

# The Software Quality Management of Information System

Hua Wang Xianyu Li Boshan Shi Yunjian Zhang and Weina Liu

Beijing Shi QingHe DaLou Zi Jiu, 100085

Beijing, China

919wh\_cn@sina.com

**Abstract**—Based upon the intention summary of software quality management, this paper give plain description of PDCA circulation, software quality scheme, quality control, quality assurance and quality improvement. Meanwhile, one quality evaluation is put forward. This will benefit the software quality management of information system.

**Keywords**- *information system; software process; quality management.*

## I. INTRODUCTION

In view of the complexity of software architecture and difficulty of software integration, software quality management should be strictly and actively carried through during the process of the information system development. Quality management is to ascertain quality guideline, objective and responsibility, and to perform the whole functions via quality scheme, control, assurance and improvement. Software quality management runs through the whole life cycle of requirement analysis, design, code, testing, delivery and maintenance.

Software quality management is a process where the project is ensured to meet the needs of the given standards, and is made up of quality scheme, control, assurance and improvement. Software quality management proceeds by following the thinking of systems engineering and adopting the definable and measurable methods. In the light of quality management principle and software engineering demand, we strive to implement the software quality management to insure the whole quality and operation effectiveness of the information system [1, 2, 3].

## II. PDCA TECHNIQUE

Software life-cycle phases consist of many processes; the management and control of software processes means that PDCA technique is naturally implemented.

PDCA is the combination of the first letter of Plan, Doing, Check and Action. PDCA circulation is also called Deming Ring. PDCA technique should be handled into all phases and processes of software development. PDCA circulation consists of four phases and eight steps. PDCA circulation sketch is shown in Fig. 1.

### A. PDCA Step

PDCA circulation has the following steps.

1) *Plan phase*. It contains four steps.

a) *The first step*. Quality actuality is analyzed to find out the existed questions.

b) *The second step*. The influence factors that result in quality questions are analyzed.

c) *The third step*. The main causes are found out.

d) *The fourth step*. The measures are pertinently drawn out, and the plan is put forward to put into practice the quality improvement activities.

2) *Doing phase*. The measures are actually taken according to the plan and objective, and it is the fifth step.

3) *Check phase*. The implement complexion and effect is inspected to find the experience and lectures, and it is the sixth step.

4) *Action phase*. It contains two steps.

a) *The seventh step*. The former regulations and standards are revised to consolidate the achievement and avoid the like problem.

b) *The eighth step*. The unsolved questions are switched to next circulation as the further information.

### B. PDCA Characteristic

PDCA circulation has the following characteristics.

- Big ring and small ring are mutually accelerated of PDCA circulation.
- PDCA circulation continually runs and quality is improved step by step.
- PDCA circulation is mainly impelled by the action phase.

And the process of software quality management is one PDCA circulation and quality scheme, control, assurance and improvement correspond to the related PDCA phases.

## III. SOFTWARE QUALITY SCHEME

By compiling and conducting the project of quality management and the plan of development work, the related requirement, design and test works are systemically arranged, and the main nodes and the corresponding objectives are specified. Moreover, the accomplishment of these works is monitored and inspected. A series of software engineering standards are founded and project management tools are equipped. The templates of technical files and track records are prescribed, and this is helpful to the fulfillment of process tasks and the quality of technical files [4, 5, 6].

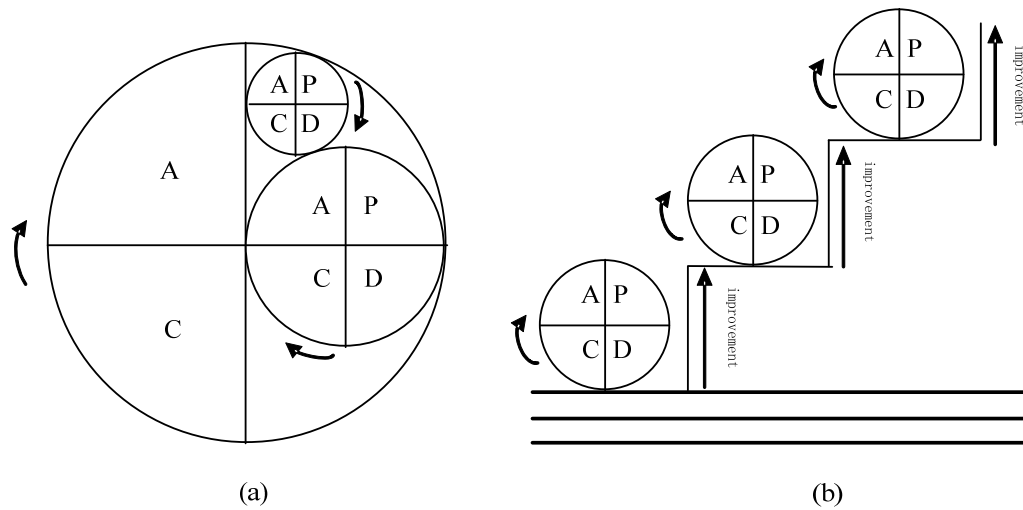


Figure 1. PDCA Circulation Sketch

#### IV. SOFTWARE QUALITY CONTROL

A series of quality control technologies are applied to the processes of software development in order that software quality demands could be better fulfilled.

