

Information collection and investigation for software process improvement – a case study

Espen Frimann Koren

Industrial Systems Development Group, Department of Informatics, University of Oslo
P.O. Box 1080 Blindern, N-0316 Oslo, Norway
Tel: +47 22 84 00 60 Fax: +47 22 84 00 72
espenfk@ifi.uio.no

Keywords. Software process improvement, process assessment

Classification. First year ph.d. (2 months)

Abstract. This paper outlines some research questions in relation to software process improvement. Software process improvement should be a continuous activity where you go through with a set of activities to achieve a higher level of maturity. These activities can be seen as a four step process; information collection and investigation, decision making and planning, implementation, and evaluation. This paper focuses on the first step and presents a case study that show some experience with investigation and information gathering for software process improvement. The case study illustrates how several sources of information may be combined to provide necessary information to identify goals and leverage points.

1 Introduction

Software engineering is a complicated discipline and a many-sided discipline. On one side software engineers work with human perception and learning to create useful and usable systems. On the other side they work with systems integration and cooperation across heterogeneous hardware, network and software platforms. Zahran starts his book about software process improvement [16] with a discussion about ‘process thinking’. He refers to Watts Humphrey, the main architect behind the Capability Maturity Model (CMM), and his book [7], where Humphrey says: ‘An important first step in addressing software problems is to treat the entire software task as a process that can be controlled, measured, and improved.’

There are two important aspects of Software process improvement (SPI). First it is about moving software development towards an engineering discipline. Second it is about improving the quality of the software developed.

In this paper we discuss some aspects of a framework [16] for SPI in light of interesting research questions related to the framework. The research questions are related to methods and techniques that may be used. A case study is presented that addresses what kind of information to collect and how to collect it.

The remainder of this paper is organised as follows: Section 2 describes the framework and discuss relevant research questions. Section 3 presents the case study. Section 4 discusses the implications of the case study. Section 5 concludes and outlines future work.

2 Software process improvement

Zahran [16] proposes a framework for software process improvement (SPI). It consists of the following four components:

- Software process infrastructure. There are two types of infrastructures necessary. The first is the organisational infrastructure; roles and responsibilities. The second is the technical infrastructure; tools and facilities. The roles have the responsibility for establishing, monitoring and enforcing the process activities. The tools are for example data and document storage and retrieval tools for manipulation of process data (documents, measurement data, etc.).
- Software process improvement roadmap. There are various SPI model initiatives: CMM ([14]), ISO 9001/9000-3 ([11], [10]), ISO/IEC 15504 and ISO/IEC 12207 ([8], [9]), MIL-STD-498 ([4]), SPICE ([6]), BOOTSTRAP ([12]) and others, for example SPIQ [5] and [1]. Without a process improvement roadmap, an organisation could be facing some of the following risks:
 - Inconsistent basis for process assessment and improvement
 - Difficulty in focusing the improvement plans on the most important areas
 - Potential inconsistency in successive improvement plans
 - Inability to benchmark across organisations.
- Software process assessment method. A software process assessment is an evaluation of the software process of an organisation. CMM, ISO/IEC 15504, SPICE and BOOTSTRAP all contain assessment methods. Process assessment provides a baseline of the current status of the process and is used as a basis for improvement. It is important to use a process assessment method to avoid the following risks:
 - The assessment results may miss some important aspects of the process
 - Different assessment teams could use different approaches that could lead to inconsistencies in results
- Software process improvement plan. The plan describes which actions to perform based on the assessment findings. The actions should improve the process discipline and effectiveness. It is important to run the improvements as a project with manager, budget and a sponsorship from the management.

There are four phases in a SPI cycle: assessing, planning, implementing and evaluating. The purpose of the SPI cycle is to move the process to a higher level of maturity. The full cycle may be depicted as in Fig. 1. Given an improvement roadmap you assess the existing process by investigation and information gathering. You

decide which improvements to conduct and plan the implementation. Then the plan is implemented and the improvement effort is evaluated.

Information collection and investigation	Decision making and planning	Implementation	Evaluation
--	------------------------------	----------------	------------

Fig. 1. A full SPI cycle

2.1 Information collection and investigation

The purpose of this step is to create a picture of the current status of the organisation. More specifically, it is to get a picture of the software development process. The assessment will discover indicators of a need for process improvement; that is goals and leverage points for process improvement. An indicator could be discontented customers and/or employees, a decrease in the ability to deliver or exploitation of the employees. These observable facts may only be symptoms of other problems that are not so easily detectable. This leads us to the research questions: What kind of information do we need? How do we get this information?

