Theoretical Model of Software Process Improvement for CMM and CMMI based on QFD

Yonghui CAO[1,2]
1 School of Management, Zhejiang University
2 School of Economics & Management, Henan Institute of Science and Technology

Abstract
In this paper, we first introduce Software Process Improvement (SPI) and Quality Function Deployment (QFD); then study theoretical model of SPI for CMM and CMMI based on QFD. Through the research, we hope to achieve three goals: first, to develop a method, based on QFD, for the integration and prioritization of requirements from multiple perspectives; second, to map process requirements, including business requirements, to CMM or CMMI with the help of QFD; third, to be able to prioritize software process improvement actions based on process requirements. Finally, we also draw conclusions.
Keywords: *Software Process Improvement; CMM; CMMI; QFD*

1. Introduction

Software Process Improvement (SPI) has is the key to the survival of many software development organizations. Many international SPI models/standards are developed for SPI. The Capability Maturity Model (CMM) and Capability Maturity Model Integrated (CMMI) from the Software Engineering Institute are two SPI models. Like all the other standards and models on software process improvement, CMM and CMMI address the question of "what to do" while leaving "how to do it" to organizations. Therefore, some methodology is needed to transform CMM activities or CMMI Practices into a set of actions that are detailed enough to be followed by software engineers.

In this study, frameworks were developed to help map business and other process requirements of an organization to CMM and CMMI elements, and help develop action plans to satisfy those requirements using Quality Function Deployment (QFD).

QFD was first introduced in Japan by Dr. Yoji Akao in 1966. In 1972, Mitsubishi Heavy Industry put it in practice at Kobe Shipyards. Translating customers' requirements into product design requirements and relative production requirements is the most popular application of QFD. The house of quality is the focus of QFD. The customers' requirements are sometimes called customers' voice. The main idea is: quality of a product is defined by customers, not engineers. This kind of concept is mainstream of today's business practice. A frequent heard term is "injecting customers' voice into the product design". Customer requirements are often stated in non-technical or non-measurable terms. With QFD, these non-technical terms could be analyzed and converted into technical specifications. The structure of QFD is simple. The process of data analysis and converting is a complex and time-consuming one. This is often owing to the subjective nature of data itself and the potential complexities of the QFD charts.

In the traditional approach, sequential product design approach, some design defects will not be found until the final stages. To correct this kind of design defect, the design process has to start over from the early design stage. In QFD, the process requires a mufti-disciplinary team. With a mufti-disciplinary team, design defects that will result in costly prototyping and time re-design can be found and solved in the early stages of design.

QFD is not only a map for product design. It is also a map for quality improvement for current products. With the House of Quality, a design team could see how a company's product met customer requirements and what the market position of company's product regarding to "qualities" was. This will provide directions for market and quality improvement.

Currently, data mining is a hot issue. In today's computer era, every one is flooded by information. There is a great deal of information involved in designing a product. How to present correct information in the correct format becomes one of the key issues in product design. If information is presented in the wrong format, this could result in longer design time or even faulty design. QFD provides a good data-presenting format for product design. QFD is also a good format of data presentation for supporting other kinds of decision-making.

2 .SPI Framework Based on CMM Using QFD

CMM is used in the framework as the reference model because of its popularity in the industry. Although the support for CMM from SEI has discontinued and CMMI has been recommended since then, it takes time for many companies currently using CMM to switch to CMMI.

The framework is designed in such a way that the process requirements can be reflected through the proposed framework all the way down to the action plans. As a result, the priority value of each requirement is adjusted after the impacts from the other requirements are assessed.

The set of requirements with adjusted priorities are related to the key goals in CMM KPAs. The goals are prioritized based on those process requirements. Thus, the goals that achieve higher overall satisfaction of process requirements get higher importance. In order to achieve these goals, CMM has KPs categorized into five common features. Both the common features and the KPs contained in them can have different priorities. The priorities of the common features are determined by their natures in CMM. For instance, "Commitment to Perform" should be considered before "Verifying Implementation." The priorities of KPs in various common features, on the other hand, are determined by their correlations with KPA goals. Thus, the KPs in each common feature are prioritized separately based on the priorities of the goals. KPs that aim to achieve higher overall satisfaction of key goals receive higher importance values. Separate sets of action plans are derived from KPs in each of the common features. The actions that help to support more important KPs receive higher priorities.