##### A. Software Technical Review

The software technical reviews are emphasized to fully uncover the problems of the present works. The reviews are carried through of requirement analysis, general and particular design, validation testing, joint test and development milestone validation.

Review teams go through the phase work and products, and form the review notion. The index and trend of software quality is analyzed, the process and outcome are noted, and the problems and proposals of reviews are dealt with and certificated before the next phase. The standard of the software technical review is that present works have exactly actualized fulfilled the scheme put forward before and present scheme is clear and feasible to put into practice afterward.

Milestone reviews are carried through to confirm that the related tests have been completely and effectively conducted, the software system has satisfied the demands of related phases, and the technical states of software have been distinctly founded. After the last milestone review, the software system can perfectly be delivered to the users.

##### B. Software Test Control

In the process of software development, we carry out the unit test, the component test, the validation test and the system joint test which proceed according to the related standard, plan and procedure. We carry through clear and elaborate prescript for the test condition, approach, content and pass criteria, establish the test plan and test description, and note down test results and conclusions.

With respect to the tests, the closed loop policy of test implement, fault analysis, program modification and validity validation is strictly brought into effect. Moreover, the

problems of tests are fully solved and prevented from accumulation and diffusion, and the software tests are examined respectively.

We should further take the personnel, tools, and circumstance into account, investigate testing technique and approach, and optimize the process of software tests, so that the sufficiency and efficiency could be improved of software test, and the quality ensured of software process and product.

##### C. Software Configuration Control

The software configuration management is strictly implemented. The plan is constructed to make clear the activities of software configuration management. The control process of software revision is shown in Fig. 2.

The function baseline, assignment baseline and product baseline are set up, and marker method is fixed on of the programs, documents and versions. In addition, the development library, controlled library and product library is founded at project level and collectivity level with the support of the interior local area network.

The alteration and access of various baselines and configurations are strictly managed. The variations that surpass the baselines are actualized according to the control demand, and the configuration state is renewed in time. And software configuration is audited before software tests and delivery. Thus the integrality, consistency and traceability of software configurations is well ensured.

#### V. SOFTWARE QUALITY ASSURANCE

In order to ensure the delivered software could meet each requirement and use needs, the necessary measures of software quality assurance are prescribed as follows.

- To make sure the rules and standards that software projects should comply with;
- To compile software quality assurance plan;
- To examine and look through the software processes of requirement management, configuration management, testing, tracking, etc;

- To audit the work products of requirement, design, test, etc;
- To evaluate the quality of software product.

The problems found in the tests, reviews and are solved in time. And the quality of software product is practically analyzed and evaluated, and the quality trend is predicted and compared with the quality goals via line chart, variation analysis, and causation chart and so on. The process of software quality assurance is shown in Fig. 3.

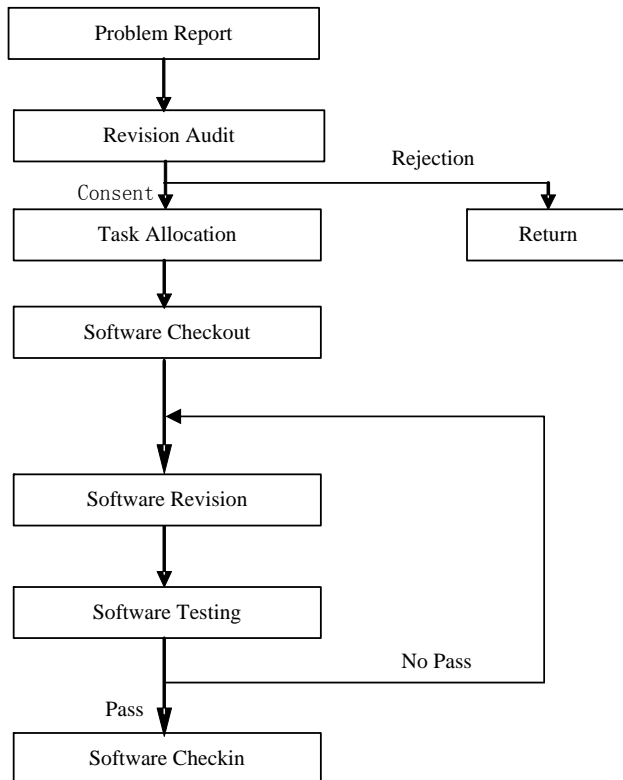


Figure 2. Control Process of Software Revision

## VI. SOFTWARE QUALITY IMPROVEMENT

In the process of software development, we run the SFRACAS all the time. The software requirement analysis, the software design, the software tests and acceptance testing are accomplished and reviewed, the problems found in the tests and reviews are entirely analyzed and solved. It ensures the effectivity of measures implemented and the actuality of problem solution, so that software quality could be continually increased.