What kind of information do we need?

Mackey ([13]) concludes that before undertaking improvements to the development process, companies should understand both the process areas where they lag competitors or industry norms, as well as which process areas have the greatest potential to enhance bottom-line results.

Zahran ([16]) describes three issues that must be considered for SPI:

- **Assessment scoping.** This stresses the importance of understanding the business objectives of the SPI effort. The assessment must be put into the right context, for example whether it is part of a capability evaluation. The scope of the assessment could range from one small group or project to the complete organisation, and for one specific process that needs evaluation to the full set of processes organisation wide. The scope of the assessment will have impact on the approach, the steps, the instruments, the level of sponsorship and the budget.
- **Process culture.** Assessing the process culture is difficult, because culture cannot be touched. The purpose is to find the level of trust in and commitment to the existing process. Process culture is how the process is being used. The assessment must reveal attitudes, competencies, routines, skills, etc. amongst the employees.
- **Process infrastructure.** The process infrastructure consists of three levels; organisational, project and personal. The process infrastructure is the documented artefacts and routines.

The various standardisation efforts have categorisations of information that must be collected. Botten ([3]) describes an approach for merging multiple assessment data

is developed so that it is possible to compare assessments or use various assessment philosophies when going through with process improvement.

How do we get this information?

Zahran ([16]) defines two types of assessments:

- Self-assessment. Self-assessment can be team-based or continuous.
- Independent assessment. Independent assessment is usually projects.

Depending on the source of information, the instruments used to gather the information will vary. The work resembles empirical research to a large degree, so it is useful to look at empirical research methods.

An assessment should cover three issues; scoping, culture and infrastructure. To determine the scope, a workshop can be useful. It is important to get as many viewpoints as possible to identify the most important areas of concern. The workshop should result in a report which will be used as input to planning the fact gathering.

The infrastructure must be relieved using document studies. Both process documentation and project documentation should be studied.

The most difficult task is to gather the culture. The most obvious technique is interviews, but also more sophisticated techniques like observation and participation should be evaluated. Interviews tend to be too personal, and people may have problems expressing their sincere attitudes to an interviewer. Using participation the observer will work as a part of the group and can observe the day-to-day progress and attitudes. Depending on how the observer is introduced the adoption of him will influence the quality of the information gathered.

Open research questions

We believe that the process assessment is mainly based on experience not on research. This is due to facts reported by Pfleeger. The amount of information you will have to gather following CMM or even Rational Unified Process is enormous. A very important question is how to manage the data collected in the assessments. Research could be conducted to investigate is whether one could find a subset of information that is more essential than others to discover the goals and the leverage points of SPI.

There are many types of organisations and our impression is that most assessment programmes are tuned towards large production line companies. Many software companies do not fit into that description. A lot of companies do contract development in the customer's domain. They also need processes and process improvement programmes. The challenge in SPI for these kinds of organisations is the number of processes they are engaged in and the dynamics of their environment. Research may be conducted to identify and evaluate assessment methods for these environments.

3 Case study

The case study is conducted at a product development company in Norway. Their product is a blend of hardware and software and is used in telecommunication. They

have noticed some organisational commotion, and need to evaluate, and possibly change, their development process. The author has so far conducted some interviews to collect information to be used as basis for decision making and planning of process improvements.

3.1 Fact gathering methods

These are the main sources of information:

- The company standards
- Memo from the quality manager
- Interviews
- Post-mortem project meetings

The company standards are published on the company intranet. The memo is a report stating facts about the current situation and suggests actions to be taken. The interviews are conducted by the author. The post-mortem project meetings are very common in the organisation and they have good experiences from earlier meetings. Results from the meetings are not reported here.

3.1.1 The interviews

There are two important issues to address when conducting interviews:

- Determine the level of structure (structured or open-ended)
- Decide whom to interview

We decided to use open-ended questions. We wanted to get as ‘clean’ responses as possible by not using leading questions. The interviews had a theme though; how the interviewee experienced the development process in the daily work:

- How are they organised, how do they experience exchange, etc.
- How do they work in the projects, how is the relation between the projects and the organisation, the company standards, etc.
- Which parts does the complete product consist of, what is the role of the interviewee’s subsystem in the whole, etc.

