The Mechanism of Software Trustworthiness Growth based on Evolution

Li Shi*,
School of Computer Science and Technology, Huaibei Normal University, Huaibei, China
Email: Sally_shili@163.com

Shanlin Yang[a] and Ke Huang[b]
a. Institute of Computer Network System, School of Management, Hefei University of Technology, Hefei, China
b. School of Computer Science and Technology, Huaibei Normal University, Huaibei, China

***Abstract*—Based on the software evolution theory, this paper discusses the software trustworthiness growth process in which software trustworthiness gradually develops from low trustworthy state into high trustworthy state, presents the cycle model of software trustworthiness growth and summarizes the key factors of influencing trustworthiness growth. On this basis, the relative expansion concepts of trustworthy software are put forward, five typical characteristics of software trustworthiness are analyzed and software trustworthiness evolution system is constructed, which have great significance in correctly understanding the essence of software trustworthiness and exploring the strategy and technology of software trustworthiness growth.**

***Index Terms*—trustworthy software; software evolution; mechanism of software trustworthiness growth; software trustworthiness evolution**

I. Introduction

Software is everywhere in modern information society. However, in the wide range of application field and the dynamic evolution of application environment, increasingly large-scale software system is not always to be trustworthy, and often brings about a variety of malfunction or failure, which directly or indirectly leads to major accidents or disasters [1]. Software trustworthiness (ST) issues have become increasingly prominent.

In recent years the problem of ST has become a general consensus, and got full attention from the various sectors of society. Many government organizations, academic groups and the business communities are committed to software trustworthiness research in related fields [2-4], and have made a number of valuable academic achievements [5-12]. However, the features of the complexity, openness and evolution of software system bring about difficulties in software trustworthiness research. Among them, the continuous evolution of software brings more complexity and challenge to the complex problem of ST. The research of ST is still in the discrete and sparse state in recent years, and many of the key issues, such as the scientific understanding on ST, the correct recognition of the essential characteristics of ST, the reveal of dynamic mechanism and evolution laws of ST, remain to be further researched.

Continuing evolution of software leads to the difficulty of software trustworthiness research. The software evolution process is the unification of software process and software evolution, which subsequently described as a multi-level and multi-cycle feedback-driven system [13]. In the software evolution process, through continuous adjustment and optimization of the Forward Engineering and Reverse Engineering, the software system trustworthiness can be continuously improved and enhanced, gradually develops from untrustworthiness, low trustworthiness to high trustworthiness, which process we call the Software Trustworthiness Growth Process (STGP). However, the traditional software study is mainly concerned with the behavior of the software reliability, safety and other issues, ignoring the fact that there are different forms of trustworthiness-influencing factors at each stage of the STGP. In fact, all these factors come together to influence software trustworthiness. The simple assumption of these factors being independence or ignoring the existence of the factors will bring understanding deviation to software trustworthiness, block the improvement of software trustworthiness, and lead to the difficulties in the trustworthy software control and management. Therefore, it is necessary to put the problems of software trustworthiness growth and evolution into the trustworthy software research, so as to enrich and develop the fundamental theory of the trustworthy software research and to establish the theoretical basis for software trustworthiness evaluation and improvement.

Based on the theory of software evolution process, this paper proposes the Software Trustworthiness Growth Mechanism (STGM) including constructing STGP cycle model and analyzing the different forms of trustworthiness influencing factors in STGP. Based on the above, the extension concept of software trustworthiness and trustworthy software is put forward. Then, the typical characteristics of software trustworthiness are summarized. Finally, the dynamic Software Trustworthiness Evolution System (STES) is presented. The research results are of great help to identify and cognize the nature of software trustworthiness and grasp the dynamic incentives and evolution law of STGP, which is the key to de-

doi:10.4304/jsw.9.9.2322-2328

veloping, evaluating, optimizing and managing the trustworthy software.

II. Theoretical Background

A. The Software Evolution Process

The research on software evolution process consists of software evolution and software process. Of the two parts, the software process refers to the software life cycle involving a series of related process [14], which is the process of developing and managing a software production. On the concept of software evolution, Evelyn defines it as changes of the software system over the time [15]. Lehman et al. define software evolution as an action of System Dynamics that makes software system continuously maintained and improved [16]. Lehman's research work on software evolution lasted for nearly 40 years. In 1997, he put forward eight software evolution laws [17], wherein the Feedback Law illustrates that the software evolution process is a feedback-driven system of continuous improvement and extension with the change of circumstance and user requirement. The feedback requires that the software system must undergo corresponding evolution activity, which theoretically guarantees the ceaseless improvement and enhancement of the software trustworthiness, and provides a theoretical basis for the study of the Software Trustworthiness Growth Mechanism (STGM).

