The Project Management Practice of the Development of the Application Software

XianYu Li, Hua Wang, Gang Xiang, Hui He, Hui Wang
Beijing Shi QingHe DaLou Zi Jiu, 100085
Beijing, China
919wh_cn@sina.com

***Abstract*—Based upon the summary of software project management, this paper concretely puts forward the project works, such as integration management, risk management, requirements management, epiboly management, quality management, configuration management, resources management and schedule management, and clearly points out that we should pay attention to project process inspection, enhance project measurement tracking and strengthen project production summing-up. Thenceforth a basic conclusion is presented.**

Keywords- project management, software engineering, application software

I. INTRODUCTION

Project management is that knowledge, skill, means and technology are applied to effectively arrange, organize, execute and control the project so as to achieve the dynamic management of the whole process and the harmonious realization of the project object. Project management contains the basic processes of startup, scheme, execution, control and ending, which join sequentially and may overlap to some extent [1, 2, 3].

Software project management is the application of methods and principles of project management to the domain of software engineering, and the processes are as follows:

- The startup of software project. The target and scope are made clear, and the technical scenario and management requirements are taken into account.
- The establishment of the project plan. The workload and resources are reckoned, and the schedule, configuration management, risk management and quality management are instituted.
- The tracking and control of the project plan. Under the project processes, the project plan is complied with, and properly regulated with respect to some inevitable variations, so that the integrality and consistency is insured.
- The reviews of activities of the project plan. The fulfillment circs of the project plan is reviewed, and the execution complexion of the project is estimated.
- The ending of the project. The project is made sure that whether it is completed according to the contract, and the outcoming and documents are compiled, inspected and preserved.

II. RESUME

Software project management mainly includes the following content [1, 2].

- Software integration management comprises the processes where every project element can be properly harmonized, such as scheme, execution and variation control of the project plan.
- Software risk management is a practical process where the risks are estimated and controlled that affect the software project, process or products.
- Software requirements management is a sort of systematization scenario, and a process of satisfaction and coherence of the requirements.
- Software epiboly management is that the qualified software epiboly units are selected and effectively regulated.
- 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, assurance, control and improvement.
- Software configuration management is to identify and define the software configurations, dominate the release and variation and report the state, so as to insure the completeness and correctness of software configurations.
- Software resources management comprises the programme and analysis of manpower, the reuse of software resources, and the management of CASE tools and hardware equipments.
- Software schedule management comprises the task definition, ordering, duration estimation and schedule domination.

Further more, software project management involves the workload and cost estimation.

III. PRACTICAL ANALYSIS

During the development of the application software, the spiral process model is adopted to guide the software process management [4, 5, 6].

978-1-4577-0536-6/11/$26.00 ©2011 IEEE

We adopt the OOA method to denote the functions and information model by usecase chart, the OOD method to describe the information flow and design by sequence chart, activity chart and class chart, and the OOP method to program by VC++, Delphi and PB.

During the development of the application software, the software engineering standards are followed of the software requirements analysis, software test, and software reliability and safety. The software project management structure is shown as Fig. 1.

