Industrial Paper

Process Improvement Method Using a Non-model Approach

Akihiro Hayashi †and Nobuhiro Kataoka ‡

†The University of Shizuoka Institute of Science and Technology, Japan
‡Interprise Laboratory, Japan
pixysbrain@gmail.com, kataoka9@kataoka9.com

Abstract - Recently, the model approach-based process improvement method has been considered for improving the quality, cost, and delivery time (QCD) of system development projects. However, even when the capability maturity model integration (CMMI) maturity level 3 is attained, its actual effect has not yet been confirmed. On the other hand, the non-model approach identifies the process area that should be improved to achieve the specific goal of each organization's goal, and an actual effect is expected. However, the process improvement method of the non-model approach has not yet been formally established. In this paper, we propose a non-model approach method that has a real effect, such as QCD, and which is aligned to an organization's objectives. We apply this method to realize improvements in a real case process, and the results confirm the actual effectiveness with respect to the QCD.

Keywords: Non-Model Approach, Process Improvement, Defect Reports Analysis

1 INTRODUCTION

Owing to the prevalence of CMMI [1], which was developed in the United States, and the international standardization model of process assessment ISO/IEC 12207 [2] and/or 15504 [3], the process improvement for system development projects is gaining ground. A considerable number of companies are making the effort to introduce process improvement for their organizations. Consequently, the best approaches to process improvement have been widely discussed in Japan.

Process improvement is defined as the realization of a reduction for rework in system development, and an attempt to increase the QCD of system development projects.

There are two approaches to realizing actual process improvement: the model approach and non-model approach.

The model approach was introduced in CMMI, ISO/IEC 12207 and/or 15504, and aimed at achieving the conformance for the process management. The model approach includes process improvement approaches such as the CMMI's IDEAL approach and ISO/IEC 15504 ' s Geese approach.

However, the non-model approach does not adopt a process model but, instead, focuses on improving individual issues pertaining to each organization. In this approach, multiple processes are not discussed at the same time; however, unique processes are identified and improved to realize the business goal and foster the achievement of the organization. In the non-model approach, the total quality management (TQM), goal question metric (GQM), and plan-do-check-act (PDCA) approaches are known.

Model approach-based process improvement focuses on enhancing conformance to best practices such as CMMI. There are some cases that the activities do not substantially contribute to the improvement effect on business performance. On the contrary, non-model approach-based process improvements directly address points to be improved, and thus substantial improvement in performance can be expected.

However, no specific improvement approach has been established for non-model approaches. The outstanding issue rests on the nature of the non-model approach and how to proceed with procedures to establish an improvement approach.

According to the statistical information issued from CMMI Institute, 95% of companies adopting the model approach have completed improvements up to CMMI Level 3. In general, the CMMI appraisal is performed only once every three years. When an organization continues to conduct process improvements after achieving CMMI Level 3, the CMMI ' s appraisal report will not be referenced any more.

In the usual process improvement, we have conducted a very basic way of process improvement; namely, refer to the defect management reports of the development project, identify the area where the problem is concentrated, and then solve the cause of the upstream process that caused the problem in the downstream. We have confirmed some improvement through these basic activities. The motivation for this research is whether we could establish a process improvement method using a non-model approach by describing that method in a versatile manner.

Recently, there has been much focus on the model approach-based process improvement. However, approximately 30 years has passed since the software CMM Ver 1.0 was proposed, and yet there is often no real QCD improvement expected in the model approach. This is because in the model approach, the priority is sometimes on the activity to enhance conformance in order to achieve the maturity level. In this research, we aimed to improve the effect of QCD by focusing on improving performance using the non- model approach. When we applied the approach to the actual process improvement, the improvement effect was verified.

The procedure, we propose, first visualizes the magnitude of rework in each process by using the defect reports developed during the project life cycle. Next, we quantitatively identify the work products of the upstream process that have result in the largest rework. Next, we improve the process by adding the necessary practices to the process of development standard in the organization. When we apply the proposed method to an actual project and examine the effect for 3 years, the rework ratio, which was originally 51%, had improved to 42%. The organizational objective test defect ratio improved from 42% to 37%. We have got a prospect that this method is effective.

In a prior study in the field of software process improvement, Fukuyama et al. [4] reported that "what" is to improved is described in CMM, but the perspective of "how" it should be improved was missing. Then, they proposed a tool named SPIS (A Software Process Improvement Support System) that supports process improvement. However, this study focuses on the use of tools in the category of assessment.

Sakamoto et al. [5] analyzed the development process formally, and they quantified the effort of the amount of reduction achieved quantitatively, then showed the concrete benefit. Thereby, the method motivated the stakeholders of process improvement. In this study, they emphasized the improvement effects such as the reduction in the number of defects that are realized by improving the method of code review, unit tests, and joint tests on the development life cycle. However, it is not a proposal for process improvement in the sense of what improvement process can be obtained to improve the process in the non-model approach.

