Development Process Visualization and Project Management

● Yuichi Arita ● Noriyasu Nakayama ● Yutaka Awata
(Manuscript received May 31, 2006)

The environment surrounding enterprises is complex and rapidly changing. To supply superior products at the optimum timing and achieve continuous evolution, enterprises must clearly visualize and manage complex product development processes by considering the quality, cost, speed, and environment. To address these needs, we have initiated two activities that focus on the level of process maturity based on Capability Maturity Model Integration (CMMI). The first activity is for standardizing new development processes with low maturity to obtain a certification of environment friendliness. The second activity is for handling existing development processes with high maturity. In this paper, we describe these two activities and introduce project management tools that support project tasks such as progress management.

1. Introduction

The need to increase the speed of product development and reduce costs is more acute than ever. To meet this need while maintaining a high level of design quality, development project management and repetitive optimization of development processes have become more important.

In conventional development, workflow errors due to insufficient sharing of development process information and communication errors due to inadequate information management often occur.

This paper introduces Fujitsu's technology for development innovation for solving problems such as workflow errors and communication errors.

2. Analyzing and solving problems

We think that projects can be classified into two types. Some projects, for example, LSI development projects, have a high maturity and a well-established design flow. On the other hand, some projects have a low maturity, for example, new projects without a well-defined process flow. We therefore started handling projects according to their process maturity by using the Capability Maturity Model Integration (CMMI) concept[1] (**Figure 1**).

The development flow of a low-maturity project (level 1 in Figure 1) is frequently changed, and as a result, the person in charge is not aware of important project details. To avoid this problem, the development flow of such projects must be established as soon as possible by adjusting the mutual interest of persons responsible for development processes and by repeating trails for increasing the number of concurrently executable tasks. Establishing such an optimal operation flow can greatly improve the development flow.

On the other hand, in projects having a high maturity (level 2 or higher), the development flow is not frequently changed, and as result, the person in charge has a much better understanding of the project. In these cases, there must be a plan for collecting execution history information for statistical analysis and evaluation of

Maturity level	Characteristics of process
Level 2 or higher	The process is planned, and its management procedures are established and observed.
Level 1	There are no systematic development/management processes.

Figure 1
Characteristics of processes in each maturity level.

development process execution results and for reviewing the process. The load on the manager and person in charge during project execution can be reduced by automating project tasks such as product management. Task omissions can be prevented by performing system checks.

In this paper, we describe how IT tools are used in the two types of problem-solving approaches described above.

In Section 3, as examples of low-maturity project handling activities, we describe some environment-handling process standardization activities. These activities are performed to efficiently build new environment evaluation/verification processes into existing development processes. Because these new processes have a low maturity, they require a highly adaptable support environment.

In Section 4, as examples of high-maturity project handling activities, we introduce some working group (WG) activities for the analysis, evaluation, and process review of a project that terminated in the repetitive optimization of a development process. In Section 5, we introduce some project management tools for supporting project operations such as product management.

3. Handling new process with low maturity

This section describes the environment evaluation process that is applied to acquire the EcoLeaf environmental label for a product.[2)]

EcoLeaf is a type III environmental labeling program that is promoted by the Japan Environmental Management Association for Industry (JEMAI). It makes quantitative product information that has been calculated using Life Cycle Assessment (LCA) publicly available.

Initially, Fujitsu's EcoLeaf environmental label acquisition activities were low-maturity activities with many unclear points. We predicted the following problems regarding these activities:

1) The evaluation process would be short and understaffed because it would be performed during the busy product development period.
2) The evaluation procedure and standards were unclear.
3) The input/output information needed for operations was unclear.
4) Many departments would be involved in these activities, making the relationships between those departments, the information types, and the information transfer timing unclear.
5) The persons in charge of product development participate in the evaluation process a maximum of three times a year, and the development process does not mature outside of these participation periods.