III. Mechanism of Software Trustworthiness Growth Based on Evolution

A. STGP Cycle Model

The Software Trustworthiness Growth Process (STGP) includes the Concept Link, Development Link, Application Link and Improvement Link in the paper. For the description of software trustworthiness growth process, the four-stage STGP cycle model is proposed as shown in figure 1. The Concept Link (CL) is the first stage of the cycle model, in which, the key attributes of software trustworthiness are determined according to its specific needs. The Development Link (DL) is the second stage of STGP, in which, through a series of orderly software process or software reengineering development activities, the trustworthy software products are produced to meet the specific needs, and the static evaluation of its trustworthiness is undertaken. The Application Link (AL) is the third stage, in which the main task is to collect information and seek opportunities to improve the trustworthiness during the interaction between the man-software system and software systems. The Improvement Link (IL) is the fourth stage, in which the researchers, according to the foregoing evaluation results and the untrustworthy factors, reanalyze and recover the user changing requirements, deepen the understanding of trustworthy requirements, thus to identify the existing problems and decide from which aspect to improve the trustworthiness. At this time, a STGP cycle has been completed, then, the next cycle begins.

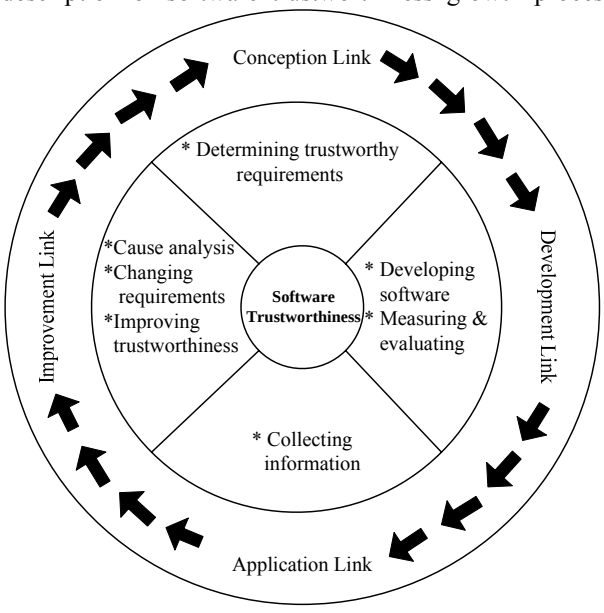

Figure 1. Software trustworthiness growth process (STGP) cycle model

B. Influencing Factors of Software Trustworthiness

Each stage in STGP has different forms of trustworthiness influencing factors. After analyzing different stages of STGP and the factors that affect the ST in each stage, we have obtained a set of trustworthy factors (STF) from the perspectives of software development and design, software operation and trustworthiness evolution etc., mainly including the following four factors, described in mathematical language as $STF_i \in STF,\ i = 1,2,3,4$.

The four factors appear in the different link of STGP respectively: The first factors, STF_1 for short, is called simply as the demand factors, including the demand information obtained in the concept link and the information of demand change got in the improvement link. The second factors, represented as STF_2 , is the process factors, which refers to the factors of software process in the development link. The third process of software running in the application link, referred as environment factors and indicated as STF_3 . The last factors is shortened over the interaction factors, expressed as STF_4 , including the factors of interactive behavior of the man-software and software-software exist in the application link of STGP cycle model. The final trustworthiness of software system is directly or indirectly affected by the four factors. The details of each factor are as follows:

(1) Demand factors - STF_1 .

The diversity of application background and the dynamic evolution of external environment lead to continuous changes in the needs of software users. Therefore, to reasonably access the users' needs and its changing information is highly important to improve the software trustworthiness.

However, because of the complexity and high requirement of trustworthy software, there are often some inadequacies using traditional physical method and model method to access the ST, such as inaccurate demand in-

formation, incomplete description of real requirements, inconsistence between demand description and demand understanding, and difficult balance of conflicting demands.

