

The Modeling of Dual Drive Software Based on Requirement and Runtime Variability

LiuZhigang^{1,2}, ShiLei³,Gezhen^{*4,a},WangXiaozi²

¹National Pilot School of Software,Yunnan University, Kunming 650500,China;

²Nanchang Institute of Science &Technology,Nanchang 330108,China;

³Henan Polytechnic Institute,Nanyang 473000,China;

⁴School of Information Engineering,Wuhan Technology and Business University,Wuhan 430065,China

^agezhen1983@sina.cn

Keywords: SaaS evolution model, Communication process algebra (ACP), Sensitivity analysis, Dynamic environment.

Abstract. In order to solve the operating environment of the dynamic evolution, this paper builds the internal reflection of the SaaS evolution model describing according to the needs and variability of the software dynamic evolution, the model is the formal definition by using communication process algebra (ACP) as the main form, and each piece uses ACP to describe a process item. On this basis, based on cloud computing dynamic environment, this paper studies effective prediction software requirements and change sensitivity method in software process, and taking the requirements and operational variability for dual drive model establishes PAAS software dynamic evolution platform. Finally, this paper carries out improvement for the software process, the use of the Petri net and ACP verify the improvement process, and the improvement results are evaluated and packaged.

Introduction

In the 1960s, the emergence of large capacity and high speed computer made the application scope of the computer expand rapidly, the software development industry was in rapid growth, the high-level language began to appear, the development of operating system caused changes in the applications of the computer, and a large number of data processing resulted in the birth of the first generation of database management systems [1-2]. At the same time, system software was larger and larger, the complex degree was more and more high, software reliability problem was more and more outstanding, the requirements of the system software and the complexity of software itself were sharply raised, so the software crisis began to erupt, this was the so-called "software crisis" [4,5]. In order to solve this problem, this paper applies the computer science theory and technology as well as project management principles and methods, to meet the user requirements' software product definition, development, deployment and engineering maintenance in accordance with the budget and schedule [7-9]. After decades of continuous efforts, the theory of software engineering has been greatly enriched and perfect, all kinds of software design methods are constantly emerging, software industry is booming, so as to promote the continuous development of computer science.

Internet speeds up the information to enter all aspects of the social economy and the daily life of the masses, especially the development of mobile internet [10]. Data shows that the Taobao website is more than tens of millions of transactions every day, one-day volume of data generates over 50 TB (1TB is equal to 1000 GB) and storage 40 PB (1PB is equal to 1000TB); in Baidu company, the amount of data is close to 1000 PB, the number of web pages are stored close to 1 trillion pages, about 60 billion search requests are processed every day. In short, big data exists in all walks of life, a big data era is coming, it brings new opportunities and adjustment for the dynamic evolution study of

software, and cloud computing new environment provides strong support and new ideas for solving the era of big data software dynamic evolution.

Research Status of Software Evolution

Software engineering goal is to improve the software productivity and the quality of the products. Software automation is to improve the fundamental way software productivity, but also can help to improve the quality of software. Since the 1960s, software evolution phenomenon has been subject to people's attention. With the deepening and development of software engineering disciplines, the importance and universality of software evolution are more and more strong [11]. The Lehman's law lasted for 22 years, and it concluded that the software system must be constantly evolving. The concept of software evolution has gone through the change of several times, it is from the original software maintenance to re-engineering and reconstructing and other concept, and the software evolution research has from the software static evolution to dynamic evolution. Static evolution refers to the software evolution in the shutdown state, dynamic evolution refers to the software evolution in the implementation period, and dynamic evolution is more complex and more meaningful evolution forms. The complexity of dynamic evolution is reflected in the need to deal with the problem of state transfer, and its advantages are mainly embodied in the continuous availability. So, with the passage of time, we are more and more requirements for the software dynamic evolution, and software evolution has become more dynamic. Under the new situation, software evolution continues to move in the direction of dynamic, it manifests that dynamic evolution is the basic characteristics of internet software, and dynamic evolution technology is foundation of grid computing. The pursuit of dynamic evolution ability is the purpose of autonomic computing, and the Internet needs software dynamic evolution and so on.

Research on the effective prediction dual drive model based on Software Requirement and Runtime Variability

This paper is intended to be in the background of big data. Based on cloud computing dynamic environment, it studies the effective prediction software requirements and change sensitivity method in software process, and it takes the requirements and operational variability as dual drive model, establishing PAAS software dynamic evolution platform [12]. Finally, the software process carries out improvement, the improvement process is verified by the Petri net and ACP, and the improvement results are evaluated and packaged.

Requirement change forecast and change sensitivity analysis. Based on the relationship between domestic and software runtime variability, this paper goes through the change demand forecasting and the sensitivity analysis, to establish the double actuated mode of oriented SaaS software dynamic evolution. Secondly, using the cross application of Petri nets and ACP supports the validation of software structure, properties and behavior. Finally, using the sensitivity analysis results determine whether or not to adopt the decision to change the software process, the sensitivity analysis results can be used as the basis for the organization plan, the improvement of the implementation and the tracking of the software process.