Needless to say, the predicted schedule of environmental label acquisition was determined, but there was not enough time to define the evaluation process and construct the appropriate support system in advance.

To solve these problems, we reviewed the evaluation process and the relationship between the evaluation process and product development process in parallel with the label acquisition activities. We also advanced the support system development so items that reduce the required person-hours more effectively were developed earlier.

Figure 2 shows an outline of the visualization system as viewed from a user when the development process has advanced considerably.

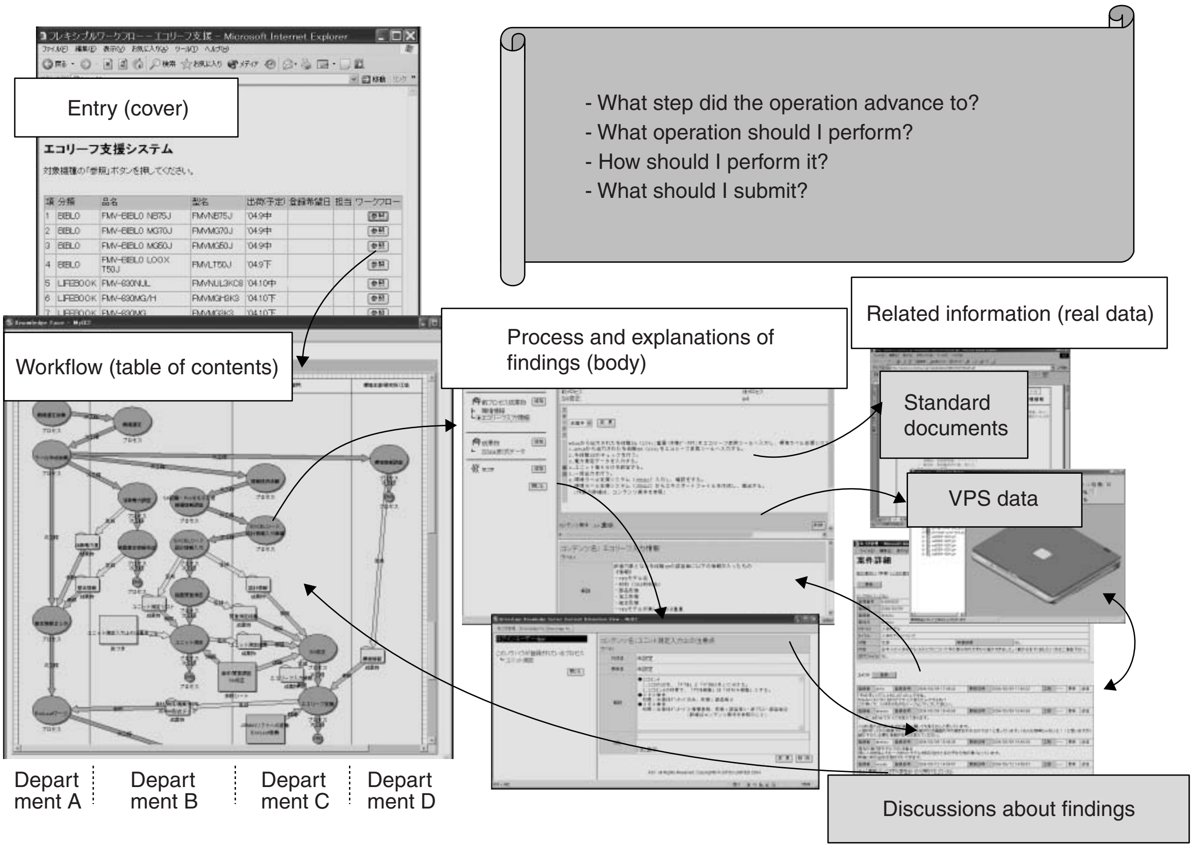

Figure 2
Overview of visualization system.

Users access this system via a Web browser and select the item they want to visualize. The visualization system supplies a function for checking and registering the process information, details of each step, current progress status, relation to other departments, related standard information, and process-associated technical knowledge and findings.

