

Process metrics for system quality with specifications' shifts from a bid phase to an operation phase

Noriko Hanakawa

Information management department
Hannan University
Matsubara, Osaka, Japan
hanakawa@hannan-u.ac.jp

Masaki Obana

Department of Information Science and Technology
Osaka Institute of Technology
Hirakata, Osaka, Japan
masaki.obana@oit.ac.jp

Abstract— Recently, large-scale computer system is important for social human life. Such large-scale system is basically ordered through a competitive bidding. A development company that won the bidding starts developing. After that, customers use the computer system on their business operations. However, customers do not use some functions. Functions proposed in a system proposal in a bid phase are not same functions that customers frequently use in an operation phase. Functions change in various phases. Therefore, we propose process metrics for system quality. The feature of the metrics is target duration from a bid phase to an operation phase. Specification shifts of functions are clear in a simple rule. The metrics quantitatively measure specification shifts. We check active or non-active of each function on each phase. The total number of change of active or non-active of all functions is an important element of the metrics. Moreover, timing of the change of active or non-active of functions is also a significant factor of the metrics. As a result of applying a real project, we found that usage of the uncompleted package software makes development process complicated.

Keywords— competitive bidding; usage frequency; active or non-active function; unreasonable price and time; system proposal.

I. INTRODUCTION

Recently, large-scale computer system has been frequently developed in industry. The large-scale computer system means society infrastructure system such as public transportation system, civic service system of government, banking system, and education system of universities. The system is not only important but also essential for our modern society.

In general, such large-scale computer system has difficulty for development process including requirement analysis, design software, programming techniques, test techniques, and operation and maintenance. In addition, a bidding activity is important for development of large-scale computer system. A bid activity means that a customer organization chooses a system development company using system proposals from development companies. Customer organization selects the best system proposal, then the customer organization orders the new computer system from a development company that made a successful bid. A system proposal includes not only system functions (software and hardware) but also total price, and schedule (due day). Customer organization contracts with the development company on the basis of the prices and schedule

that are provided in the system proposal. Hence, system proposal is very important in a view of company management. If system proposal is inadequate or insufficient, implemented functions may be inadequate, and cost may be unexpectedly increased, and delivery day may be late.

In addition, it is more important to make use of new functions by customers in real business activities. A system consultant says that 64% functions of new system are not used by customers [1]. Of course, new system must run without faults and errors. However, if new system has many unnecessary functions, customer organization wastes much money and time although faults and errors do not occur. Implemented functions are based on a system proposal that made a successful bid in a bidding phase. There are deep relationships between a system proposal and frequency of new function usage.

Therefore, we propose process metrics for system quality from a bid phase to an operation phase. Especially, we focus on specification shift during system development. System specification shifts mean that functions of new system change as new system is developed. For example, Function A is proposed in a system proposal in a bid phase. However, customers realize that Function A is unnecessary when a system design phase. In contrast, Function B is not included in the system proposal. However, at an operation phase, customers desire Function B although Function B is not included design documents and total prices. System specifications are changing on each development phase. If a system proposal is perfect, specification shifts do not occur because all specification are necessary and sufficient. If a system proposal is not perfect, specification shifts frequently occur. Our proposed metrics indicates system quality based on such specification shifts.

In section 2, related work is shown. Section 3 shows the proposed metrics in process model. In section 4, the metrics and model are applied to real computer system development. Section 5 shows summary and future researches.

II. RELATED WORK

In top level process researches, Edward et al. shows a model of the early estimating/planning stages of a project (EEPS

model) [2]. Because of unclear data in requirement analysis, there were 30% budget error. Jamieson et al. gave a model for pre- and post-contract phases in agile development [3]. These researches mostly cover our research topics. However, environment surrounding development computer system continuously change. Such continuous changes have to be considered in the present day.