As a result, the process requirements are reflected in KPA goals, KPs, and the actions. The actions both follow the process maturity standards in CMM and satisfy the process requirements. Those actions with higher importance values help to achieve higher process requirements satisfaction.
As illustrated in Figure 1, this framework starts with the elicitation and integration of requirements. In this phase, the requirements for the improvement of the organizational process are gathered from various branches/departments, including the business goals from the executive board. For instance, one of the business goals may state that "Our product should lead in the competition," or a software process requirement from the management level may be that "The employee productivity should be increased." Depending on which branches and departments they come from, these software process requirements are grouped into perspectives with each branch/department being a perspective.
In Figure 1, various perspectives are represented as Pl through Pn. Each perspective contains multiple requirements. The software process requirements in perspective 1 are represented as R1-1, R1-2, etc. These perspectives of software process requirements can then be prioritized based on their relative importance within the organization and integrated into one single set of requirements. In Figure1, these integrated requirements are represented as R1 through Rm, where m is the total number of software process requirements from all perspectives. The prioritization ensures that requirements from different perspectives are comparable with each other, and the integration reflects the correlations among requirements from different perspectives. The deliverable of this phase is a set of prioritized and integrated software process requirements, which serves as the input to the next phase.

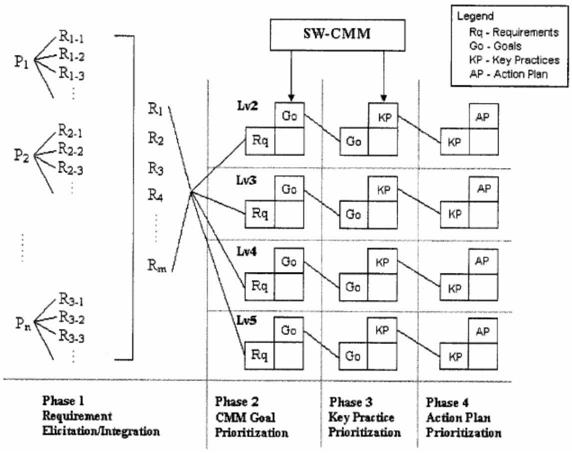

Fig. 1 Software Process Improvement through CMM Using QFD

The second through fourth phases of this framework are applied to Level 2 to Level 5 of the CMM model, The prioritized and integrated requirements from Phase 1 are linked to all KPA goals in each of the four levels in CMM using relationship matrices, These prioritized KPA goals are used as the basis for the prioritization of KPs. Finally, the prioritized KPs are transformed into prioritized action plans using House of Quality (HoQ).

In the second phase, which is "CMM goal prioritization," the goals of all KPAs in a particular CMM level are selected and prioritized based on the requirements from the previous phase. There are two objectives of this framework and this phase is significant in terms of achieving both. First, the organization needs to comply with the CMM standard. At the same time, the organization needs to ensure that by reaching a particular maturity level, the process is also satisfying the business and other requirements within the organization. In Phase 2, a relationship matrix is used to establish connections between the requirements from the organization and KPA goals in CMM. This matrix demonstrates that complying with the CMM standard also helps satisfy the business and other requirements in the organization. Second, the final set of action plans needs to be prioritized based on the priorities of requirements so that more important actions receive more resources. KPA goals serve as the bridge between requirements and the action plan. By prioritizing KPA goals, requirements from the organization can be transformed to the KPs in the third phase, and finally to the action plans in the final phase. In this way, a set of actions can be executed not only to achieve a specific maturity level in CMM, but also to satisfy organizational process requirements.

The third phase of the proposed framework, which is "key practice prioritization," involves the prioritization of KPs within all KPAs of a specific level. The prioritization is carried out on the basis of the deliverables from Phase 2. According to CMM specifications, all these KPs have to be performed in order to reach that particular maturity level. However, these KPs serve as a bridge between the requirements and the final actions, and it is necessary to know how these KPs reflect the software process requirements. In order to show the connections between the requirements and the final action plans, these KPs have to be prioritized based on KPA goals, which are now reflecting requirements priorities. The mapping between KPA goals and KPs has been provided in Appendix E of the 1995 SEI CMM book, and it can be modified if necessary.

In the fourth phase of the framework, which is "action plan development and prioritization," a set of actions is derived from the prioritized KPs. These actions should reflect the requirements integrated in the first phase. Meanwhile, they also state what needs to be executed in order to reach a particular CMM maturity level. These actions guide the process improvement. Thus, more resources should be assigned to those actions with high priorities.