Using this function, the user can visualize the whole process and understand what needs to be done, what information is needed, who has that information, what tool to use, what standard to reference, what products to pass to the next process, and to whom the products should be passed.

We used the Virtual Products Simulator (VPS),[3),4)] which we had already used to develop virtual products in product development processes. The product data was therefore improved and used as evaluation data for obtaining the EcoLeaf label. As a result, the VPS data for design verification can be used as environmental label acquisition data without modification and therefore without an increase in work.

To support continuous growth, it is important to inherit and utilize technical knowledge. However, effective inheritance of technical knowledge is difficult to achieve. This system enables technical knowledge and findings to be associated with the processes and VPS data. This association can be made in both directions; that is, technical knowledge can be checked using process information and process information can be checked using technical knowledge.

The authors participated in the environmental evaluation activities as well as the process review and system development. When we found useful technical information during the evaluation, we registered it as technical knowledge, and if possible, improved the process and added new systems.

When one cycle of the environmental label acquisition process terminated, the accumulated technical knowledge and the new processes that were developed to improve the product were reviewed to create a template for future activities for higher efficiency and standardization.

Processes were reviewed by referencing the workflow shown in Figure 2, which is similar to an activity diagram of Unified Modeling Language (UML), and the Design Structure Matrix (DSM).[5] The mutual association of the workflow with the DSM created a new simulation function that enables effective analysis from both the work side and information side, which is not possible in a conventional system.

These activities were executed for up to 10 cycles. Compared with the first cycle of the activities, we acquired effective results; that is, the person-hours needed to acquire the EcoLeaf environmental label were reduced by a factor of from 10 to 20.

In the future, we will further improve the processes by enhancing the quantitative analysis functions.

In the above activities, we added new tools and improved the system in parallel with the process improvement. This was possible because the authors participated in the work and the system structure was appropriate for technical knowledge inheritance and flexible support of process changes.

The following explains the function and structure for creating a visualization system that effectively supports process changes.

To quickly follow process changes, it must be possible to quickly change the visualization system according to changes in the development process. To achieve this, the basic architecture of the visualization system was divided into three layers: the legacy layer, rectification layer, and visualization layer (**Figure 3**).

The legacy layer of the visualization system comprises the existing system group. Although it is unrealistic to restructure the existing IT resources at each process change, new processes and business models can be handled by arranging the necessary information on the rectification layer and by slightly changing the environment and tools on the visualization layer.

The rectification layer is used to arrange and process the information received from the legacy and visualization layers. The rectification layer is a platform for associating with the existing system and for absorbing changes. It has a function for effectively utilizing VPS data and processes the VPS data, integrates it with the information received from the legacy layer, and passes the resulting data to the visualization layer.

The visualization layer visualizes the processes, products, and technical knowledge. Because this layer is separated from the rectification layer, new functions with different data display techniques can be easily structured. In the example of acquiring the EcoLeaf environmental label, information association between systems is visualized using a process-based approach.

Needless to say, the same effective system architecture can be used when information from the new system as well as the existing system is necessary.

By devising the architecture mentioned above, we realized a visualization system that inherits technical know-how and knowledge and is very robust against process changes.

4. Handling high-maturity development processes

In a high-maturity field, the design flow and design know-how can be saved as implicit knowledge by individuals and departments. However, if the project is executed by two or more departments or the associated departments have different development processes, workflow errors and communication errors are liable to occur.