On the other hand, in recently, “Chojoryu” for software engineering has been proposed by Muroya [4]. Muroya also provided a method of a contract for software development, and important of customers’ activities in top level software development process. Breiner et al. also discuss requirement engineering in a bidding stage [5]. These researches claim importance of competitive bidding. However, research of competitive bidding process in software engineering just have started. Concrete research results are not described. In addition, Takano et al. show effective bidding strategy in a competitive bidding simulation [6]. Pablo et al. propose effective competitive bidding model in scoring and position probability graph [7]. Also the other researches discussed effective bidding and accuracy of cost estimation. Management fields actively study bidding way, bidding accuracy, and bidding simulation. However, these researches discuss just ways of bidding. Our research target is not bidding system. Our research target is whole development process including competitive bidding in software engineering research field.

III. PROCESS METRICS FOR SYSTEM QUALITY

A. Concept

Fig.1 shows our proposed metrics’ concept. The concept is simple. We focus on specification shifts during a system development period. The feature is the duration of a system development period. The duration includes not only development phases (analysis, design, implement, and test) but also a bid phase (system proposals) and operation phase. In operation phase, we check frequency of each function usage. In Fig.1, “O” means that a function is active. “X” means that a function is not active.

For example, "Function A" in Fig.1 is continuously active from a bid phase to operation phase. That is, customers recognized necessary of "Function A" at a bid phase. Then, developers designed "Function A", implemented "Function A", tested "Function A". Finally, customers frequently use "Function A" in the operation phase. That is the perfect process because the specification does not change. The case is the best process. In contrast, "Function E" is the worst case. Customers needed "Function E" at a bid phase. Then, developers developed "Function E". However customers did not use "Function E". This is the worst process because the specification changed in the final development stage. Although “Function E” was developed at great expense, customers did not actually need “Function E”. Cost and time for “Function E” are wasteful. Moreover, "Function C" is not a so bad case. At first, customers recognized unnecessary of "Function C". However, at the design phase, customers and developers perceived necessity of "Function C". Development of "Function C" starts from the design phase. That is no bad case because design activity is useful to detect the lack of "Function C".

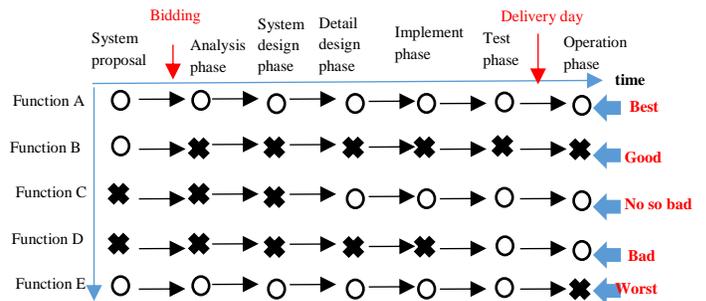


Fig. 1. A concept of process metrics for system quality

In short, our proposed metrics measure specification shifts during a period from a bid phase to an operation phase. If specification shifts do not occur, system quality is high. If specification shifts frequently occur, system quality is low. Especially, in our paper, specification shifts mean that a function’s necessity changes from active "O" to non-active "X". Or, specification shifts mean that a function’s necessity changes from non-active "X" to active "O".

B. 3.2 A proposed process basic metric $SQ(F_i)$ for each function

We propose a basic metric $SQ(F_i)$. $SQ(F_i)$ measures a process of a function “ F_i ” specification shift. A software is consists of many functions. The proposed metric $SQ(F_i)$ is as follows;

$$\text{if } A_i = \emptyset \quad SQ(F_i) = 1 \quad (1)$$

$$\text{if } A_i \neq \emptyset \quad SQ(F_i) = \prod_{k=1}^{aa} \left(\frac{nn - a_k + 1}{nn} \right) \quad (2)$$