The authors of of this paper [6] proposed a method that identifies the process that needs to be improved by following the upstream process in accordance with the ISO/IEC 15504. The knowledge-management tools that formalize the knowledge, which was formalize-able but not formalized, originally pinpointed the process. We believe that this research is effective in that it identifies a process that includes the defects in individual organizations. However, in reality, few organizations utilize ISO/IEC 15504. Therefore, this is not a proposal for a process improvement method that can be used in general development projects.

Other studies that are related to process improvement have been reported. However, no prior studies have been reported that focus on the non-model approach, which identifies and improves processes that cause reworks by analyzing upstream defect reports ([7]- [11]).The methodologies for the non-model approaches have also been discussed as of the present; however, they are less practical compared to TQM, GQM and PDCA, which can be used in any field and not just process improvement. These methodologies do not focus on improving the QCD of the system development project. In non-model approaches, established methodologies similar to the IDEAL approach in CMMI have not yet been proposed. In this research, we propose a non-model approach-based methodology, which proves correct when applied to the actual process improvement site.

The remainder of this paper is organized as follows. In Section 2, we explain the concept and rework of the current process improvement method, after which we describe the issues to be solved. In Section 3, to solve the issues, we propose a method that visualizes the rework process by analyzing and improving the upstream defect reports. In Section 4, we explain a case study where this proposed method is applied to real-world organizations that have already achieved CMMI maturity level 3. In Section 5, we discuss the result of the case study, and we then conclude the paper in Section 6.

2 HYPOTHESES AND ISSUES OF THE REWORKED PROCESS

In this section, we present an overview of the current process-improvement activities and failure factors in the system development project. Then, we describe the hypothesis and issues to be solved using the non-model process-improvement approach.

2.1 Concept of Process Improvement Approach

Recently, model approaches such as CMMI have been used as approaches to realizing process improvements. The model approach is defined as an approach that improves the ability to manage processes by introducing the best practice model and improving the conformance to the process. As a proof of the effectiveness of the model approach, graphs are presented that show how the QCD has improved since the introduction of CMMI by organizations in accordance with level up maturity level 1-2-3 [12].

CMMI is an evaluation model of a system development project. It is the summary of the best practices used by the most excellent companies. Processes are evaluated in 5 stages. CMMI Maturity Level 3 is the level at which systematic process control is conducted. For organizations that have been improved up to Level 3, project groups under the control will manage projects using a common process consistently. Thus, the variation in quality and/or process will be less compared to Levels 1 and 2. CMMI was developed as a procurement model of the US Department of Defense. If any procurement projects occurred, the procurement section of an organization could expect that if they choose the organization that achieved CMMI Level 3 as the development partner, any project under that organization will demonstrate consistent performance. For this purpose, Level 3 is adopted as a procurement standard.

However, the model approach does not guarantee improvements in the QCD for organizations where CMMI and/or other approaches have been introduced. The graph presented as evidence that a process has improved is just one incident. More importantly, there are cases that report no specific improvement in the QCD effects for organizations in which CMMI was introduced and achieved maturity level 3 [13].

On the other hand, a non-model approach is defined as an approach that improves the ability to manage processes by improving the performance of individual processes required for each organization without using a given process as best practice. In the non-model approach, rather than improving the process from the beginning to the point at which the best practice is determined, it identifies the processes that are required to solve individual issues in each organization, and improves the processes to their ideal states.

Figure 1 shows the causes of failure and their ratio in system development projects [14]. According to this survey, 45% of the causes of failure in system development are related to

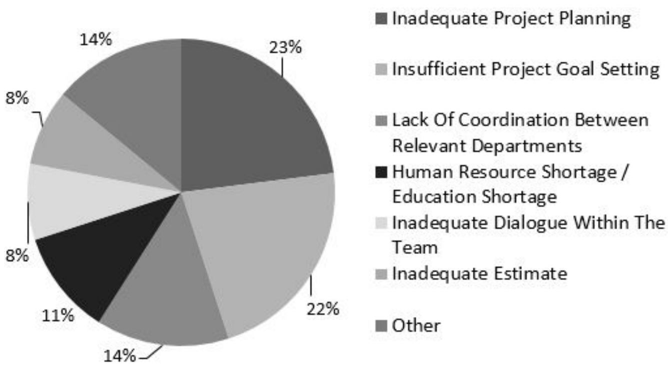

Figure 1: Causes of failure and their frequency of occurrence in system development projects

two factors: the inadequacy of the project plan and insufficient goal setting. Even if we consider other causes of failure, many of them are process problems that are caused by management. Given such a premise, in order to maximize effect of process improvement, we will make improvements with respect to the process area of "project planning," which is where most of the problems are concentrated.