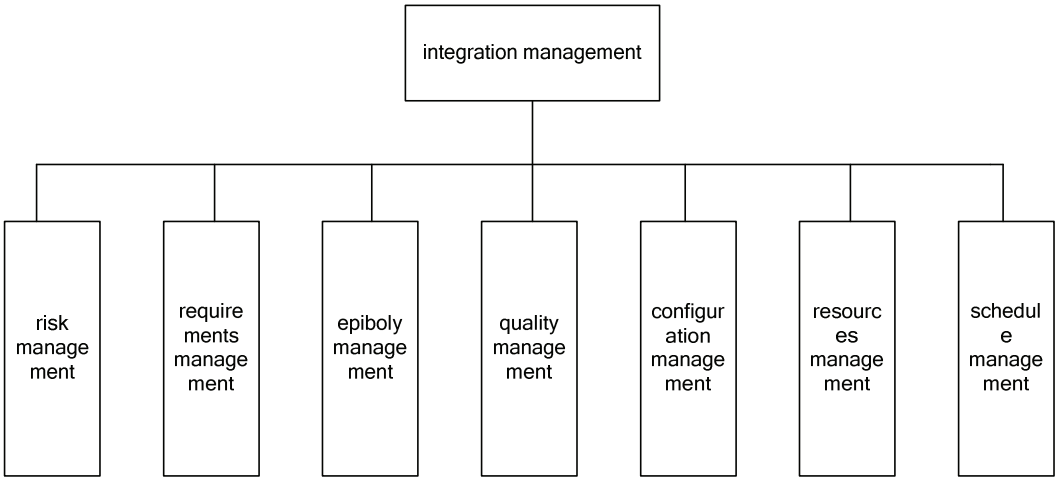

Figure 1. The Software Project Management Structure

A. Integration Management

The project plan is established to integrate and harmonize all the task contents of quality assurance, configuration management, work production, task schedule, resources sustainment and problems that probably exist.

The project plan is approved by technical master. The development activities are conducted with the inspection and tracking according to the plan, and the phase reviews are carried through in the light of review requirements. Moreover, the software unit test, component test, validation test, and the installation and handover of the software are systemically schemed. The project plan is regulated by the alteration procedure with the agreement of collectivity group.

B. Risk Management

The main risks and corresponding measures of the software project are shown in Table I.

TABLE I. RISK ANALYSIS

Activity	Risk Description	Policy
requirements management	the growth of the users' requirements	to collect requirements by prototype; and carry out explicit regulation of the requirements.
requirements analysis	the strong relevancy among the software system	to strengthen the requirements analysis of software interface and the joint test.
epiboly management	epiboly unit cann't provide conformable production in time	to fully evaluate the technical ability and the credit standing of the epiboly unit, and make clear the responsibility, right and benefit in the contract.

C. Requirements Management

The connotation of the software requirements is so abundant and complex that we should fully and exactly understand and describe users' demands, and finally come to the software requirements.

The requirements of software functions, performance, architecture, information exchange and system constitution are distinctly managed during the occasions of project demonstration, overall scenario compilation and contract negotiation, and continually tracked during the development of the application software with users' affirmation.

D. Epiboly Management

The software epiboly is reviewed and approved, and the epiboly process is carried out under the supervision, and the following measures are taken.

- The quality management system of the software epiboly unit is checked to affirm its satisfaction of needs.
- The software production is referred in time in the light of collectivity demands.
- The software production is inspected for correctness and standardization during the software technical reviews.
- The software product is tested, tried out and validated with problems effectively solved.
- The software configuration management should meet the needs of project collectivity, and the regulation of

the transaction and variation of configurations be insured.

E. Quality Management

The rules and standards are fixed on and complied with during the software development.

The software requirements specification and the design description are strictly reviewed, so that the correctness of requirements and the rationality of design are fully ensured.

The software tests, evaluating test and try out are carried out to validate the coincidence of the software functions and performance during the corresponding phases.

The software processes and products are examined by independent software quality assurance persons.

The problems found in the tests, try out and reviews are solved in time.

And the quality of software product is practically analysed and evaluated.

F. Configuration Management

The software configuration management is strictly implemented. The plan is constructed to make clear the activities of software configuration management.

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 " Two-level and Three-library" configuration management model is described in Fig. 2.

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.