$SQ(F_i)$: process quality of the i -th function

aa : the total number of element of set A_i

a_k : the k -th element of set A_i

$A_i = \{a \mid a \text{ is phase No. and } F_{i(n)} \neq F_{i(n+1)}\}$

$F_{i(n)}$: Mark("O" or "X") of the n -th phase of the i -th function

nn : the total number of phase

Based on Fig.2, the $SQ(F_i)$ is process quality of a function. At first, a meaning of a set “ A_i ” is explained. Elements of the set “ A_i ” are phase numbers that the current phase mark is different from the next phase mark. The phase mark means “O” or “X”, that is, “O” is the function F_i is active in the phase, “X” is the function F_i is non-active in the phase. For example, in F_2 of Fig.2, the mark of the phase P_1 is “O” although the mark of the phase P_2 is “X”. The marks of the phase P_2 or later are “X”. Then, the set A_i is $\{2\}$. Moreover, in the case of F_{20} of Fig.2, there are twice changes the marks. The mark “X” of phase P_2 changes to the mark “O” of phase P_3 , the mark “O” of phase P_9 changes to the mark “X” of phase P_{10} . Then, the set A is $\{3,10\}$.

Next, we explain equation (1) and equation (2). In the equation (1), a value of $SQ(F_i)$ is 1 if the set A_i is an empty set. That is, all marks are same like F_1 of Fig.2. The value of $SQ(F_i)$ is maximum. In equation (2), a value of $SQ(F_i)$ is calculated. The values of $SQ(F_i)$ is basically more than 0 and less than 1. The value of $SQ(F_i)$ is influenced the position of marks’ changes. If a mark changes at early phase like F_2 of Fig.2, the value of $SQ(F_i)$ is large. If a mark changes at last phase like F_4 of Fig.2,

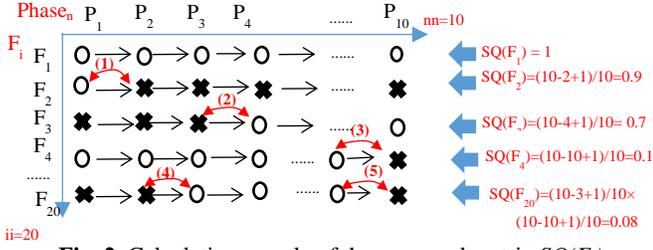


Fig. 2. Calculation sample of the proposed metric $SQ(F_i)$

the value of $SQ(F_i)$ is small. In addition, if there are several mark changes like F_{20} of Fig.2, the value of $SQ(F_i)$ is smaller and smaller. The meaning of the calculation of equation (2) is that the mark's change at the early stage of development is not so bad, the mark's change at the last stage of development is bad, frequent changes is worse than once mark's change at the last stage.

C. Features of the proposed metric $SQ(F_i)$

A most feature of the proposed metrics is the first phase and the last phase of Fig.2. The first phase of Fig.2 means a bid phase. The functions of the first phase are described in a system proposal. The system proposal is written by a bid company. The system proposal includes system development price and development time based on the described functions [9]. Customers usually contract with the system development company based on the prices and the time. The functions of the first phase is very important at the view of business management. On the other hand, the last phase is an operation phase. An operation phase usually does not includes a development process. However, the operation phase is important. Basically, frequencies of new functions are counted by operational user logs. If customers frequently use the new functions, the marks at the last phase will be "O". If customers do not use the new functions, the marks at the last phase will be "X". Of course, trial use or test use of the new function is eliminated from the frequencies. The new functions have to be useful for customers' business processes.

The first phase mark and the last phase mark are very important. Fig.3 shows a summary of process pattern on the first phase mark and the last phase mark. A development process is good when a mark of the first phase is same to a mark of the last phase. A good system proposal may lead to a good process. That is, a function is proposed in a system proposal, then, the function is implemented in development phases. At last customers frequently use the function. The proposed function in the system proposal is valuable. However, if a mark of the first phase is different from a mark of the last phase, the development process has various problems. When a mark of the first phase is "O" although a mark of the last phase is "X", customers did not need the function although the function is proposed. The cost of the unnecessary function is added to the development price in the system proposal. Customers wasted the cost of the function. In contrast, when a mark of the first phase is "X" although a mark of the last phase is "O", customers need the function although the function is not proposed. In this case, customers notice the necessity of the function halfway through their development. The cost of the necessary function is not included the development price in the system proposal. Therefore, customers

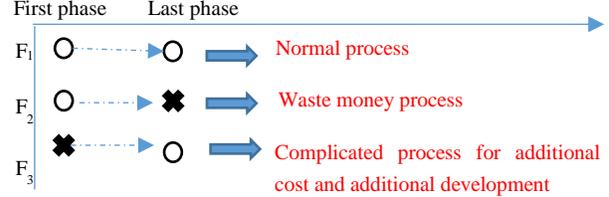


Fig. 3. Various process patterns on the first and the last phase

need additional budget for the new function. In addition, developers have to review influence of adding the new function to the original functions. The development process may be complicated, customers may raise money in order to add the function.