However, even if the "project planning" process is improved until it has an ideal state, the QCD of the system development project will not necessarily improve. The reason for this is that the process improvement does not directly improve the QCD, but reduces the "rework" that was originally unnecessary. Then, this resulted in the improvement of the QCD.

Therefore, for process improvement that is realized using the non-model approach, we need to define the rework, and identify and improve the process that can reduce the need for the rework.

2.2 What Is a Rework?

In order to explain what is a rework,let us quote the project cost classification in system development proposed by Bill Curtis in Fig. 2 [14].

In system development, the project cost can be classified as an implementation cost and quality cost. The implementation cost is the total cost to manage a project, and includes the preparation of a project plan, progress management, reporting, and the engineering cost for manufacturing. Next, the quality cost is the cost to improve the quality of the product to be provided. Furthermore, quality costs are classified as compliance costs and noncompliance costs. The compliance cost is the sum of the costs related to evaluations, such as reviews, tests, and audits for quality-control purposes, and preventive cost such as training, procedures, tools, and data classification.

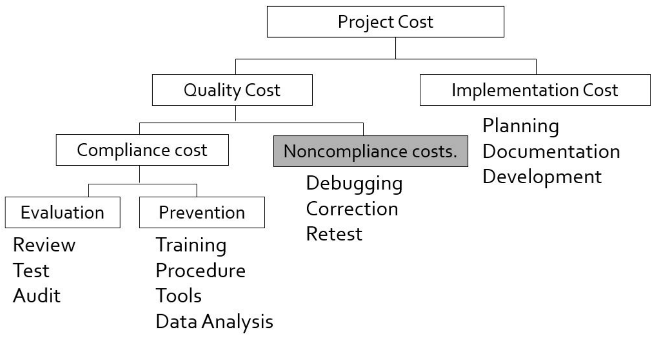

Figure 2: Classification of project costs

Meanwhile, noncompliance is the cost of debugging, correction, retesting, etc., which occur because of the cost of activities to ensure compliance. Because noncompliance costs correct defects that are noted in conmliance activities, there is no noncompliance cost if there is no noncompliance. According to Bill Curtis' research, as a result of investigating leading global companies such as IBM/TRW/NASA-SEL/HP/Raytheon, 30-40% of a project's cost is the noncompliance cost.

Even if the productivity itself is not expected to improve, the removal of this noncompliance cost by process improvement is expected to significantly improve the QCD. In this research, we define the work generated due to noncompliance in Fig. 2 as "rework."

2.3 Hypothesis of Process Improvement to Reduce Rework

A rework can be classified into two types: Noncompliance due to defects before the upstream process and noncompliance due to defects in the process. For example, when a defect is detected by a source-code review, if it was wrongly coded owing to the use of an incorrect design document, it becomes noncompliance due to a defect before the upstream process. When the correct design document is used, if it is an error in the code, it is a noncompliance that is due to a defect in the process. In order to reduce the reworks, it would be prudent to improve the upstream process that resulted in the noncompliance defect in downstream processes.

In this research, we focus on defect reports in order to reduce reworks due to defects in the upper stream process. The defect report describes defects that are detected during the process. The contents of the defect reports are mainly based on basic information such as the date, the person in charge, the defect registration number, and the target system. In addition, the defect detection process, the defect outline, the generation location, the treatment outline, and the defect incorporation process are included.

Here, if the defect-detection process and the mixing process coincide with each other, it is a noncompliance due to a defect in the current process. On the other hand, if the detection process and the mixing process do not match, it can be regarded as a defect caused by the upper stream process. The basis of process improvement is root-cause management, and improving the upstream process can result in greater benefit by improvements in the reduction of rework.

From this perspective, we hypothesized that it is possible to reduce the rework by tracing back to the upstream process that caused the defects using the defect reports; we can identify and improve the processes that are more beneficial to process improvement in the downstream process.

2.4 Issues to Be Solved

In order to establish a process-improvement method that is based on a non-model approach, the following two issues should be solved.

1. Establish procedures to identify processes that caused rework.

   The process-generating rework is different for each organization. We have not established the method to identify and improve the process of the upstream process that caused the rework in individual organizations.

2. Establish procedures that contribute to the realization of organizational objectives.

   An organization's expectation for process improvement should be to meet some organizational goals. In many cases, there may be an individual goal other than improving QCD which accompanies the reduction of rework. For this reason also, we chose the non-model approach. Procedures for process improvement to contribute to achieving such individual organizational objectives are not yet established

3 IMPLEMENTATION METHOD OF PROCESS IMPROVEMENT ACTIVITIES USING NON-MODEL APPROACH

In Section 3, to solve the issue described in Section 2.4, we propose the process improvement implementation procedure using the rework information of defect reports. The proposed method consists of five phases, the setting of organizational goals for process improvement, the visualization of the rework effort, the identification of work products in the upstream process to be improved, the root-cause analysis for embedding noncompliance, as well as additional practices to eliminate root causes.