To evaluate the development process after a project is terminated, necessary information must

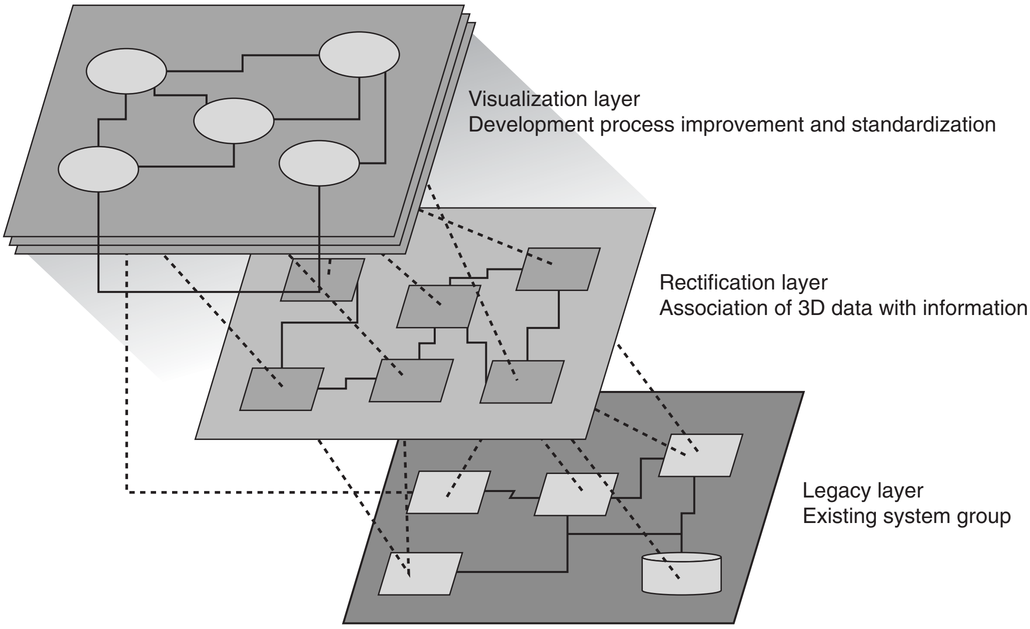

Figure 3
Visualization system consisting of three layers.

be continuously collected from the beginning of the project by using the same metrics conditions.

To solve the above problems and establish a more efficient development process, we started the Development Process Innovation WG. This WG checks the development processes of departments, creates a template of the standard development process, and creates an efficient development process from the template.

The goals of this WG are as follows:

1) Efficient project starts by using the standard development process definitions
2) Repetitive optimization by evaluation and improvement of development process at project completion
3) Uniformizing the design quality of project products
4) Evaluation, introduction, and expansion of new design tools

Because the development process depends on the type of product to be developed, it is usually necessary to form sub-WGs. Some examples of sub-WGs are:

1) The Application Specific Integrated Circuit/Field Programmable Gate Array (ASIC/FPGA) Development WG
2) The High-Speed Electrical Installation Design WG
3) The Software/Firmware Development WG

In the ASIC/FPGA Development WG, various operations were divided and assigned to individuals and later combined with each other. The operations were, for example, defining the products, including intermediate products in the development process; creating a checklist of items in each design process; creating the design-quality information items to be collected; and creating an operation standard for problem management and bug management.

The Software/Firmware Development WG operated in the same way, but used a development process in which each project could support different development techniques, for example, the waterfall[note 1] and agile techniques.[note 2] Especially,

note 1) A conventional development technique in which processes are sequentially executed.
note 2) A new development technique in which development is repeated to cover changes.

the Software Development WG already knew project management techniques such as CMMI and Project Management Body of Knowledge (PMBOK)[6] and used them aggressively.

In discussions in the sub-WG, it was sometimes found there were project management techniques common to different types of development products, for example, development process definitions that were common to hardware and software. To use common techniques such as using the standard functions of the project management tools (explained in the next section), a function addition was proposed to the Tool Enhance WG.

Development processes are created as electronic files in a format suitable for import to the project management tools. Therefore, the development processes created by the Development Process Innovation WG can be immediately applied to projects. That is, as long as the project management tools are used, the person in charge of project work will advance operations according to development processes in which conventional product development know-how is built. These development processes are the same as those requested by the person in charge of project management.