In this way, the proposed metric have a potential of significant problems such as process complexity and development cost's increment.

D. System metrics for all functions

SQ_{avg} is an average value of all $SQ(F_i)$. Of course, the best value of SQ_{avg} is 1, worst value of SQ_{avg} is extremely near zero.

$$SQ_{avg} = \text{AVG}(SQ(F_1), SQ(F_2), \dots, SQ(F_{ii})) \quad (3)$$

ii: the total number of function

$$SQ_{normal} = \frac{\text{the number of function with normal process}}{\text{the total number of function}} \quad (4)$$

SQ_{normal} means percentage of the number of functions with normal process in the total number of functions. Normal process means that all marks are "O". That is, the function was included in the system proposal, then the function was implemented. At last, customers frequently use the function on business operation. Of course, the best value of SQ_{normal} is 1, the worst value of SQ_{normal} is 0. SQ_{avg} and SQ_{normal} may be one of guides of system quality from a bid phase to an operation phase.

IV. APPLICATION TO A REAL PROJECT

A. Target project

The target project is an educational system development project. The system request was provided at November of 2015, the competitive bidding was held at April of 2016, starting design phase was at May of 2016, starting implement phase was at September of 2017, and starting operation was at April of 2017. Therefore, the number of phase (the value of nn_i of formula (2)) is 4. The system had 82 functions, then the number of function (the value of ii of formula (3)) is 82.

B. Checking active functions and non-active functions

Judgement of active or non-active is based on objective data such as development documents and operation log. In the system proposal phase, we checked the system proposal documents. If a function was described in the system proposal, the function is active. If a function was not described in the system proposal, the function is non-active. As same way, we checked all design documents, and we confirmed the all implementations. In the operation phase, user operation logs of the system is helpful. The

user operation log is recorded customers' operations. A huge amount of operation logs have been accumulated. If any users used a function, the function is active. No one used a function, the function is non-active.

C. Patterns of active "O" and non-active "X"

Fig.4 shows the results of active "O" and non-active "X". There are 7 patterns of the active and non-active on the target project. Pattern A (OOOO) is normal and the best process pattern. The number of functions with Pattern A is 36. Pattern B (OOOX) is a waste process because users do not use functions although the functions were developed consuming cost and time. The number of function with Pattern B is 12. Pattern C (OOXX) is a bad process. On the way of implement phase, developers confirmed unnecessary of the functions. Pattern D (OXXX) is no good process. However, the Pattern D is better than the Pattern C because developers knew unnecessary of the functions at the design phase. Pattern E (XOOO) is a problematic process in cost management. System proposal was not included the functions, however, the functions were developed in the development phase. The cost was not included in the system proposal. Pattern F (XXOO) is a troubled process because developers and customers confirmed unnecessary of the functions at implement phase. That is too late. Cost and time may be in serious trouble. Pattern G is the worst process. No one knows necessity of the functions until system running. Obviously, cost and time are shorted in the Pattern G.

D. Measuring metrics

We measured the value of $SQ(F_i)$ in equation (1) and (2), SQ_{avg} in equation (3), SQ_{normal} in equation (4). Fig.5 shows SQ_{normal} . The value of SQ_{avg} is 0.747, the value of SQ_{normal} is 0.44. In the target project, only 44% functions had the normal process "OOOO". The 56% functions had any changes between active and non-active. In short, the functions proposed in the system proposal were not necessary and sufficient. The 30% functions in the system proposal were not necessary, the 26% functions that were not proposed in the system proposal were necessary.