Improvement approaches such as TQM, GQM and PDCA have been discussed even in non-model approaches. However, they are less practical to implement. The person in charge of process improvement do not know which process to choose and how to improve them; therefore, the method to be established ought not to be an abstract one but a " procedural " method. By improving the process according to the procedure, substantial performance improvement can be expected. For that purpose, it is desirable to adopt a method of naturally deriving the result by following the procedure.

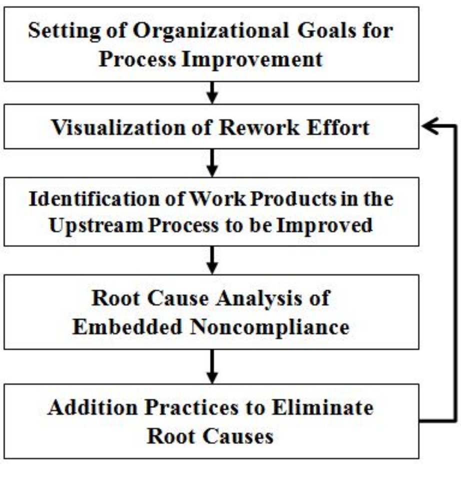

Figure 3: Implementation method for process improvement in non-model approach

In the model approach, as a result of improving the conformance to the model, the performance is expected to improve as well. However, by improving the conformance to the model, it is inevitable that unnecessary practices of that organization will be improved in the process. In this research, the proposal first identifies the process that caused the rework, and revises the work products in the upstream process that caused the defect to the process. Thus, it is possible to improve the necessary processes directly. This, however, does not mean that performance improves as a result of improved conformance as in the model approaches. It is only a very simple process improvement aimed at performance improvement.

3.1 Setting of Organizational Goals for Process Improvement

First, we set up organizational goals for process improvement. Organizational goals are to be achieved using process improvement. At the same time, an index to measure the organizational goal quantitatively should be determined.

Examples of organizational objectives include achieving a given maturity level, improving the quality and process performance objectives (QPPO), improving productivity, reducing market outflow defects, and so on.

The indexes used to measure the organizational objectives quantitatively are achievement requirements that assess whether organizational objectives have been achieved. If in the case where an organizational goal is to reduce the market outflow defects, the quantitative index should be "the ratio of the number of market outflow defects will be 10% lower than the previous year." The index should be a concrete expression such that the cycle of data collection and/or verification analysis could be carried out as instantaneously as possible.

3.2 Visualization of Rework Effort

Generally, if requirements that are related to QCD, such as the order value, quality requirement, and development time are advanced, they may influence the total project effort. Therefore, we calculate the effort ratio for each process in the whole project, and then we visualize the effort expended for each process. In this case, the effort is defined as the modified effort described in the defect reports.

3.3 Identification of Work Products in the Upstream Process to Be Improved

In the waterfall-development life cycle, work products that are created in a certain process become the input to the next process. Defects that occur in the downstream process are caused by defects in work products from the upstream process. For example, a requirement definition is considered a work product in the requirement-management process, and the requirement definition form becomes an input to the design process. The reason for the occurrence of some problems in the design process may be because of some element of trouble in the requirement-definition document.

We first pinpoint the upper stream process the defect contamination based on the amount of rework effort, which is described in Section 3.2. The work products made in each process are documented as a work product list at the time that the configuration-management plan is developed. These documents should include work products that caused defects in the downstream process. Considering the unique circumstances of each organization, work products that contain defects are identified logically.

3.4 Root Cause Analysis of Embedded Noncomliance

The reasons for which work products are incomplete are related to cases where the work products themselves were defective, where it is difficult to highlight the defects by peer review, and where the defects are not detected only by peer review. This may be the case where an intention in the upstream process is not transmitted correctly or when similar defects that had to be corrected at the same time are missed. This would be difficult to determine in the review. In such a case, we will follow the work products and processes that caused the upstream process.

3.5 Addition Practices to Eliminate Root Causes

In organizations in which process improvements are introduced, their activities are carried out according to the ”organization’s set of standard processes (OSSP)”. The OSSP is classified in the process area, and is described as the process that produces a specific kind of work product. Therefore, we can identify a process that is performed according to the result of the work product.

By identifying work products that resulted in a defect in the downstream process, we can identify the process that is performed by examining the OSSP. Therefore, we can improve the process by adding practices to eliminate the cause of the defect embedded in the work product.

The above is the proposed method employed in this research. However, even if one practice is added to a process that has already been established and operated, it is difficult to have a notable effect. More importantly, to contribute to the achievement of organizational objectives, multiple process improvement methods should be simultaneously utilized.

In this research, if defects due to the influence from upstream process account for more than 10% of the defects, the cycle of 3.2 to 3.5 for this proposed method shall be continued.