To help and accelerate software maintenance and alteration, we sum up the information of software problems, and form the typical cases of software problems that practically depict problem phenomenon, cause analysis, corrective measure and validation.

The software alteration processes are strictly managed. And software configuration is audited before software tests and delivery so as to ensure the availability and controllability of software configurations.

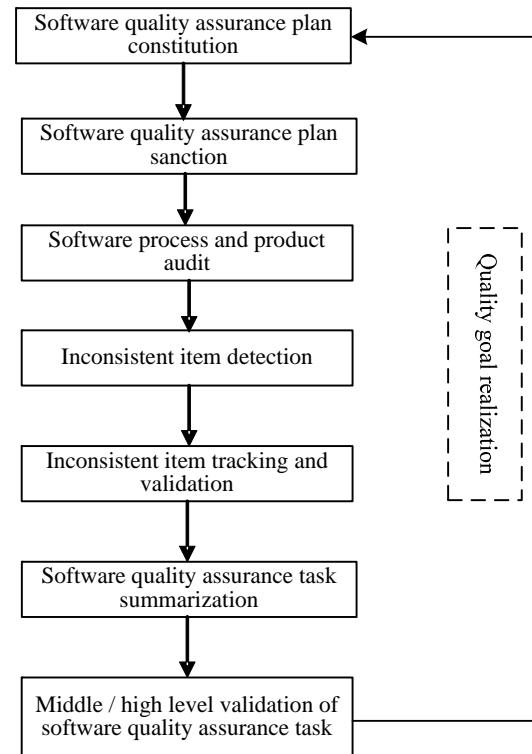


Figure 3. The Process of Software Quality Assurance

## VII. SOFTWARE QUALITY EVALUATION

Related statistic has indicated that to regularly implement software engineering methods could come up to higher quality. Software capability may generally be scaled with five grades of CMM, i.e., I grade—first grade, II grade—repeatable grade, III grade—defined grade, IV grade—managed grade, V grade—optimizing grade. With respect to each maturity grade, CMM lists some Process Areas that are the must when an organization would achieve the relevant maturity grade.

Compared with the software capability grades, we are now just in between II grade and III grade. There are still many things to do with respect to measurement and analysis, requirement development, supplier agreement management and etc. Thus we need further strengthen the technical and management ability of software development processes, richen the activity content and standard of software process practices, and upgrade the capability and level of software quality management [7, 8].

Software quality is also evaluated by means of related quality model including the quality characteristics of functionality, reliability, usability, efficiency and maintainability, and be further quantified by adopting software quality metric model of characteristic, sub-characteristic, metric, data element. Thus we can examine the fulfillment extent of the specified quality requirements for software product, analyze the practical relation and

evolution trend among quality factors, which serve as the input information of new round of quality management.

#### VIII. CONCLUSION

The software quality work will be more continuously advanced and expanded along with the improvement of software process capability and systematism of quality management. We should improve the technique and management capability of software processes, and advance the progress of software quality scheme, quality control, quality assurance and quality improvement, so as to promote the continual improvement of the quality of the application software and the information system.

#### ACKNOWLEDGMENT

We are grateful to the specialists and secretaries for their guidance, understanding and patience, and they are a constant source of support and encouragement.

#### REFERENCES

- [1] G. W. He, W. Wang, Software Reliability. Beijing: National Defense Industry Press, 2001.
- [2] Z. Y. Zhou, Modern Software Engineering. Beijing: Science Press, 2002.
- [3] Z. Shi, One Decade of Spaceflight Software Engineering (Report). 2005.
- [4] R. J. Zheng, Practical Software Engineering. Beijing: Qinghua University Press, 2002.
- [5] L. Ruan, M. Y. Lu, et al., Quality and Reliability Management of Equipment Software. Beijing: National Defence Industry Press, 2006.
- [6] D. Galin, Software Quality Assurance. Pearson Education Limited, 2004.
- [7] C. Tasso, Software Quality Engineering. Computational Mechanics Publications, 1997.
- [8] L. P. Shari, Software Engineering: Theory and Practice. Prentice-Hall, Inc., 1998.