The above framework addresses the problem that CMM specifies only "what to do" but not "how to do." By incorporating requirements from the organization into action plans through KPA goals and KPs, the connection between the objectives of the organization and CMM maturity levels becomes clear.

3 .SPI Framework Based on CMMI Model using QFD

3.1. SPI framework for CMMI staged model using QFD

SPI framework based on CMMI is gaining popularity in the industry. In addition, QFD is used to help with the SPI based on CMMI.

First, software process requirements, which are from multiples perspectives, are prioritized so that requirements with more and stronger impacts on other requirements can receive higher priority values. Second, business and other requirements within an organization are mapped to CMMI Process Areas and practices. Because a connection is established, the organization can clearly see how CMMI helps with its business goals. Third, in CMMI, QFD helps transform requirements of the organization into process actions through Process Areas (PAs) and Practices. Therefore, the ordering of the actions taken is based on how they are related to both the software process requirements and the corresponding Practices in CMMI.

The SPI framework for CMMI staged model, as shown in Figure 2

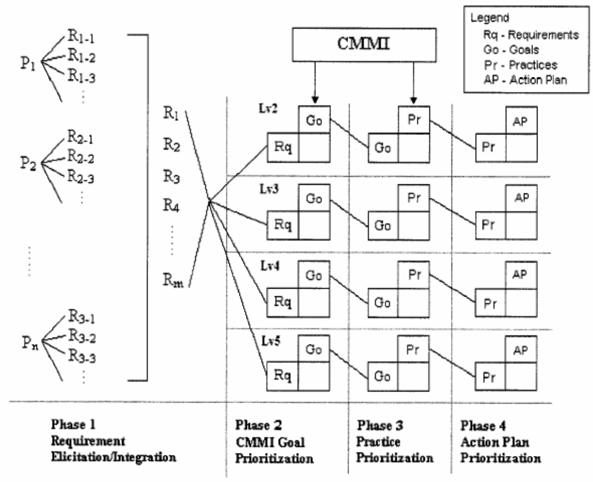

Fig. 2 Software Process Improvement through CMMI Staged Model Using QFD

For each of the four maturity levels, the set of requirements with adjusted priorities are related to the goals. The goals are prioritized based on those process requirements. Thus, the goals that achieve higher overall satisfaction of process requirements get higher importance.

In order to achieve these goals, CMMI staged model has generic practices categorized into four common features as well as the specific practices which correspond to the "Activities Performed" common feature in CMM. The priorities of Practices are determined by their correlations with goals. Thus, the generic practices in each common feature and the specific practices are prioritized separately based on the priorities of the goals. Practices that aim to achieve higher overall satisfaction of goals receive higher importance values. Separate sets of action plans are derived from the generic practices in each of the common features as well as from the specific practices. The actions that help to support more important Practices receive higher priorities.

As a result, the process requirements are reflected in PA goals, Practices, and the actions. The actions both follow the process maturity standards in CMMI staged model and satisfy the process requirements. Those actions with higher importance values help to achieve higher process requirements satisfaction.

Because of the close resemblance between CMMI staged model and CMM, the four phases for the SPI framework based on CMMI staged model as shown in Figure 2.

In Figure 2, phase 1 is exactly the same with the SPI framework based on CMM. Various perspectives are represented as P1 through Pn. Each perspective contains multiple requirements. The software process requirements in perspective 1 are represented as R1-1, R1-2, etc. These perspectives of software process requirements can then be prioritized based on their relative importance within the organization and integrated into one single set of requirements. In Figure 2, these integrated requirements are represented as R1 through Rm, where m is the total number of software process requirements from all perspectives. The prioritization ensures that requirements from different perspectives are comparable with each other, and the integration reflects the correlations among requirements from different perspectives. The deliverable of this phase is a set of prioritized and integrated software process requirements, which serves as the input to the next phase.

The second through fourth phases of this framework are applied to Level 2 to Level 5 of the CMMI staged model. The prioritized and integrated requirements from Phase 1 are linked to all goals in each of the four levels in CMMI staged model using relationship matrices. These prioritized goals are used as the basis for the prioritization of Practices. Finally, the prioritized Practices are transformed into prioritized action plans using House of Quality (HoQ).