4 APPLICATION EVALUATION

In Section 4, by performing a case study where the method proposed in Section 3 is applied to a real case of process-improve- ment activities, we can evaluate the effectiveness of the proposed method.

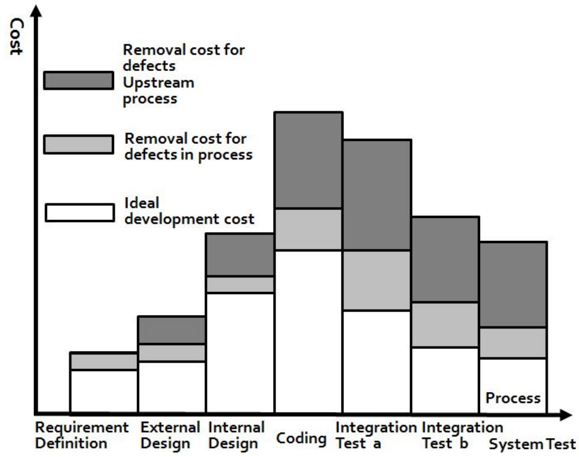

Figure 4: Development Lifecycle and Effort Ratio

4.1 The Target Case

Company A has been developing an embedded system mainly for the purpose of measurement equipment. They have been working on CMMI activities for 10 years, and achieved CMMI maturity level 3 four years ago. Although the process areas required for system development activities have been introduced in the organization, there is so far no concrete improvement in the QCD.

Company A adopts the waterfall model’s development lifecycle, which consists of a requirements definition, external design, internal design, manufacturing, integration test A, integration test B, and system test. Integration testing is the testing of interfaces between components. It distinguishes integration test A, which performs to satisfy external specifications and internal specifications. Integration test B targets external specifications at interfaces between subsystems.

Company A has chosen this proposal method rather than CMMI high maturity in order to achieve effective process improvement after achieving CMMI maturity level 3 and developing the underlying process.

4.2 1st Round

The organization goal was set as ”Left Shift by Reduction of Test Defect Ratio”. The test defect ratio defines the ratio of the number of detected defects in the test process vs. the entire life cycle upon completion of the manufacturing process. Left shift is defined as an activity to prevent them from rework in the downstream process by detecting larger defects in the upstream process.

When Company A introduced this proposed method, the test defect ratio was 42%. The index used to measure the organizational objective quantitatively is defined such that ”the Test Defect Ratio was Declined for Three Consecutive Years from the Start Point”. Figure 4 shows the development life cycle used by Company A to describe the effort on the horizontal axis and each process on the vertical axis. The effort on the horizontal axis is defined as the ratio of the effort for each process in the total effort for the project.

Company A carries out a peer review in the upstream process against the requirement definition document, external design document, internal design document, and source code document. In the downstream process, the tests are conducted in integration test A and integration test B, and the system test and logs are respectively documented. The peer-review record and the test log are included in defect reports.

We calculate the rework effort from these defect reports, and in Fig. 4, we draw noncompliance costs for defects in the upstream process as dark, incompatible costs for defects in the current process as light, and the ideal development cost as white. The sum of the noncompliance costs for defects before the upstream process and the nonconformance cost for the current process was 51%.

When summing the defect reports, 23% of the dark color area was embedded in the external design process. Company A deals with custom-made measuring equipment, products which once paid for, are used for more than 10 years, and derivative development that is related to function expansion is repeated. In the external design process, a system architecture design document, a screen design document for software, and external and internal interface design documents are made. Among them, the architectural design documents are reused, and the screen design document is already agreed with the customer, so we suspect that the defect is embedded in the interface design document.

Next, in order to analyze the root cause of the incompatibility embedded in the interface design document, we conducted investigations from three points of view: 1) Input to the interface design document, 2) completeness of the interface design document itself, and 3) the absence of a peer review of the interface design document. The incompatibility between the interface document and other classified work products is significant.

In the development by Company A, there is a sub project manager (PM) who is in charge of electricity, machinery, and software. It is necessary to simultaneously modify the interface design document under the management of the sub PM when we receive a change request from customer. However, if the analysis of the scope of the impact due to the change is insufficient in some cases, some interface design documents will be missed. This causes defects in the downstream process. In other words, it was determined that the root cause of the trouble was the change-management process.

Company A also introduced a change-management process, but often changed the source code suddenly in response to change requests, and sometimes did not revise the interface design document. There was no hand to the so-called "skewer management," which simultaneously and consistently modifies the interface design document and source code.

In the OSSP of Company A, there is a practice of impact analysis, which is done by opening a configuration control board (CCB) at the time of change management. However, since CCB is not an essential practice in CMMI, Company A arbitrarily performs CCB. Therefore, Company A incorporated practices to perform CCBs based on the judgment of the PM and sub PM when a customer's requirements changed.