In addition, dynamic nature of the demand factors results in the evolution characteristic of software trustworthiness. For one thing, the change of the application field of the software system and the diversification of users' trustworthy requirement lead to the difference of key trustworthy attributes included in software trustworthiness. For another, in order to satisfy this kind of change and adapt to the new user requirement, some trustworthy attributes and then the whole software trustworthiness will be improved after undergoing the uninterrupted reengineering during the dynamic evolution process of software system. This situation generally causes the phenomenon that a software manifests as trustworthy in a particular period or environment, while the same one could be not trustworthy in another period or environment, or just the opposite. Therefore, the dynamic characteristics of the demand factors play an important part in the final trustworthiness of software.

(2) Process factors - STF_2 .

Process factors include two kinds of ST: Technical factors and management factors. At present, many management and technical works have run through each part of the software life cycle. Similar to technical factors, management factors have become one of the main aspects of trustworthy software process. As a set of standards for process assessments and software capability evaluation, Capability Maturity Model for Software (SW-CMM) focuses on the process management of software development and the engineering ability assessment [18]. The higher level of SW-CMM certification the development organization has, the higher its mature capacity is, and the stronger its capacity of technical innovation management and bug prevention management is, thus it is capable of guaranteeing the trustworthiness of software products. Therefore, it can be said that the process maturity of software development organizations is in proportion to its software products trustworthiness. Based on this consideration, it is necessary to determine the affecting criteria in process factors from two dimensionalities, namely, technical trustworthy and management trustworthy of software process.

(3) Environment factors - STF_3 .

The trustworthy factors of software external running environment based on the computer mainly exist in the third stage of STGP cycle model. The factors specifically involve three aspects, namely, physical environment, computer network environment and operation environment. These fundamental resources provide an important platform for software trustworthiness. To be specific, a stable physical environment is necessary in order to provide high trustworthy guarantee for software collaboration and software running. Next, it is necessary to prevent the illegal or malicious access from some users and devices when software runs in the environment based on computer, especially in the open network environment. Therefore, the trustworthiness of network environment and running environment is one of important foundations of guaranteeing software behavior trustworthy.

(4) Interaction factors - STF_4 .

When software is applied, the interactive behavior and process existing in the people-software system or software systems are often related to knowledge communication, information transformation and data transmission. In order to ensure interaction results high trustworthy, not only the behavior trustworthiness of software itself should be more effectively considered, but also the appropriate group synergy, stable dynamic logical, and reliable link delay of interactive process. Thus, the interaction factors including the effectiveness, collaboration, logic, and timeliness of interactive activities are of significance to the final trustworthiness of software.

The four factors above commonly influence the software trustworthiness from the angle of system in the STGP. Their effective relationship with software trustworthiness is as shown in figure 2. Among which, the software trustworthiness is the ultimate goal of the trustworthy software, the four types of factors serve the software trustworthiness. However, this does not mean that the factors are not important. In fact, the relationship between the final trustworthiness of software and the four factors is causal relationship. That is, if there is no trustworthiness of demand factors, process factors, environment factors and interaction factors in the STGP, it is unable to realize the software trustworthiness. Therefore, it is important and is necessary to emphasize the effect of the four factors.

C. The Extension Concept

So far, how to define software trustworthiness and trustworthy software still lack a consistent criterion. In order to come to a better understanding on the preceding basic issues of trustworthy software, to explore and enrich the basic theory research on the trustworthy software, we attempt to expound the related concept of trustworthy software from the narrow and broad senses based on the research results of software evolution process and STGP.

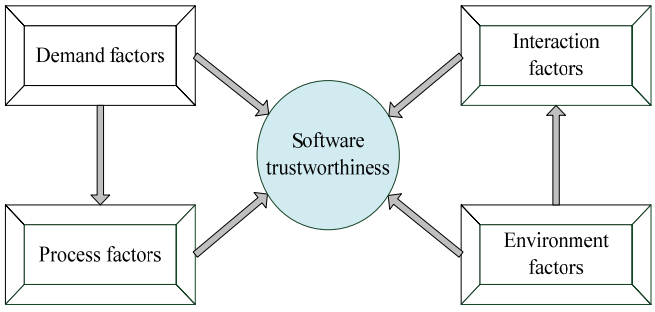

Figure 2. Four factors of influencing software trustworthiness

Definition 1. The software trustworthiness in its narrow sense refers to an integrated system property that is oriented for users' personalized and diversified needs, as a whole emergies from the trustworthy attributes such as reliability, availability, safety, integrity etc. at the special time and environment, and reflects the commitment rela-

tionship between object behavior of software and subject perception of user.