In this way, functions proposed in a system proposal are uncertain information. Development price and time are decided based on such uncertain information in a system proposal. Then, the price and the time often become a basic information for making a contract with a development company. That is an unreasonable procedure. This unreasonable procedure will be discussed in the following section.

V. SUMMARY

We proposed process metrics $SQ(F_i)$, SQ_{normal} , SQ_{avg} for system quality from a bid phase to an operation phase. The concept of the metrics is simple. The best case is that functions are always active in all phases. The worst case is that active functions change to non-active functions at the last phase. The metrics were applied to a real project. As a result, only 44% of functions have the best process like pattern "OOOO". 15% of functions have the worst process like pattern "OOOX". SQ_{avg} is 0.747. The 30% functions in the system proposal were not necessary, the 26% functions that were not proposed in the system proposal were necessary.

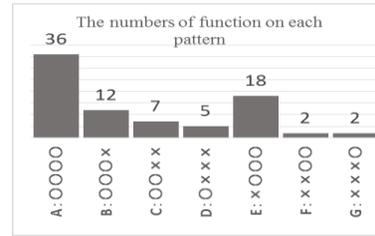


Fig. 4 The numbers of functions on each pattern

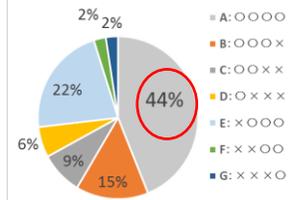


Fig. 5. A value of SQ_{normal}

In future, we will apply the metrics to several projects and several organizations. The features of projects and organizations will be clear using values of the metrics. Especially, system development including package software [8] will be discussed. Moreover, we will achieve a concrete way of estimating price and time in a system proposal [10] using values of the metrics on each organization.

ACKNOWLEDGEMENTS

This work was partially supported by JSPS KAKENHI Grant Number JP26330093.

REFERENCES

- [1] Rising sun consultation company HP <http://risingsun-system.biz/pdca-cycle-it-investment>, last accessed 2017/6/3.
- [2] J.S.Edwards, T.T.Moores, "A conflict between the use of estimating and planning tools in the management of information systems", *European Journal of Information Systems* 3(2), 1994, pp.139-147.
- [3] D. Jamieson, K. Vinsen, G. Callender, "Agile procurement: New acquisition approach to agile software development", *Proceedings of 31st EUROMICRO Conference on Software Engineering and Advanced Applications*, 2005, pp. 266-273.
- [4] IPA Software Engineering Center: A "Cho-joryu" Approach. In *The Seventeen Principles for System Development*. <http://www.ipa.go.jp/english/sec/reports/20120502.html>, last accessed 2017/6/3.
- [5] K.Breiner, M.Gillmann, A. Kalenborn, C. Müller, "Requirements Engineering in the Bidding Stage of Software Projects – A Research Preview", In: Fricker, Samuel A., Schneider, Kurt (eds.) *REFSQ 2015 21st International Working Conference, LNCS*, vol. 9013, pp.270-276, Springer (2015).
- [6] Y.Takano, N.Ishii, M. Muraki, "A sequential competitive bidding strategy considering inaccurate cost estimates", *Omega*, vol. 42, issue 1, 2014, pp.132-140.
- [7] P.Ballesteros-Pérez, M. C. González-Cruz, A. Cañavate-Grimal, "On competitive bidding: scoring and position probability graphs", *International Journal of Project Management*, vol. 31, issue 3, 2013, pp.434-448.
- [8] N.Kataoka, H.Koizumi, K.Takasaki, N.Shiratori, "Remote joint application design process using package software", In *ICOIN-12 Twelfth International Conference on Information Networking*, 1998, pp.495-500.
- [9] C.Lopez-Martin, C. Isaza, A. Chavoya, "Software development effort prediction of industrial projects applying a general regression neural network", *International journal Empirical Software Engineering*, vol.17, num.6, 2012, pp.738-756.
- [10] M. Usman, E. Mendes, F. Weidt, R. Britto, "Effort estimation in agile software development: a systematic literature review", *Proceedings of the 10th International Conference on Predictive Models in Software Engineering*, 2014.