In addition, as there is a possibility that the impact analysis itself is inadequate even after performing the CCB, we also added practices to number the traceability matrix from the requirement definition document to the interface design document. If the traceability matrix is assigned a number, it can also be useful for analyzing the influence range, and it is possible to prevent omissions to corrections of similar parts in the downstream process.

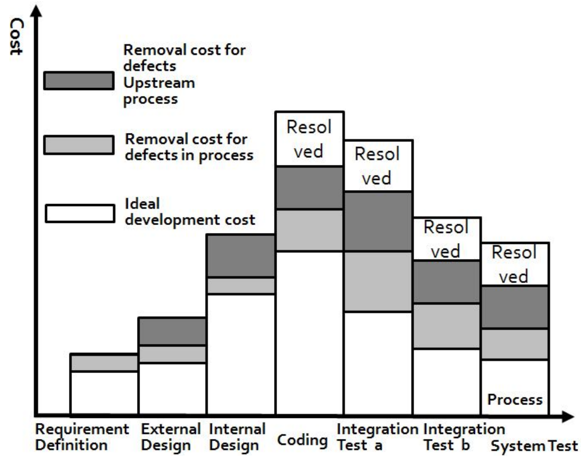

Figure 5: Development lifecycle and effort ratio (2nd)

4.3 2nd Round

Figure 5 is a graph that was created for the visualization of rework after the completion of the 1st round of process improvement proposed in this paper. Because the height of the bar graph signifies the effort ratio of each process, it is the same as in the 1st round. "Resolved" is the area that is expected to be resolved in the 1st round of the proposed method.

After we excluded the portion referred to as "resolved," when we compiled the defect reports for each process, 12% of the dark area was in the internal design process. This means that even in the second round, more than 10% of the defects were concentrated within a process. Therefore, we proceeded to the next step.

Software design documents and state-transition diagrams are primarily created in the internal design process. In the embedded development, the software design document is patterned beforehand, and defects other than careless mistakes are not often embedded. Therefore, it was inferred that there is a defect in the state-transition diagram. Based on an actual examination, defects were included in the interface between the newly developed part and the diverted part in the derivative development. When a state-transition diagram contains a defect, there should be a notification of a mistake when the state-transition diagram is developed. However, because it is difficult to complete a complicated state-transition diagram, we determined that the main reason of the defect was the absence of peer-review.

In OSSP of Company A, the implementation rate of the peer review was classified according to the criteria of the scale of work products. In this case, the peer-review implementation rate for documents such as design documents was 100%, and for transition diagrams and source codes, it was around

30%, centering on "dangerous places" based on the PM judgment.

Therefore, in the peer review of the state-transition diagram, when the reviewer reached 30% the exit criteria, they should then terminate the peer review. For this reason, the part including the defect may have deviated from the peer-review target range.

In other words, 30% of the peer-review implementation-rate was not included in the range that the PM judged as "a potentially dangerous place", and that was the root cause. Therefore, a new practice was added such that the PM and sub PM must negotiate the validity of the peer-review scope and prepare the minutes of the meeting.

There are three peer review methods at company A: the buddy method, team method, and reading-out method. The buddy method is performed by two persons, a creator and reviewer. The team method is a method in which multiple reviewers conduct an individual preliminary review. The reading-out method is a walk-through method in conference format. It does not include a preliminary review, and the producer of the work reads the review subject at the review meeting, and explains the work product in order from the beginning to the end. The peer review becomes strict in this order, and instead of requiring many attempts, the defect detection rate increases.

Company A adopts the buddy method when anything is specified, but in the case of "newly developed parts" that are in the "dangerous place", we added the practice to the review with the reading-out method.

4.4 3rd Round

Similarly, in the 3rd round, we created a graph corresponding to Fig. 5, and we visualized the classification of incompatibility using color coding. However, because the defects before the pre-process did not exceed 10%, we decided to end the cycle of this proposal.

4.5 Application Result

At Company A, a project will be launched each time a request occurs. There are relatively short-term projects with duration of about six months. After achieving CMMI maturity level 3, we applied the proposed method every year to improve the process. In this section, we evaluate the effectiveness of the proposed method by using the result of applications over the three-year period.

1. Have procedures been established to identify and improve the process that caused the rework?

   After confirming the organizational goal of process improvement, we established the process improvement procedure that first visualized the rework effort and identified the work product of the upstream process to be improved. Then, we identified the root cause of the defect by further analyzing to determine the cause of the process incompatibility, and we added the practice for process improvement. Although the proposed method focused on reworks that are due to upstream defects,