To be specific, software vendors commit themselves to achieving the software trustworthy level, just as the user requested, before the software system is developed. When the new software system is brought out, its behavior is often evaluated based on the objective data of software trustworthy attributes and the personal experience of user (or the third-party organization) , and judges whether or not the software system meets the commitment. This will become the criterion for evaluating the software trustworthiness.

The above definition of ST has the following connotation: The trustworthy attribute is associated with the specific trustworthy needs, and limited by the software type, application environment, users' characteristics, etc. Therefore, the users' needs become the key factor of leading software construction, measurement and evaluation.

Definition 2. The software trustworthiness in its broad sense refers to a comprehensive property of all trustworthy attributes that meet specific trustworthy needs in the different links of STGP. The trustworthy performance of each link has a direct impact on the ultimate trustworthiness of software system.

The broad definition of ST contains the following connotation:

(1) The software trustworthiness involves many links in software evolution process, which is not a single concept, but an aggregate concept.

(2) Each link has factors that influence software trustworthiness, and the aftereffect exists among these factors, namely, the trustworthy performance of software in every link is influenced by the trustworthy state of previous links.

Based on the narrow and broad definition of ST, the trustworthy software is defined as follows.

Definition 3. The trustworthy software refers to the software system that has good trustworthiness evolution and enhanced ability, and is able to overcome external interference. Its output behavior always conforms to the expected goal, and its operation result always meets the specific needs.

This definition includes the following two meanings:

(1) With the change of external environment and users' needs, high trustworthy software system has strong self-development capacity to achieve continuous evolution of software trustworthiness.

(2) The performance of software trustworthiness is always consistent with the expected goal, therefore, it is more emphasized that the output behavior and the results of trustworthy software are predictable and controllable.

D. The Typical Characteristics of ST

By means of the extended concept, the essence in software trustworthiness has been further cognized, and its characteristics can be found as the following:

(1) Multiple attributes. The trustworthiness is a comprehensive measurement attribute including many non-functional software attributes, such as safety, maintainability, reliability, transportability, usability, operability, etc.

(2) Dynamic evolution. With the dynamic change of software application requirements, and the software operation environment and the users' trustworthy needs, the understanding on the software trustworthiness is also affected, which results in obvious dynamic evolution of software trustworthiness. On one hand, along with the change of system application field, and application environment and the users' credibility needs, the key trusted attributes of software trustworthiness will change. On the other hand, in order to meet the change of environment and adapt to the new needs of users, some non-trusted