The project management tools save historical data, including project quality information. The Development Process Innovation WG analyzes terminated projects and reviews the development process definitions. The development processes are optimized continuously as shown in **Figure 4**.

The uniqueness of each department is later added by customizing the standardized development processes. If the uniqueness is effective and can also be widely used in other departments, the uniqueness should be incorporated in the standard development process.

5. Introducing project management tools

We developed a Web-application project management system that supports development process execution (**Figure 5**).

If the Development Process Innovation WG only creates a standard development process as necessary documents, the load on the persons in charge of a project is very heavy. We considered it necessary to introduce the project management tools (shown in the lower half of Figure 5) to maintain and improve the development quality and increase the development efficiency.

The project management system is used by closely associating with the activities of the Development Process Innovation WG. That is, this system manages development requirements,

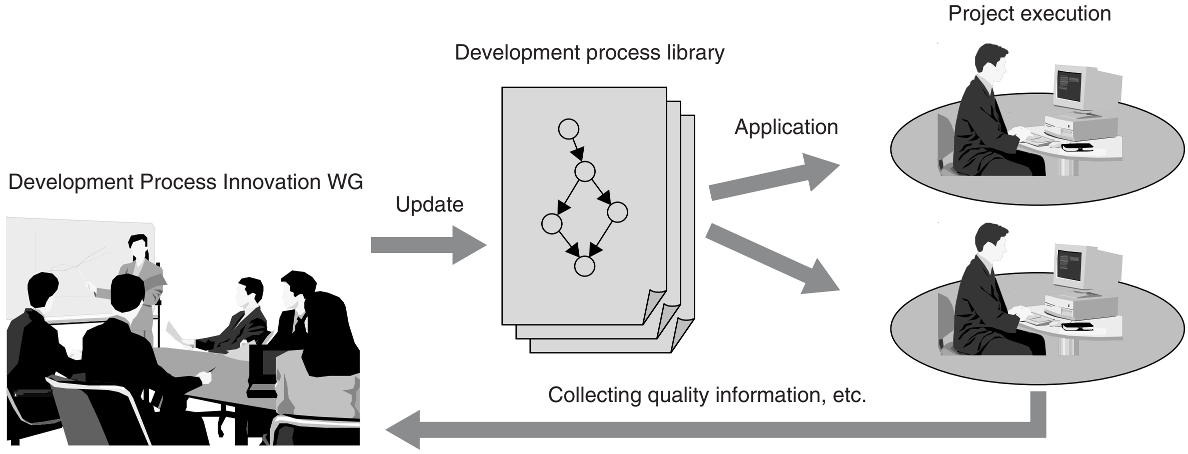

Figure 4
Optimizing development processes.

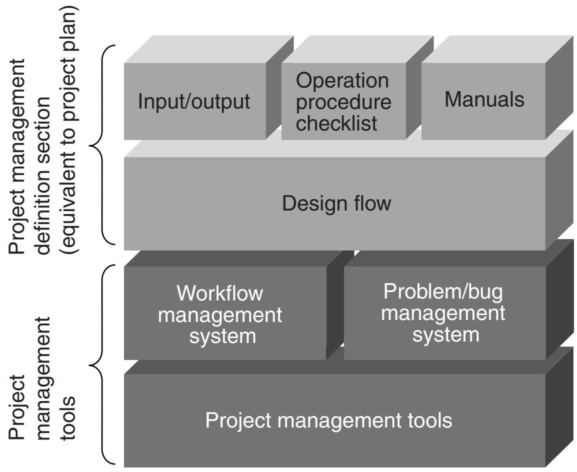

Figure 5
Structure of project management system.

test items, risks, and products during the product life cycle from specification creation, development, and shipping to maintenance. This system also supports error management, cause analysis, and process analysis at project completion.