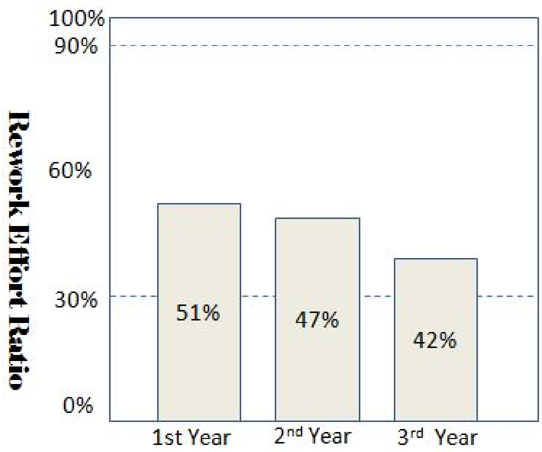

Figure 6: Rework effort ratio

   for comparison with the case where CMMI was introduced, it is necessary to include reworks due to defects of upstream processes and due to in-process defects, as in Fig. 6. The rework effort ratio, which was originally 51%, improved to 42%. Even though they achieved CMMI maturity level 3, for an organization that could not confirm the specific effect of the improvement, an improvement effect of approximately 9% was confirmed three years after introducing this proposed method.

2. Has an improvement procedure that contributed to realizing organizational objectives been established?

   In the proposed method, we set up organizational objectives at the initial stage, and at the same time, we set indices to quantitatively measure organizational objectives in order to see that process improvement contributes to the realization of organizational objectives. Figure 7 shows the trend of the test defect ratio over three years after introducing this proposed method. The test defect ratio improved from 42% to 37%. The number of projects in the first year after the introduction of this proposed method was 102, and the number of dprojects after three years was 96. Here, as the null hypothesis, we wet the "Mother proportions of two groups are equal" and tested the difference between the ratios of the two groups. If the number of projects is a trial number, the improvement from 42% to 37%, the test statistic exceeds the p value. It can be said that the confirmed improvement is a significant effect.

5 DISCUSSION

In this section, we evaluate the validity of this proposal by analyzing the result of applying the proposed method of this research to the case study of chapter 4.

5.1 Effectiveness as a Non-model Approach Procedure

In the proposed method, as shown in Fig. 3, if the improvement policy is agreed first, the rest of procedure can follow the

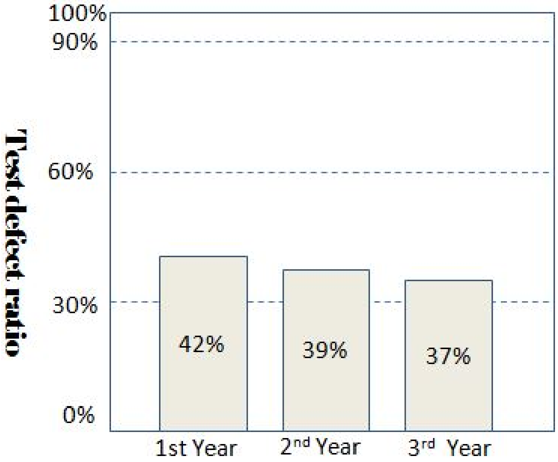

Figure 7: Test defect Ratio

improvement process according to the procedure and confirm the reproducibility.

Taking company A as an example, projects under the organization resulted in a unified form of the defect management reports in each process. By using the defect management reports, the rework efforts are then visualized as shown in Fig. 4, from which one can identify the areas with the most rework. Thereafter, by referring to the configuration management process, it is possible to uniquely identify the upstream process that caused the rework. We will carry out the peer review again and confirm the intent of the upstream process, from which one can identify the root cause the process in downstream. Finally, process improvement is achieved by adding practices to solve the root cause.

This methodology is not based on empirical results or implicit knowledge of process improvement. The same conclusion can be reproduced by tracing the upstream process according to the procedure. It is considered that the proposed method can be utilized as a non-model approach procedure.

5.2 Effectiveness as Proposed Method of Process Improvement

According to the case of company A, the rework rate was improved from 51% to 42% and the test defect ratio was improved from 42% to 37%

The rework rate improved by 9% over three years, which is only 2% lower than 40% of the leading company. Originally, CMMI was intended to be suitable for large-scale enterprises like the leading company. There has been some discussions that it may not be suitable for a small-scale enterprises such as company A. Improvement to the remaining 2% of the lower limit is thus considered acceptable.

The test defect ratio was improved by 5% over three years. This value was not expected from the beginning; it was obtained as a side effect owing to the reduction in the rework ratio. In the announcement for the SEPG conference to be held every year in North America, because there are many companies aiming for a test defect ratio of approximately 35%, the target value is again improved to 2% for average companies.

Because introducing one improvement method is a not a panacea for all problems,, both the rework ratio and the test defect ratio have to be improved to 2% up to the general target value. This may be acceptable as an improvement for the first three years.

5.3 Requirement for Introducing This Proposal Method

In order to implement the procedure of this proposed method, it is necessary that a certain degree of process has been established prior to that time. The proposed method worked effectively because it can be implemented practically, such as creating a unified defect management sheet, identifying work products created in each process, and managing work product output from each process. It is not possible to introduce anything without the foundation of a process management.