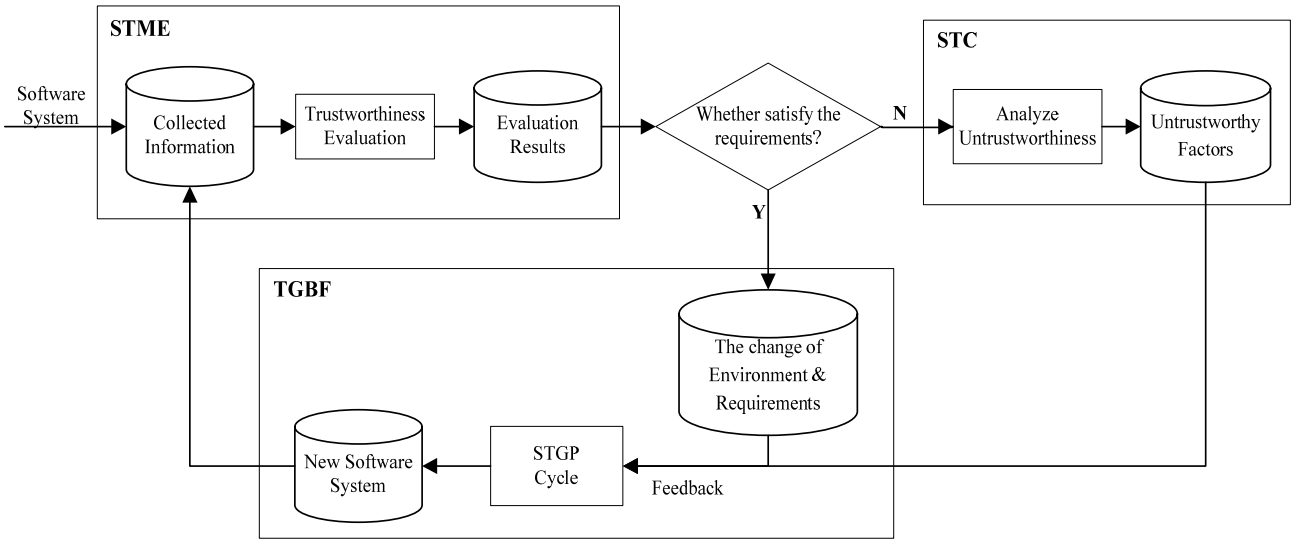

Figure 3. The software trustworthiness dynamic evolution system

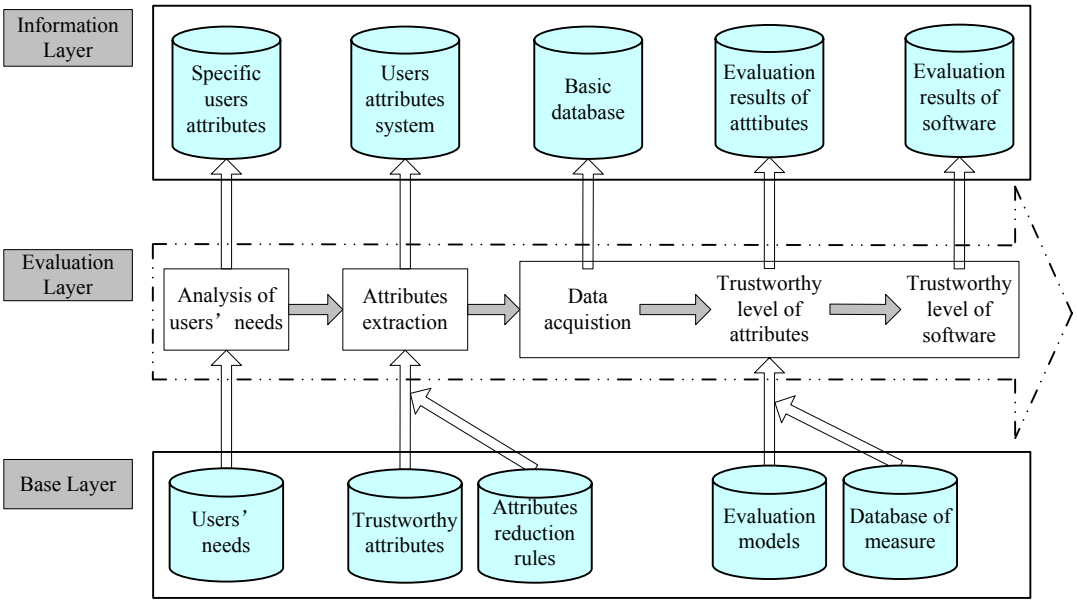

Figure 4. The Framework of Software Trustworthiness Measurement and Evaluation

attributes of the software system are improved through the dynamic evolution process of uninterrupted reengineering, and then the software-system trustworthiness is enhanced.

(3) The emergence. In the links of the software trustworthiness growth process and the dynamic evolution of various factors, the software trustworthiness emerges through a variety of the collective behavior of trustworthy attributes, which is a new, whole and unique property of the software system, and rather than the simple sum of multiple trustworthy attributes.

(4) Situation correlation. Software trustworthiness is the emergent behavior of whole software trustworthy attributes under the fixed-time section and specific application environment. Specifically, in the STGP process, the software trustworthiness depends not only on the software-development environment, operation environment and application environment, but also on the specific needs from the participants (developers, managers, organizers and individual users) at different stages. When the environment and users' needs change, the results of the software trustworthiness evaluation will produce dynamic change.

(5) The nonlinearity. In the software trustworthiness evolution process, with the software trustworthy state continuously adjusted and optimized by the Forward Engineering and Reverse Engineering, the trustworthiness of software system undergoes nonlinear change from untrustworthiness, low trustworthiness to gradually high trustworthiness, as well as from the high gradually to a decline, and then quits the stage of history in its whole life cycle. The whole process exhibits non-linear change.

