

Research of Software Development Model Based on the Theory of Software Engineering

Lei Wang

Department of Computing
The Chinese People's Armed Police Forces Academy
Langfang,065000,China
E-mail:WangLeiwjxy@qq.com

Abstract—purpose of research of software development is to attempt to develop lower cost , high reliability, and a variety of software to meet user needs. This paper introduce the advantages and disadvantages of various software development model from software engineering perspective.It proposed use of scientific principles and methods of engineering specifications to guide the development of the software , run and maintain , and trying to organize the work of software development , and then explore a new way to get rid of the software crisis.

Keywords-Software, software engineering, software development model

I. INTRODUCTION

The concept of Software Engineering first appeared in the 1968 Conference held in the Science and Technology Committee under the North Atlantic Treaty Organization. The meeting for the software crisis of the hazards of the field of software design and software development methodologies and the software development process , a number of issues to discuss and work out a solution to get rid of the software crisis ideas and measures.

In view of the software development process encountered all sorts of trouble, people hope to be able to by formulating rational and effective and scientific solution, the development of reliable quality, low cost, high reliability, economical and easy to maintenance software products. In NATO meeting for the first time not only puts forward to use "software engineering" to define software development need knowledge, also suggested that software development should be similar engineering activities. Since 1968, and formally put forward until now, has accumulated a large quantity of research achievements made, and a large number of technical practice, nowadays, software engineering has developed into a professional subject.

II. THE SOFTWARE DEVELOPMENT MODEL

Software development model refers to the structural framework of the software development process , activities and tasks. Software development model can visually express the whole process of software development, it clearly specified software development to complete the main activities and tasks, is the basis of the work of the entire software development projects . For different software

system,it can adopt different development model for development. Among them, the typical software development model are: the build-and-fix model, the waterfall model,the rapid prototype model, the spiral model, the incremental model, the fountain model, the intelligent model, the hybrid model etc.

A. *The model of software development based on the build - and - fix mode*

At present, many software products still continue to use he build-and-fix model to the development, in particular, the program scale is particularly common in the hundreds of lines of small-scale software projects [1]. However , in this model , neither specification, nor the process design , the software with customer demand once again to be constantly modified. Specifically , the project requirements in the development process of this model , developers get immediate programming based on customer needs , the first version of the software will generate after debugging success then presented to the user. Then, in the use process, if the program appears a mistake, or the users put forward the update round of demand, developers have to re - modify the program code , until the user satisfaction. This is a similar to "the manual workshop" type development mode, for those small-scale software projects can barely cope with, but for those large-scale development projects become stretched, its main shortcoming are as follows:

- The development model lacks of planning and design link, software structure is always with the user demand change and continually being modified, it may lead to the final program structure is constantly changing, and it makes the development process more and more messy, and thus it may lead to unexpected failure of software development.
- The model ignore the link of needs, it brings great risk for software development.
- The model did not consider to test and the maintainability of the program, the development process lacks of document material, which make software later maintenance work very difficult.

B. *The model of software development based on the waterfall mode*

The waterfall model which is put forward by Winston Royce is first appeared software development model,it

occupies an important position in software engineering, it provides the basic framework of software development. Until the early 1980s, it has been one of the software development model, which has been recognized as is widely used in [2]. The core idea of the waterfall model is the use of structured analysis and design method of the logic and physical realization of phase separation, according to processes problem simplification, in order to facilitate the cooperation of division of labor. Waterfall model gives a fixed order, the software life cycle is divided into planning, requirement analysis, software design, programming, software testing and maintenance are six basic activities, and provides various activities in strict accordance with the linear way, from the top down, they join each other. The gradual transition of fixed order, shaped like a waterfall, the whereabouts of step by step, the development process from one phase of "flow" to the next stage, which is the origin of the name "waterfall model". Although the waterfall model requires that each stage must be carefully validated, and with people the most easy to grasp and can skillfully use the "linear" thinking method. However, this model of linear process too idealistic, is no longer suitable for modern complex changeable software development model, currently has almost been eliminated by the industry. Its main disadvantage is that:

- Division of the various stages of the modes is completely fixed, will be generated between a large number of documents in the various stages, this greatly increases the workload of the development of the software.
- The development model is linear; users have to wait until the end of the whole process to see development efforts, thereby increasing the risk of the development.
- Due to the characteristics of the development model, resulting in a lack of feedback between the various developments stages of the software project, a lot of early errors were found only in the testing phase of the project life cycle can, in turn may lead to the occurrence of software crisis.