For company A, the improvement up to CMMI Level 3 was achieved, thus it can be said that the proposed method functioned effectively. However, one cannot expect that the proposed scheme will work similarly where the process is not introduced like the organization at CMMI Level 1.

If a company that achieved CMMI Level 3 desires to continue with further process improvement, it is natural for the company to aim for Level 4 or 5, which is a high-maturity level. However, if organizations are wary of improvements based on the model approach, then there is the option to introduce improvements using this proposed method, which is based on the non-model approach.

6 CONCLUSION

While many process-improvement activities are based on model approaches, in this paper, we proposed a process improvement method based on a non-model approach that appears to contribute to the achievement of each organization's business goals.

The company A that was referred to in this paper did not confirm the concrete effect while attaining the CMMI maturity level, but in the three-year period after adopting this proposed method, we confirmed the effect of realizing the goal and reduction in the number of reworks. We therefore believe that this proposed method is effective.

To further improve the proposed method, it is necessary to change some subjective judgments in the procedure to make them more objective. The defect reports are sometimes created during the project activity, the embedded upstream process defects are "surmised". Even when multiple factors causing defects are considered, it is sometimes difficult to express them accurately in defect reports.

REFERENCES

[1] SEI, CMMI for DEV Ver1.3, SEI, CMU,(2011).
[2] ISO/IEC 12207, " Information Technology - Software life cycle processes, " JIS X 0160,(2012)(in Japanese).
[3] Software Process Seminar, "Receiving the standard announcement of TR, " Japan Standards Association, In-

formation Technology Standardization Center,(2000)(in Japanese).

[4] S.Fukuyama, S.Miyamura, H.Takagi, R.Tanaka, “A Software Process Improvement Support System SPIS,” IEICE Trans. Inf. Syst.,Vol.E83-D No.4 pp.747-756,(2000)(in Japanese).

[5] K.Sakamoto, T.Tanaka, S.Kusumoto, K.Matsumoto and T.Kikuno, “An Improvement of Software Process Based on Benefit Estimation,” In the Trans of the IEICE. pp.740-748,(2000)(in Japanese).

[6] A.Hayashi and N.Kataoka, “Identification Method in Software Process Improvement Areas,” IEEJ Transactions on Electronics, Information and Systems 128(7), pp.1231-1241,(2008)(in Japanese).

[7] H.Tuan, C.Liu, and C.Chen, “Using ABC Model for Software Process Improvement: A Balanced Per-Spective,” Proceedings of the 39th Annual Hawaii International Conference on System Sciences(HICSS’06)(2006).

[8] L.Mathiassen, O.Ngwenyama, and I.Aaen, “Managing Change in Software Process Improvement,” IEEE Software,pp.84-91,(2005).

[9] D.Kozlov, “Evaluation and Improvement of the Software and Process Quality through Their Cross-Impact,” 11th IEEE Int’l Software Metrics Symposium (METRICS’05), pp.43,(2005).

[10] B.Freimut, C.Denger, and M.Ketterer, ”An Industrial Case Study of Implementing and Validating Defect Classification for Process Improvement and Quality Management, ” pp.19, 11th IEEE Int’l Software Metrics Symposium (METRICS’05),(2005).

[11] D.Caivano, “Continuous Software Process Improvement through Statistical Process Control,” Ninth European Conf on Software Maintenance and Reengineering (CSMR’05), pp.288-293,(2005).

[12] J.Herbsleb, D.Zubrow, Dennis Goldenson, Will Hayes, Mark Paulk, “Software quality and the Capability Maturity Model,” Communications of the ACM, Volume 40 Issue 6, pp.30-40,(1997).

[13] J.Herbsleb,A.Carleton, J.Rozum, J.Siegel, and D.Zubrow, “Benefits Of CMM-Based Software Process Improvement: Initial Results,” Software Engineering Institute, CMU/SEI-94-TR-13,(1994).

[14] A.Hayashi, “Process Improvement Method by Non-Model Approach,” IEICE, SWIM2016-13, vol. 116, no. 341, pp.11-18,(2016)(in Japanese).

(Received September 25, 2017)
(Revised December 26, 2017)

Akihiro Hayashi received an MBA and a Ph.D degree from the University of Tsukuba, Tokyo, Japan. His professional career includes International excellent companies: Motorola, NTT, and IBM. He has become a Professor of the University of Shizuoka Institute of Science and Technology in 2018. His research interests include Total Quality Management, Process Improvement, and Quantitative Project Management.

Nobuhiro Kataoka received a Ph.D degree from the University of Tohoku, Japan. He joined Mitsubishi Electric Corp. in 1968 and became a technology director at Information System Technology Center in 1997 and a Professor of Tokai University in 2000-2009. His research interests include Quality Control, Business Process Modeling, Business Modelization Methodologies. He is now a part-time instructor of Tokyo Denki University. He is also a member of IPSJ and a fellow of IEICE.