Research on double driving model of oriented SaaS software dynamic evolution demand and runtime variability. According to the needs and runtime variability of software dynamic evolution, this paper builds the internal reflection of describing SaaS evolution model, the model is formalized definition by using communication process algebra (ACP) as the main form, and each component uses a process description of ACP. On this basis, using ACP system carries out the process calculus, transformation and inference. According to the difference of operator, simple composition is regarded as the internal calculation, involving the interactive actions' parallel operator and left

parallel operator will be regarded as external interaction, using encapsulation operation is limited to a local on the part of interactive packaging.

Establishing the theoretical basis and verification method of supporting behavior. According to software dynamic evolution process produced by EPMM modeling, the software process carries out description from the perspective of process algebra. Using the communication process algebra ACP carries out expansion on EPMM as the formal derivation of software process evolution verification, and the validity and reliability of the system are studied. Based on establishing the verification theory, this paper will be cross application on Petri nets and ACP; combining both of these advantages, this paper puts forward the process algebra verification scheme based on Petri nets, to support the verification of model structure, properties and behavior in the software evolution process. Using Petri net can support the evolution process of intuitive specification and verification software. Based on the ACP theory, it supports the evolution process of software equation derivation and the behavior under the support of EPMM system.

Software process improvement. The software process improvement is set to three stages, including the understanding stage, evaluation stage and packing stages. The three stages are the cycle, and the results are used as the initial value of the next cycle when the end of the last time, to improve the software process through the continuous cycle.

Research and Feasibility Analysis of the Dual Drive Model Based on Demand and Operation under the Background of Dig Data

Under the background of big data, cloud computing is the basis for the software dynamic evolution environment, this paper studies the effective prediction software requirements and change sensitivity in the software process, to take the requirements and runtime variability for dual drive model, establishing the PAAS software dynamic evolution platform. Using Petri net and ACP verify the improvement process, the software improvement results are evaluated and packaged. The research programs and courses are in the following.

The requirement change forecast and change sensitivity analysis. At present, with the continuous evolution of software, a large number of historical data can be directly utilized including a large number of document work product in the software process, a large amounts of data demand change and a large amounts of data generated by process management. Under the open environment, the use of the Internet space has the large amount of image data, this paper carries out group notes for opening source community and software community, so they can produce a rich knowledge and wisdom big data, and these data are huge. For complex data structures, this paper intends to develop auxiliary tool identification and evaluation changing category, to carry out data mining and data prediction around the change of data; at the same time, the data precision rate and recall rate are evaluated according to the change demand of the experience heuristic prediction.

The change of sensitivity analysis mainly focuses on the change source and the affected entity. Through the change of acceptance and rejection as well as change propagation index, this paper measures the sensitivity, the sensitive degree carries out classification in accordance with the requirements, and this paper implements different change impact analysis. Based on the sensitivity degree of change demand source, change impact analysis methods are divided into four kinds, including semantic analysis method, heuristic analysis method, probability analysis method and exhaustion method, one or more of the four methods carry out the combination according to the needs. According to the structure character of the different software projects and products, the use of economic benefit analysis method measures the expected return brought by real-time changes, to determine whether or not to implement the software change process according to the requirements.

The oriented SaaS software dynamic evolution demand and time variable double driving model. According to the demand and variability of software dynamic evolution, this paper is the establishment of the external manifestation of oriented SaaS structure model, it designs the oriented SaaS structure model of supporting dynamic evolution, including static view and dynamic view. Using the static view describes the software static structure, and using the dynamic view describes the dynamic evolution in the software process. In order to maintain the mapping of the related components and the corresponding mechanism in the evolution model, this paper proposes the Petri network structure.

To establish the theoretical basis and verification method of supporting behavior verification. The basic part of the verification theory is established, including the construction of EPPM, the algebraic semantics model construction of software evolution process and the reliability and completeness of EPMM.

(1) EPMM construction. According to the characteristics of software evolution, this paper is the introduction of ACP operator and extends the EPMM. The axiom system is defined from the interleaving semantics and the sharing time point, so as to put forward EPMM system as a formal derivation system of supporting model validation.

(2) The algebraic semantics of software evolution process. The paper shows that a software evolution process model is issued together by all enable activities, in which an enable activity is described by the multiple iterations execution of synchronous synthesis task. The introduction of causal state calculates the algebraic terms of software evolution based on Petri firing rules in EPMM system, so it takes the algebra significance of software evolution process. In the angle from the interactive simulation, the equivalence of software evolution model and its algebraic representation are verified, and then the accuracy of the semantics is explained.

(3) Structural verification. Petri net is used to define software evolution process structure model, and the structure model should ensure the consistency with EPMM modeling method, grammar and constraints, whether there are isolated nodes and deadlock node in the software process model.