The correct cognition for the essential characteristics of software trustworthiness is conducive to comprehending the dynamic change causes and the basic law of evolution, which is a key part to develop, evaluate, control and manage trustworthy software.

E. Software Trustworthiness Evolution System

To combine the software evolution process with the mechanism of software trustworthiness growth, dynamic Software Trustworthiness Evolution System (STES) is constructed, as shown in figure 3. STES mainly consists of three subsystems, namely, the Software Trustworthiness Measurement and Evaluation (STME), the Software Trustworthiness Control (STC), and the Trustworthiness Growth Based Feedback (TGBF).

Specifically, the subsystem of STME is a complicated calculation process in a three-dimensional space consisting of the evaluators, the evaluation attributes system and the software. The subsystem of STME is divided into three parts, that is, a base layer, an evaluation layer and an information layer, as shown in figure 4. Wherein the base layer containing important fundamental information such as users' needs, trustworthy attributes, attributes reduction rules, evaluation models etc., is a key part of the evaluation, and provides the data, methods and technical support for the smooth development of the software trustworthiness dynamic evaluation. Aim at some users' specific needs, the evaluation layer extracts relevant evaluation attributes from the trustworthy attributes database by using of the attributes reduction rules. And then, a certain kind of dynamic evaluation model from the attributes measurement database is adopted to complete the ST evaluation. The relevant information produced by the evaluation layer is stored in the information layer, wherein the evaluation results contain a large amount of useful trustworthy information about measurement elements, various attributes and the software trustworthiness itself. This information forms A Trustworthy Information Chain (ATIC) of the software evaluation, which not only reflects a series of the related trustworthy results of the evaluated software, but also has an important reference value to control, optimize and improve software trustworthiness.

According to the fact that whether the results of the software trustworthiness evaluation satisfy the given requirements or not, the subsystem of STC formulates the corresponding control measures. If the trustworthiness has achieved the desired objective, skip the STC link, and directly feed the users' changing demand and the changed information of the external environment back to the subsystem of TGBF to ensure the continuous evolution of software trustworthiness. If it does not meet the trustworthy requirements, the key untrustworthy factors have to be detected by means of some techniques such as sensitivity analysis, and then the detection results are feed back to the subsystem of the TGBF, and the untrustworthy factors will be further optimized by means of design and improvement.

After receiving the feedback information from the STC subsystem or from the changes of the software application environment and the users' needs, the subsystem of TGBF will produce the feedback signal of the change to act on the genetic software system. And then, based on the STGP cycle model, a series of activities contributing to the software-system trustworthiness growth will be carried out to realize the upgrade of the software trustworthiness.

To sum up, the subsystem of STME is a foundation to carry out software trustworthiness control and manage trustworthiness evolution. The subsystem of STC is an important link to improve software trustworthiness. And the subsystem of TGBF is an effective way to realize software trustworthiness growth. The three subsystems, through the iterative cycle for many times, gradually improve the trustworthiness of software system, and realize the continuous evolution of trustworthiness in its life cycle.

VI . CONCLUSIONS

In order to explore and enrich the basically theoretical research on trustworthy software, this paper, based on the theoretical research findings of the software evolution, discusses the software trustworthiness growth mechanism, presents the cycle model of software trustworthiness growth and analyzes various forms of trustworthiness influencing factors which may exist in different stages of the software trustworthiness growth process.

On this basis, the basic concepts of the research on the trustworthy software are extended, five typical characteristics of software trustworthiness are analyzed and the trustworthiness growth system is constructed, and all those lay the groundwork for such key techniques as the scientific evaluation of software trustworthiness and the control of the untrustworthy factors. The proposed system will make the trustworthy software known from the static passive response to the dynamic active monitoring, form the trustworthy control and growth theory with the characteristic of orienting software dynamic evolution, and ensure the realization of the trustworthy software dynamic evaluation mechanism based on evolution.

From the perspective of software trustworthiness evolution, the key scientific skills, such as the detection of the untrustworthy factor set in the STGP, bring us a great challenge. How to model and detect the nodal weakness and evaluate its effect on the trustworthiness of the whole system, and how to specifically design strategies to improve system trustworthiness and develop techniques to meet the needs of guidance for the activities of the specific software trustworthiness evolution will become the research focus in the future.

