

A tool-supported environment for knowledge feedback cycle in software development

Noriko Hanakawa

*Faculty of Management Information, Graduate school of Corporation information
Hannan University, Japan
hanakawa@hannan-u.ac.jp*

Abstract

We propose a software development environment including knowledge feedback cycle. A feature of the environment is to support the cycle with tools and researchers. Developers in the environment can acquire knowledge of past projects and experiences without additional efforts of making summaries about past projects. Moreover, the knowledge feedback cycle includes the transfer from parts of tacit knowledge to explicit knowledge. Researcher and tools are assigned to important roles in the knowledge transfer. Therefore, the environment concept will be useful to the semi-automatic knowledge transfer.

1. Introduction

Recently, software development scale becomes bigger, and software quality's impact to our society is significantly increasing. On the other hand, lifetime of software is getting shorter. In order to develop software with certain qualities in a limited time, developers require various knowledge such as cost estimation or risk management, as well as other software engineering techniques and technologies. Some of such knowledge should be extracted and accumulated through their own experiences. However, acquiring and accumulating such knowledge require long time and large efforts. In other words, it is very difficult for developers to become matured engineers in a short period.

In order to help knowledge acquirement and accumulation for novice software engineers, we propose an environment for cycling knowledge among experienced developers, software engineering researchers and novice developers. We call it KFC (Knowledge Feedback Cycle). In the KFC environment, knowledge, mainly concerning risk management and cost estimation, is extracted from past experiences for future reuse. A feature of the environment is semi-automatic. Mainly three tools,

EPM (Empirical Project Monitor), Project Replayer, and Project Simulator, are used to capture and circulate knowledge in KFC. EPM[1] is a tool to automatically collect project data from source code repository, bug-reports and e-mails. Project Replayer is a tool used to review data of past projects. Project Simulator is used to provide actual feedback to developers.

In this proposal, we mainly present the tool-supported environment for KFC. Especially, we show how knowledge is semi-automatically feedback though the environment.

2. A semi-automatic environment for KFC

Developers understand an importance of transfer of past experience for future project. Managers have often experienced bug reports caused by same problems such as insufficient communication with customers. Managers said "Last year, I received similar bug reports caused by insufficient communication". Developers answered "I was not here. I worked at another project last year". This is a typical case in which knowledge and experiences did not transfer to future projects. Because knowledge and experience are shut into personal memories, knowledge and experiences can not be reused as an organization. Of course, many trials of knowledge transfer have been proposed in knowledge engineering field [2]. Tacit knowledge of individuals should be explicit, after that, the explicit knowledge is shared in an organization. However, if people transfer manually from tacit knowledge to explicit knowledge, for example, making rule documents, it will take long time and large efforts. Because of nuisance of making the documents, manual knowledge transfer is problematic. Therefore, we propose a software development environment supporting a semi-automatic transfer of knowledge.

The purpose of KFC is to circulate knowledge and experience of past projects to future projects. Developers are supposed to acquire new knowledge

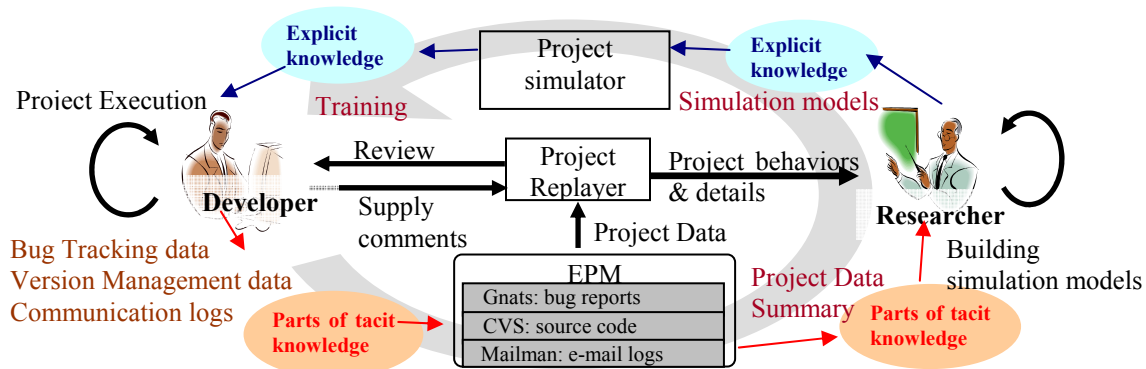


Figure. 1. An environment for Knowledge Feedback Cycle

while experiencing software development projects. To establish automatically the cycle, KFC employs three tools; EPM (Empirical Project Monitor), Project Replayer and Project Simulator (See Figure. 1). A typical scenario in the KFC would be as follows;

- Step1: Various development data (records of code modification, bug tracking, and emails) is automatically captured by EPM during the project enactment.
- Step2: Researchers analyze the collected data to construct various simulation models using the Project Replayer and analysis tools.
- Step3: Using the Project Replayer, developers review past projects. Events and accidents that are not recorded by EPM are also clarified in interview with developers.
- Step4: Regarding results of reviews and interviews, researchers refine their simulation models that were made in Step2. The models are embedded into the Project Simulator.
- Step5: Using the Project Simulator, novice developers learn complicated phenomena in past projects. Developers can also utilize the Project Simulator to make their next project plans. The planned project is regarded as the target of Step1 of the next cycle.

3. A benefits from the environment

It is a most characteristic feature of the environment that the tools and researchers support the knowledge feedback cycle. At first, while developers make usually software, development data (records of code modification, bug tracking, and email logs) is recorded automatically. Knowledge and experiences are implicitly buried under miscellaneous development data. Developers can not identify even their-own knowledge in the development data. After researchers have to analyze the development data, researchers extract knowledge and experiences. In short, researchers transfer from parts of tacit knowledge in the development data to explicit knowledge. The

explicit knowledge is embedded to a simulation model. The formulas of the simulation models present the explicit knowledge. Of course the extraction of knowledge from the development data is difficult. Project Replayer is useful to extract knowledge[3].

Next, by implementing the simulation model to the Project Simulator, the explicit knowledge becomes available to developers. Virtual projects in the Project Simulator behave based on the simulation model. While novice developers are utilizing the Project Simulator, the developers can learn phenomena that occurred in past projects. That is, the novice developers can acquire knowledge of past projects. In KFC, developers do not need any tasks of making the development documents in order to transfer the past knowledge. Developers do only usual development tasks and learning on the Project Simulator. Although researchers have to make simulation models with analyzing the development data, the knowledge transfer is semi-automatic for developers who are true “users” of the environment.

3. Summary

A concept of a semi-automatic environment for knowledge feedback cycle has been proposed. The EPM and the Project Replayer have been already developed. In future, after the Project Simulator will be built, we will confirm the usefulness of the environment.

References

- [1] Masao O., et. al., “Empirical Project Monitor: A System for Managing Software Development Projects in Real Time”, in Proceeding of ISESE2004, Vol.2 , 2004, pp. 37-38.
- [2]Nonaka I, “A Dynamic Theory of Organizational Knowledge Creation”, Organization Science, Vol.5-1, Feb. 1994.
- [3]Goto K., Hanakawa N., Iida H., “Project Replayer-An Investigation Tool to Revisit Process of Past Projects”, Proceeding of International Software Process Workshop and International Workshop on Software Process Simulation and Modeling, May 2006, pp72--pp79.