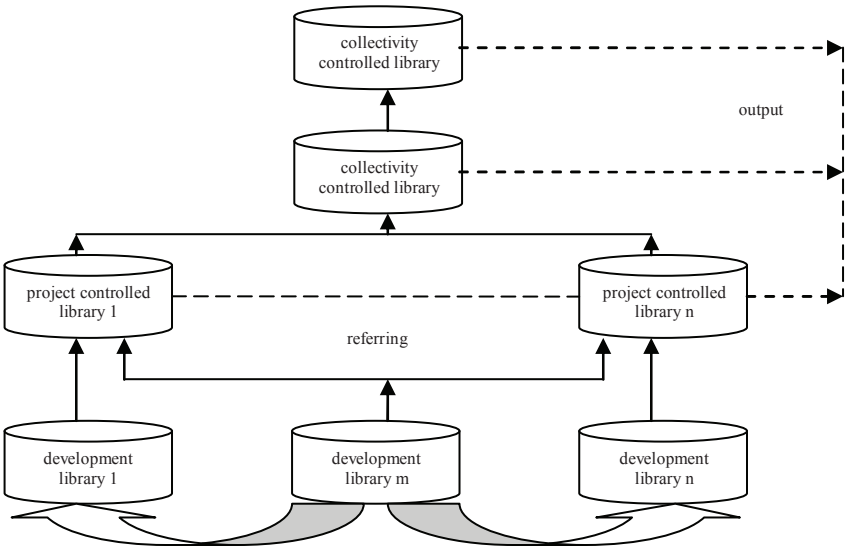

Figure 2. The " Two-level and Three- library" Configuration Management Model

G. Resources Management

We carry out the work analysis and task breakdown, and the skilled and talented persons are uniformly and effectively organized and deployed.

The development facility mainly consists of PCs, servers, LAN, design tools, quality management tools, software integration tools and flow management tools, and the test circumstance and operation data.

During the development of the application software, the percentages of primary, improved and newmade components are respectively 33%, 43.2% and 23.8%.

The project personnel are organized as shown in Table II.

TABLE II. PROJECT PERSONNEL

Title	Number	Responsibility
technologist team	11	important technical issue audit
technical collectivity team	3	technical guidance and coordination, scenario integration and optimization
software development team	50	requirements analysis, software design and programming and testing
quality assurance team	3	software process and work audit
configuration management team	2	configuration control and state management

H. Schedule Management

The schedule is described as shown in Table III.

TABLE III. SCHEDULE ORGANIZATION

Phase	Activity	Date
elementary system phase	the requirement analysis and review, the software design and review; the unit test, component test, validation test and system joint test; the elementary system review	2010.06
formal system phase	the requirement analysis and audit, the software design and audit; the unit test, component test, validation test and system joint test; the formal system review	2010.09
design finalization phase	the finalization testing and the users' try-out, the finalization document audit and the finalization system review	2010.12

IV. THE WORKS TO BE IMPROVED

Based upon the present work, we should place great emphasis on the following aspects relating to project management.

A. Paying Attention to Project Process Inspection

We should establish the quantity management regulation, and collect and sum up the data item during the project process. The project situation, such as the task completion, risk change, schedule state, problem disposal, productivity, function point accomplishment and fault statistic process, should be tracked, and the project state report and quantity analysis report completed and submitted.

B. Enhancing Project Measurement Tracking

Based upon the periodic collected datum, we should detect process windage, analyse the reason of process abnormity, judge whether the project process is steady and under control, so that we can evaluate the process capability and possible improvement, and explore the relation among the development pattern, process trend and project groups.

C. Strengthening Project Production Summing-up

When the project is completed, the software production is audited and referred to the users, and the more important thing is that it's the key period of information syncretization and process optimization. We should analyse the process performance and the innovation, and put forward the improvement proposition with respect to the organization process, so that the standard flow and the assets library can be perfected.

V. CONCLUSION

During the development of the application software, aiming at the development target and management demand, we positively carry through integration management, risk management, requirements management, epiboly management, quality management, configuration management, resources management and schedule management, and effectively ensure the system construction and task accomplishment. In addition, we should further pay attention to project process inspection, enhance project measurement tracking and strengthen project production summing-up, so as to promote the continual improvement of the technical level and process capability of the application software and the related system.

ACKNOWLEDGMENT

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

REFERENCES

[1] Zheng Tan, etc., Software Project Management. Beijing: Qinghua University Press, 2005.

[2] Zhiying Zhou, Modern Software Engineering. Beijing: Science Press, 2002.

[3] Renjie Zheng, Practical Software Engineering. Beijing: Qinghua University Press, 2002.

[4] Daniel Galin, Software Quality Assurance. Pearson Education Limited, 2004.

[5] Tasso C., Software Quality Engineering. Computational Mechanics Publications, 1997.

[6] Shari L. P., Software Engineering: Theory and Practice. Prentice-Hall, Inc., 1998.