A project is started by inputting the project management definitions, which are equivalent to the project plan requested by the project manager (upper half of Figure 5), to the project management tools (lower half of Figure 5). The Development Process Innovation WG supplies the template of the project management definitions (upper half of Figure 5). The project management tools can be used for hardware or software design according to the above definitions.

The main functions of the project management tools are as follows:

1) Unified management of products according to workflow
2) Customization at workflow execution
3) Problem management and bug management
4) Progress management

The project management tools feature the following:

1) Visualization of development processes
2) Support of concurrent operations
3) Prevention of omissions of problem management and bug management and requests for quick handling of specific tasks
4) Collection of progress data without putting a load on persons in charge
5) Linkage with existing tools
6) Compatibility with the development technical knowledge of the Fujitsu Group

Because the project management function is added to the appropriate workflow management function for development work, a one-stop service window is created. Flexible expansion into Fujitsu Group activities is enabled by these management functions and extensive customization.

We have applied the Development Process Innovation WG activities to various LSI development projects. Their targets are to maintain and improve the design quality and increase the available development/management person-hours by 10%.

6. Conclusion

This paper introduced two problem-solving approaches based on project maturity level that increase development efficiency while simultaneously improving the development quality.

Development process visualization and project management were challenging goals that could only be achieved by defining the development process in collaboration with the person in charge of product development and by supporting product development by using IT systems.

For expansion into Fujitsu Group activities, the development process had to be easy to use without placing a load on the person in charge of operations and also make it easy to collect progress/quality information. The system had to be built so that both the project manager and the persons in charge would benefit.

We will improve Fujitsu's total product development capacity, design quality, and development efficiency through continuous activities and thereby improve our customers' level of satisfaction.

References

1) CMMI Models and Modules. *http://www.sei.cmu.edu/cmmi/models/*
2) K. Fuse et al.: Evaluation of Notebook and Desktop Personal Computer through the EcoLeaf Type III Environmental Label. Proceeding of EcoDesign 2005, 2005, p-29, CD-ROM.
3) Y. Arita et al.: Fujitsu Virtual Product Simulator (FJVPS). (in Japanese), *FUJITSU*, **51**, 5, p.270-274 (2000). *http://img.jp.fujitsu.com/downloads/jp/jmag/vol51-5/paper03.pdf*
4) Y. Arita et al.: 3D Environment-Conscious Design System. Proceeding of EcoDesign 2001, IEEE Computer Society, 2001, p.231-233.
5) Tyson R. Browning: Applying the Design Structure Matrix to System Decomposition and Integration Problems: A review and New Directions. *IEEE Trans. Eng. Manag.*, **48**, 3, p.292-306 (August 2001).
6) Project Management Institute: A Guide to the Project Management Body of Knowledge (PMBOKGuide). (2000).

Yuichi Arita, *Fujitsu Ltd.*
Mr. Arita received the B.S. and M.S. degrees in Mechanical Engineering from Meiji University, Tokyo, Japan in 1987 and 1989, respectively. He joined Fujitsu Ltd., Kawasaki, Japan in 1989, where he has been engaged in development of IT systems for mechanical engineers. He is a member of the Information Processing Society of Japan.

Noriyasu Nakayama, *Fujitsu Ltd.*
Mr. Nakayama received the B.S. degree in Electrical and Electronic Engineering from Tokyo Institute of Technology, Tokyo, Japan in 1994, and the M.S. degree in Information Science from Japan Advanced Institute of Science and Technology, Ishikawa, Japan in 1996. He joined Fujitsu Ltd., Kawasaki, Japan in 1996, where he has been engaged in research and development of LSI design methodology.

Yutaka Awata, *Fujitsu Ltd.*
Mr. Awata received the BS degree in Electronic Engineering from Ehime University, Matsuyama, Japan in 1982. He joined Fujitsu Ltd., Kawasaki, Japan in 1982, where he has been engaged in research and development of LSI design methodology and LSI design. He is a member of the Institute of Electronics, Information and Communication Engineers (IEICE) of Japan.