C. *The model of software development based on the rapid prototype model*

The core idea is based on the rapid prototyping model of software development mode as much as possible to quickly build out to meet the real needs of users of the software prototype. While in the process of creating a software prototype model, the internal structure of the prototype system is not important, it is important must quickly establish and modify the prototype in real time to reflect and meet the needs of customers. Specifically:

- The construction of a rapid prototyping to achieve user interaction with the system, and then get the results of the user evaluation of the prototype to further refine the software needs to be developed based on user views. Turn it to meet the real needs of the customers through the gradual adjustment of

the prototype, and ultimately makes developers can identify and clear the real needs of the users.

- On the basis of the first step, the development of available user satisfaction of software products. Rapid prototyping method can overcome the shortcomings of the waterfall model, reducing the risk of the development to bring the software needs is not clear, and has a significant effect.

D. *The model of software development based on the spiral model*

In 1988, Barry Boehm formally issues spiral model for software system development, it waterfall model and rapid prototyping model emphasizes risk analysis ignored by other models; the development model is particularly suitable for large, complex systems development. The spiral model number of iterations along the spiral:

- Plan: identify software targets to achieve these objectives, options and constraints on the basis of.
- Risk analysis: analysis and evaluation of the selected program development strategy, efforts to eliminate all kinds of potential risks, and consider how to identify and eliminate these risks.
- The implementation of Engineering: realize the spiral cycle development target.
- The customer assessment: evaluation of the stage of development of the results, and proposed amendments, the next phase of design work.

The spiral model is in all stages of the project directly considering the technical risk, to reduce the harm of risk becomes the problem before, but there are also some disadvantages [3]. Specific as follows:

- The spiral model emphasis on risk analysis, but for many customers accept and believe this kind of analysis, and make relevant reaction is not easy. Therefore, this model often is applicable to internal large-scale software development.
- If the execution risk analysis will greatly impact on project profit, then risk analysis will be meaningless. Therefore, the spiral model is only suitable for large-scale software project development work.
- Software developers should be good at looking for possible risk, and can accurately assess the risk analysis, otherwise will lead to greater risk.

E. *The model of software development based on the Incremental Model*

Incremental model, also called evolution model. The premise is the software system is seen as a series of incremental member of set, and in the mode of software development process, in each stage submitted to the user of the software products and not a whole, but meet customer demand a subset of the operation of the product. In this way, the whole software products were decomposition became several incremental member, developers can the delivery of one [4]. Do this advantage is software development can better adapt to change, customers can continue to see the development of the software products of different part, so as

to reduce the development risk. But incremental model also has the following disadvantages:

- As a result of each component is gradually merge into the existing software system structure of, this will request the software must have an open architecture.
- In the development process, considering the change of demand is inevitable, so, the increment model to adapt to the changes of the ability to greatly is better than the waterfall model and rapid prototype model, but at the same time, is also very easy to degenerate into doing change model, so that the software process control lose integrity.

F. The model of software development based on the fountain model

Fountain model, also called object-oriented life cycle model. The shape of the fountain reflect is iterative and no clearance characteristics. The model in the project is the whole life cycle can embed child life cycle. It is of object-oriented software development method for the foundation, by the user's needs as a driving force, the object is used as a driven model. Its advantage is that it overcome that the waterfall model does not support software reuse and integrated limitations of a number of development activity.

G. The model of software development based on the Intelligent Model

Intelligent model, also called based on knowledge of software development model. It combines expert system knowledge, is knowledge engineering and the outcome of the combination of software engineering [5]. Intelligent model of the obvious characteristic is screening the code complexity, it makes maintenance work in the function level, on reasoning mechanism, under the action of through software engineering knowledge and other areas related to the generation of knowledge based rules of expert system way, improve software development efficiency, reduce the

development process of possible error, reduce the difficulty of solving this issue. This not only conforms to the people's thinking habits, also it makes the software developers which can focus more energy on specific in the expression of the problems.

H. The model of software development based on the hybrid model

Hybrid model and that process development model or meta-model, the above several different model combination becomes a hybrid model, it allows a project can follow the most effective path development.

III. CONCLUSION

In this paper, the theory of software engineering in several more classic common software development the advantages and disadvantages of the model of the system were discussed, but often in actual software project development process, every software development organizations and will be according to their current software project development category of need to choose the suitable for their software development model, and also with the current is developing specific product characteristic changes, thus reduce the selected model faults, make full use of its advantages.

REFERENCES

- [1] T.G Zang. Software engineering. Beijing: science press, 2009.
- [2] Zhong Qu, Yu Wu, Qun Liu, Bing Liu. Software engineering. Beijing: Mechanical Industry Press, 2007.
- [3] Feng Zhou,X.Y Liu, X.M Li, Dan Li. Software engineering (the Second Edition). Chongqing: Chongqing University Press, 2001.
- [4] FengYang. Software engineering theory and application. Beijing: Peking University Press.,2010.
- [5] Z.F Jia. A new university computer foundation course (the Second Edition). Beijing: China Railway Press.