(4) Performance verification. Performance verification is mainly from the software process safety and research progress, to carry out validation based on reach ability graph software evolution process; security may have a state in the software evolution process; progress requires that the software can reach a certain state, namely good state can be reached.

(5) Behavior verification. In support of EPMM axiom system, software verification model is converted into an algebraic item based on algebraic semantics, and then using formula derivation method verifies the mutual simulation relationship between the process model and the protocol procedure.

The improvement of software process. Software process improvement improves the productivity of software organizations and the quality problems of software product. In order to achieve this objective, the improved software process is divided into three stages in this paper, including the understanding stage, evaluation stage and packing stages.

(1) Understanding stage. Understanding stage is mainly in order to improve the establishment of reference, to help determine the process improvement goals and strategies. Based on requirements change prediction and change sensitivity analysis results, this paper is the establishment of quantitative and objective improvement goals, and establishing software improvements base station. As the specific organization and the characteristics of product project, this paper uses one or many kinds of methods in model, relationships, and features as the base station.

(2) Assessment stage. The objective of the evaluation phase is to implement and evaluate the improvement of the software process, including the introduction of process and the change of the evaluation process. The introduction of process changes generally includes the technology, personnel, activities, resource management methods and other changes. In order to verify the introduction of alteration stage whether to reach the understanding stage setting a goal, it is necessary to evaluate the software change by the use of evaluation process, and it is necessary to determine the change whether to reach the process improvement goals and realize the extent.

(3) Packing stage. The packaged content in general is divided into two, in which one is that software process reaches satisfactory improvement effect through the evaluation, using the model draws up the improvement rules, to build a complete knowledge base and develop the supporting tools; the other is not a successful process improvement, the data can provide data reference and technical support for the next improvement process.

Summary

The combination of swarm intelligence data and software process data, they will appear big data, but also are the formation of complex data sets. For the complexity of software technology and software engineering data, this paper studies the complex data analysis methods, the method can implement the requirements and time-varying driven data application, so this paper puts forward the dual drive model of software dynamic evolution demand and variability based on SaaS. According to the needs and runtime variability of software dynamic evolution, this paper builds SaaS evolution model to describe the internal reflection, using communication process algebra (ACP) as the main form formalizes definition for the model, and each component uses ACP to describe a process. On this basis, using ACP axiom system carries out the process calculus, transformation and inference. According to the different operators, the simple composition is regarded as the internal calculation, it will involve the parallel operator and left parallel operator of interactive actions as external interaction, using encapsulation operation carries out package on the part of interactive packaging, which will be limited to a location.

Acknowledgments

The work was supported by the Post Doctoral Orientation Funding Project of Yunnan Province in 2015. The work was supported by the Project of Youth Science Foundation of National Natural Science Foundation with the project number 41505015 and the project name *Measurement Method of Atmospheric High-resolution Microwave Radiometric Based on Virtual Mirror Antenna*. The work was also supported by the scientific research project of Wuhan Technology and Business University with the project number A2015005 and the project name *Non-destructive Testing System of Porous Ceramic Structure Based on Multi-frequency Eddy Current and Machine Vision*.

References

- [1] J.W. Liu, X.J. Mao. Research on software runtime variability dynamic evolution mechanism. *Journal of computer*, 2015, 38(131): 1-18.
- [2] R. Fan, Y.Q. Peng, Y.Q. Chen et al. A formal method of adaptive software based on extension theory. *Journal of intelligent systems*, 2015, 10(6): 901-911.
- [3] H. Zhou, X.M. Wang, Z.M. Cai. A dynamic evolution framework of MPI parallel software. *Computer application and software*, 2016, 33(1): 8-10.

- [4] Y. Zhou, Y.K. Huang, Z.Q. Huang et al. A kind of open environment software online evolution consistency verification method. *Journal of software*, 2015, 26(4): 748 -759.
- [5] T. Li. *An Approach to Modeling Software Evolution Processes*. Springer-Verlag, Berlin, 2008
- [6] Lehman M M. Laws of software evolution revisited. *Proceeding of the European Workshop on Software Process Technology*. Nancy, 1996:108-124.
- [7] Lehman M M., Ramil J F, Wernick P D, et al. Metrics and laws of software evolution: The nineties view. *Proceeding of the 4Intl. Software Metrics Symposium*, Albuquerque, NM,1997.11.
- [8] Godfrey M W, German D M. The past present and future of software evolution. In: *Frontiers of Software Maintenance, FoSM 2008*.
- [9] H. Yang, M. Ward, *Successful Evolution of Software System*, Artech House, London, 2003.
- [10] S. Wang. Research on software system dynamic evolution based on component. *Yunnan University Software Institute*, 2014(12): 1-13.
- [11] C.Y. Li. Research on software dynamic evolution based on system architecture. *Zhejiang University School of computer science and technology*, 2015: 1-10.
- [12] C.Y. Li, P.J. He, et al. *Software dynamic evolution technology*. Peking University press, 2014: 132-140.