, lays the foundation for the implementing of the lightweight improvement framework.

To verify the feasibility and efficiency of the framework, all the three projects were in strict accordance with the framework. At the start-up phase, the team planed different adaptive testing processes for each project based on its requirement. All the processes were reviewed by team members. And during the implement phase, the team performed accordance with the planed processes. Some measures, such as effort, size, and defect density were collected. At the end of these projects, the business model of coal industry was optimized further and 32 reusable testing cases and testing processes special for the network device were submitted to the knowledge base.

Combining with the result of benchmarking and inter-laboratory comparison for the testing project of article uploading management system provided by the CNAS, the testing center identified its own deficiency, for example, the team couldn't keep good control on the standardization and flexibility. Based on the analysis, the testing center prepared the action plan which focused on the automation testing tools, agile process learning and the establishment of reading party.

The practices have shown that the improvement framework is flexible, easy-to-use and efficient to inspire the initiative of the staff. The testing activities based on KM promote the software reuse activities of testing cases and increase the testing center's competition advantage.

7. Critical Factors

Although the framework can improve the testing service agilely in high standards at lower cost, there are some strong constrains that may significantly affect the actual effect.

(1)The organization should build and strengthen the knowledge-oriented culture, try their best to arouse the team personnel's work enthusiasm.

(2)The organization should understand the popular process models completely when planning the adaptive testing processes.

(3)The KM strategy must be within the management framework at the organizational level and support from senior managers should be developed.

8. Conclusions

This thesis raised the PDCA-based testing improvement framework based on the analysis of agile process, benchmarking process and KM. Although it is simple and flexible, there are some details to be worked out, such as the mechanism of knowledge mining, sharing etc, which provide the basis to further study.

Reference

[1] Goldenson DR, Gibson DL, Demonstrating the Impact and Benefits of CMMI: An Update and Preliminary Results, CMU/SEI-2003-009, Carnegie-Melon Software Engineering Institute, Pittsburgh, PA, 2003

[2] Mark Staples, Mahmood Niazi, Ross Jeffery, An Exploratory Study of Why Organizations do not Adopt CMMI, Journal of Systems and Software, Vol.80, No. 6, pp. 883-895, June 2007.

[3] Jim Highsmith, Agile Project Management, Adison-Wesley, New York, 2006.

[4] H.James Harrington, High Performance Benchmarking, McGraw –Hill, New York, 2003.

[5] ZHANG Wen-ning, LI Xi-yan, ZHOU Qing-lei, Model research of testing process based on comparison,

Computer Engineering and Design, Vol.31, No.4, pp. 696-699, Apr.2010.

[6] WAN Jiang-ping, LI Jian-zhang, HAN Hui-feng, Knowledge Management in Software Organization, Application Research of Computers，Vol.20, No.1, pp. 13-16, 2003.

[7] WU Zhan-Chun, WANG Qing, LI Ming-Shu , A PDCA-Based Software Process Control and Improvement Model, Journal of Software, Vol.17, No.8, pp. 1669-1680, Aug.2006.

[8] Li, Jian-Ping, Effects Of Topologies On Network Recovery,for ICACIA 2009.