ACKNOWLEDGMENTS

The authors appreciate the support from Anhui Province Natural Science Foundation of colleges and universities (No. KJ2013Z281), the Youth Foundation of Huaibei Normal University (Nos. 700693 and 700708). Special thanks are given to Prof. Jinbiao Huang, and other anonymous reviewers for their constructive suggestions.

REFERENCES

[1] K. Liu, Z. G. Shan, and J. Wang etc. "Overview of basic research on trustworthy software," *National Science Foundation of China,* vol. 22, no. 3, pp. 145-151, 2008 (In Chinese).

[2] NSF Program Announcement/Solicitation: Trusted computing, *www.nsf.gov/pubs/2001/nsf01160/nsf01160.html#TOC*, Dec, 2001.

[3] Trusted Computing Group (TCG), Main Specification, Version 1.2, 2004.

[4] Department of Computer Science of the Technical University Darmstadt, Trusted Systems, *www.dvsl.informatik.tudarmstadt.de/DVSI/research/index.html*.

[5] W. Hasselbring, and R. Reussner, Toward trustworthy software systems, *IEEE computer*, vol. 39, no. 4, pp. 91-92, 2006.

[6] B. Steffen, H. Wilhelm, and P. Alexandra, et al., Trustworthy software systems: a discussion of basic concepts and terminology, *ACM SIGSOFT Software Engineering Notes archive*, vol. 31, no. 6, pp. 1-18, 2006.

[7] L. Zhu, A. Aurum, and I. Gorton et al., Tradeoff and sensitivity analysis in software architecture evaluation using analytic hierarchy process, *Software Quality Journal*, vol. 13, no. 4, pp. 357-375, 2005.

[8] W. Liu, P. Cheng, Research of Requirements Change Propagation of Complex Trustworthy Software Product Based on Interaction, *Chinese Journal of Management Science*, vol. 18, no. 4, pp. 124-132, 2010 (In Chinese).

[9] M. A. Mat Isa, J.A. Manan, and H. Hashim, et al., Trusted Real Time Operating System: Identifying its Characteristics, *2012 Interational Symposium on Computer Applications and Industrial Electronics (ISCAIE 2012)*, pp. 83-88, 2012.

[10] H. Zhang, Barbara Kitchenham, and R. Jeffery, Toward trustworthy software process models: an exploratory study on transformable process modeling, *Journal of Software: Evolution and Process*, vol. 24, no. 7, pp. 741-763, 2012.

[11] D. Wang, H.Y. Sun, and J. Wang, et al., Trusted Analysis Model for Interactive Behavior of a Software System Based on Slicing Technology, *Journal of Beijing University of Technology*, vol. 39, no.5, pp. 713-721, 2013 (In Chinese).

[12] M.X. Zhu, X.X.Luo, and X.H. Chen, et al., A non-functional requirements tradeoff model in Trustworthy Software, *Information Sciences*, vol. 191, pp. 61-75, 2012.

[13] M. M. Lehman, J.F. Ramil, Software evolution and software evolution processes, *Annals of Software Engineering*, vol. 14, no. 1-4, pp. 275-309, 2002.
[14] T. Li, *An Approach to Modelling Software Evolution Processes*. Beijing: Tsinghua University Press, 2008.
[15] J. Evelyn, Barry, and F. K. Chris, How software process automation affects software evolution: a longitudinal empirical analysis, *Journal of Software Maintenance and Evolution: Research and Practice*, vol. 19, pp. 1-31, 2007.
[16] L. Belady, M. M. Lehman, A model of large program development, *IBM System Journal*, vol. 15, no. 3, pp. 225-252, 1976.
[17] M. M. Lehman, *Laws of software evolution revisited. In: Lecture notes in computer science 1149*, Berlin: Springer, pp 108-124, 1997.
[18] X.Y.Liu, J.P.Zhao, and W.Wang et al., Software Process Management System Based on SOA, *Ordnance Industry Automation*, vol 30, no.1, pp. 91 -94, 2011.(In Chinese)

Li Shi was awarded her PhD. in Management Science and Engineering from Hefei University of Technology, Hefei, China, in 2012. She is currently a teacher in School of Computer Science and Technology of Huaibei Normal University. Her main research interests are in the areas of trustworthy software evaluation and multiple criteria decision analysis.