In the second phase, which is "CMMI goal prioritization," the goals of all PAs in a particular maturity level are selected and prioritized based on the requirements from the previous phase. This phase helps to achieve two important objectives. First, the organization needs to comply with the CMMI standard. At the same time, the organization needs to ensure that by reaching a particular maturity level, the process is also satisfying the business and other requirements within the organization. In Phase 2, a relationship matrix is used to establish connections between the requirements from the organization and the goals in CMMI. This matrix demonstrates that complying with the CMMI standard also helps satisfy the business and other requirements in the organization. Second, the final set of action plans needs to be prioritized based on the priorities of requirements so that more important actions receive more resources. The goals serve as the bridge between requirements and the action plan. By prioritizing the goals, requirements from the organization can be transformed to the Practices in the third phase, and finally to the action plans in the final phase. In this way, a set of actions can be executed not only to achieve a specific maturity level in CMMI, but also to satisfy organizational process requirements.

The third phase of the framework, which is "practice prioritization," involves the prioritization of Practices within all PAs of a specific level. The prioritization is carried out on the basis of the deliverables from Phase 2. According to CMMI specifications, all these Practices have to be performed in order to reach that particular maturity level. These Practices serve as a bridge between the requirements and the final actions, and it is necessary to know how these Practices reflect the software process requirements. In order to show the connections between the requirements and the final action plans, these practices have to be prioritized based on the goals, which are now reflecting requirements priorities. The mapping between the goals and Practices has been has been clearly shown in CMMI documentation.

The fourth phase of the framework is "action plan development and prioritization". A set of actions is derived from the prioritized Practices. These actions should reflect the requirements integrated in the first phase. At the same time, they also state what needs to be executed in order to reach a particular CMMI maturity level. These actions guide the process improvement. Thus, more resources should be assigned to those actions with high priorities.

As shown in the above theoretical framework, the connection between the objectives of the organization and CMMI maturity levels becomes clear, by incorporating requirements from the organization into action plans through goals and Practices.

3.2. SPI framework for CMMI continuous model using QFD

The SPI framework for CMMI continuous model differs a lot from the staged framework. However, the same techniques of correlation-based prioritization with the help of QFD are used in the framework. In the continuous model of CMMI, the capability levels are assigned to individual PAs. Different PAs can be at different capability levels.

Each PA has two types of goal: 1) generic goals and 2) specific goals. Generic goals try to institutionalize the capability levels in CMMI, with one generic goal for each level. Specific goals describe the practices that must be implemented to satisfy the process area. These goals are satisfied by including generic practices and specific practices. Figure 3 illustrates how the practices and the actions are prioritized in the SPI framework for CMMI continuous model using QFD. The process requirements are used to in the prioritization of both PAs and Practices. The first step is to calculate the priority values of PAs. Then the Practices are prioritized from both the process requirements and PAs. Depending on which PA a Practice is from, the priority value of that Practices calculates from the requirements is multiplied by the PA priority. Finally, the action priority values are calculated from the Practice priority values.

Thus, as illustrated in Figure 3, the PAs are prioritized based on those process requirements and the PAs that help achieve higher overall satisfaction of process requirements get higher importance.

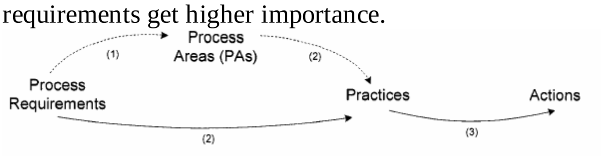

Fig. 3 Priority Calculation in SPI Framework Based on CMMI Continuous Model Using QFD

In order to make improvements on the PAs, generic practices for the generic goals and specific practices for specific goals at various capability levels are prioritized at the next phase. The priorities of Practices at different capability levels are determined by their correlations with the same set of process requirements. Because in CMMI continuous model, different PAs can have different of capability levels, the prioritization of Practices should be done for individual PAs. Thus, in this framework for CMMI continuous model, the Practices in each level of individual PAs are prioritized separately. The Practices that aim to achieve higher overall satisfaction of key goals receive higher importance values. The priority values for each PA calculated in the previous phase are used in the calculation of priorities of practices.

As a result, the process requirements are reflected in PAs, Practices, and the actions. The actions both follow the process capability standards in CMMI and satisfy the process requirements. Those actions with higher importance values help to achieve higher process requirements satisfaction.

In Figure 4, phase 1 is exactly the same with the SPI framework based on CMM. Various perspectives are represented as P1 through Pn. Each perspective contains multiple requirements. The software process requirements in perspective 1 are represented as R1-1, R1-2, etc. These perspectives of software process requirements can then be prioritized based on their relative importance within the organization and integrated into one single set of requirements.

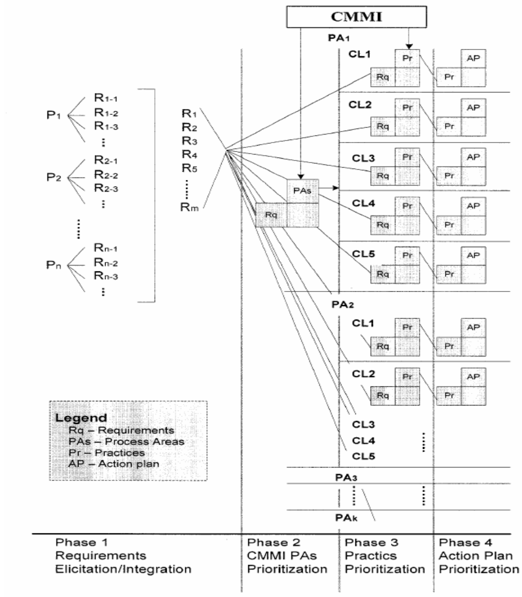

Fig. 4 Software Process Improvement through CMMI Continuous Model Using QFD

In Figure 4, these integrated requirements are represented as Rl through Rm, where m is the total number of

software process requirements from all perspectives. The prioritization ensures that requirements from different perspectives are comparable with each other, and the integration reflects the correlations among requirements from different perspectives. The deliverable of this phase is a set of prioritized and integrated software process requirements, which serves as the input to the next phase.
The second through fourth phases of this framework are applied to the PAs in the CMMI Continuous model. Because in CMMI continuous model, different capability levels are applied to different PAs, the framework for the staged model cannot be applied. Instead of mapping the prioritized and integrated requirements from Phase 1 to all the goals in a particular maturity level, they are linked to each of the PAs in Phase 2 and, depending on the target capability level, linked to each of the Practices in that level in Phase 3 using relationship matrices. In addition to the correlation values between process requirements and Practices, the priority value for each PA also participates in the calculation of the prioritization of Practices in that PA for a particular capability level. Finally, the prioritized Practices are transformed into prioritized action plans using House of Quality (HoQ).

In the second phase, which is "CMMI PA prioritization," all PAs are selected and prioritized based on the requirement priorities derived from the previous phase. This phase helps achieve two important objectives.

- The organization needs to comply with the CMMI standard. At the same time, the organization needs to ensure that by improving process areas to higher capability levels, the process is also satisfying the business and other requirements within the organization. In Phase 2, relationship matrices are used to establish connections between the requirements from the organization and each of the PAs. This matrix demonstrates that complying with the CMMI standard also helps satisfy the business and other requirements in the organization.

- The final set of action plans needs to be prioritized based on the priorities of requirements so that more important actions receive more resources. The PAs serve as the bridge between requirements and the action plan. By prioritizing the PAs, requirements from the organization can be transformed to the Practices in the third phase, and finally to the action plans in the final phase. In this way, a set of actions can be executed not only to reach higher capability levels in various PAs, but also to satisfy organizational process requirements.

The third phase of the proposed framework, which is "practice prioritization," involves the prioritization of Practices for a particular capability level within each PA. The prioritization is carried out on the basis of the deliverables from Phase 2. According to CMMI specifications, all these Practices for a capability level within a PA have to be performed in order for that PA to reach that particular capability level. However, they do not necessarily require the same amount of resources. These Practices serve as a bridge between the requirements and the final actions, and it is necessary to know how these Practices reflect the software process requirements. In order to show the connections between the requirements and the final action plans, these Practices have to be prioritized based on their correlations with requirements as well as the priority values of the Pas they belong to, which are now also reflecting requirements priorities.

In the fourth phase of the framework, which is "action plan development and prioritization," sets of actions are derived from the prioritized Practices for the desired capability levels of various PAs. These actions should reflect the requirements integrated in the first phase. Meanwhile, they also state what needs to be executed in order to reach a particular capability level of a particular PA. These actions guide the process improvement. Thus, more resources should be assigned to those actions with high priorities.
As shown in the above framework, by incorporating requirements from the organization into action plans through the goals and the Practices the connection between the objectives of the organization and PA capability levels becomes clear.

4. Conclusions

QFD is used to help an organization achieve three objectives. First, business and other requirements within an organization are mapped to CMM/CMMI goals and activities. A connection is established so that the organization can clearly see how CMM/CMMI helps with its business goals. Second, software process requirements from multiples perspectives are prioritized so that requirements with more and stronger impacts on other requirements can receive higher priority values. Third, QFD helps transform requirements of the organization into process actions through Key Process Areas (KPAs) and Key Practices (KPs) in CMM/CMMI. Therefore, the ordering of the actions taken is based on how they are related to both the software process requirements and the corresponding KPs in CMM/CMMI.

Acknowledgments

This work is financially supported by the National Natural Science Foundation of China (Project No. 90718038). Thanks for the help.

References

[1] Paulk, Mark C., Bill Curtis, Mary Beth Chrissis, Charles V. Weber. " Capability Maturity Model for Software, Version 1 .1." Technical Report. CMU/SEI-93-TR-024, ESC-TR-93-177, February, 1993.

[2] Paulk, Mark C., Charles V. Weber, Suzanne M. Garcia, Mary Beth Chrissis, Marilyn Bush. "Key Practices of the Capability Maturity Model, Version 1.1." Technical Report. CMU/SEI-93-TR-025, ESC-TR-93-178, February, 1993.

[3] Francois Coallier, "How ISO 9001 Fits Into the Software World," IEEE Software, Vol. 11，No. 1，January 1994, pp. 98-100.

[4] Paulk, Mark C., Charles V. Weber, and Mary Beth Chrissis, "The Capability Maturity Model for Software." In K. El Emam and N. H. Madhavji (eds.), Elements of Software Assessment and Improvement, IEEE CS Press, 1999.

[5] Jennifer Gremba, Chuck Myers, "The IDEAL Model: A Practical Guide for Improvement", Bridge, Issue 3, 1997.

[6] Bamberger J. 1997. Essence of the Capability Maturity Model. Computer 30(6): pp.112-114.

[7] Paulk M., Weber C., Curtis B, Chrissis M. Eds. 1995. The Capability Maturity Model: Guidelines for Improving the Software Process. Reading, MA, Addison-Wesley.

[8] Akao, Yoji, ed., Quality Function Deployment: Integrating Customer Requirements into Product Design, Cambridge, MA, Productivity Press, 1990.

[9] Liu X, Inuganti P., Veera C. 2003. An Integration Methodology for Software Quality Function deployment. Final Project Report to the Toshiba Corporation.

[10] Xiaoqing (Frank) Liu, Yan Sun, Praveen Inuganti, Chandra Sekhar Veera, and Yuji Kyoya. "A Methodology for the Tracing of Requirements in Object-Oriented Software Design Process Using Quality Function Deployment," Software Quality Professional Journal, September 2007, Volume 9, Issue 4.

[11] Akao, Yoji, Glenn H. Mazur. "Using QFD to Assure QS9000 Compliance." 4th International Symposium on Quality Function Deployment, Sydney, 1998.

[12] Zultner, R.E. "Quality Function Deployment (QFD) for Software." American Programmer, 1992.

[13] Akao Y., Hayazaki T. "Environmental Management System on ISO14000 Combined with QFD." Transactions of the Tenth Symposium on QFD. Novi, Michigan. ISBN 1-889477-10-9

[14]Ita Richardson. "Quality Function deployment-A Software Process Tool?" Third Annual International QFD Symposium. Linkoping, Sweden, Oct. 1997.

[15]Ita Richardson, Eamonn Murphy, KevinRyan, "Development of Generic Quality Function Deployment Matrix", Quality Management Journal, Vol. 9, No. 2, APRIL 2002, pp. 25-43

[16] Zultner, Richard E. "Business Process Reengineering with Quality Function Deployment: Process Innovation for Software Development." 7th Symposium on QFD (ISBNI-889477-07-9), 1995.

[17]Song, Ki-won; Kim, Jin-soo. Measurement and Management of the Level of Quality Control Process in SoC (System on Chip) Embedded Software Development, International Journal of Advanced Robotic Systems, APR 5 2012

Author Yonghui CAO received the MS degree in business management from Zhejiang Universityin 2006. He is currently a doctorate candidate in Zhejiang University. His research interest is in the areas of management information systems.