We used three dimensions to categorise the interviewees (**Table 1**).

Table 1. Distribution of the interviewees.

	Department#	Project#	Seniority(years)
Person 1	3	1	1
Person 2	2	2, 3	Long
Person 3	1	2	Long
Person 4	4	2	Long
Person 5	2	2	10
Person 6	3	1	3 ½

The interviewees knew about the interview for from one to seven days in advance, and they were told the theme of the interview so that they could prepare. The interviews were from a little above 60 minutes to about 150 minutes long. The author wrote minutes of the interview. The minutes were sent to the interviewee for

comments. No technical aids were used. Seaman ([15]) states that some audio-visual aid is necessary to conduct good interviews. In this case we decided that it would be easier to get sincere answers without using technical aids.

3.2 Results

3.2.1 Results from the memo

The memo states: “*We experience problems in coping with the dynamics in today’s projects*”. It proposes a set of causes for this experience:

- Unstable or inaccurate requirements. The users and the customers have an idea of what they want, but they have no large market for the products and there are no standards to base the requirements on.
- Ever-decreasing time to delivery. The race for being first to establish a name in the market, demands rapid development.
- Change in technology -- use of COTS
- Exploitation of staff

The last cause is very much a consequence of the three first, but it is the exploitation that has triggered the wish for SPI.

The memo proposes some improvement areas:

- The interface towards the customer and the users. It is important to achieve a common understanding between all the stakeholders what is going to be developed, what time it is going to be finished and to what cost. Problem domain modelling, requirements management, and change cost estimation are improvement areas in this perspective.
- Inter- and intra-project communication. Achieve common understanding within the project. Risk management, software categorisation, model visibility, software decomposition, and the use of components are improvement areas in this perspective

3.2.2 Results from the interviews

The company is a very good place to work in:

- Stimulating working environment.
- Good atmosphere.
- The amount of challenges in the projects.

The company is project-driven, and most work, also knowledge management and training, is done within project frames. One interviewee suggested that searching for and testing of new technologies should be done under departmental supervision to lessen the stress on projects.

One interviewee said that the organisation seemed to be too unfocused in the choice of projects to take on and technologies to adopt, in the sense that they had taken on all kinds of projects and consequently stretched the organisation too thin with respect to competence.

Another issue is that knowledge of what the other departments know and do was not very well communicated. The projects are organised in relation to the subsystem

structure. Each subsystem requires special competence, and from a high level view there is not much common between them.

One interviewee said that the organisation does not renegotiate functionality with the customers even if the change could be beneficiary for them. Discussing such issues could make the customer more confident with the product and the developing organisation.

Along the same line are support contracts. They have always provided free support to their customers. Use of COTS has led to the need for paid support, since the COTS suppliers almost always demand paid support.

The most pronounced change is the use of COTS. Since the functionality COTS delivers influence the requirements, a more bottom-up incremental approach is needed [2]. Earlier they made all the hardware and software themselves, and they used a more top-down waterfall approach, since they controlled the design of the hardware.

The development process used to work, but on the last couple of projects it did not. One interviewee mentioned two issues that they needed to address to be able to go through with the process better

- Focus on prioritising the requirements to identify the most important functionality early.
- Find the minimum amount of functionality that needs to be developed.

Two of the interviewees detailed the problem with the process further. They both worked on the same project, but at different levels of abstraction:

- “The problem was not the process in itself, but the lack of process. The project’s time to delivery was completely unrealistic. The project became stressed and made ad hoc shortcuts in the process. For instance no high level design documentation existed.”
- “The project tried to use the process many times in kind of increments, like the spiral model. A common set of documentation on a high level was maintained.”

They have problems trusting COTS. COTS suppliers promise delivery of some functionality, but do not deliver. The consequence is that the contractual promises to the customer are hard to hold. [2] notes that it is unwise to assume that advertised COTS capabilities are necessarily real.

4 Discussion

The case study illustrates how the two sources of information supplement each other. The main factor in the memo was the exploitation of the employees. That issue did not come up during the interviews. The memo suggested three causes for the exploitation. The interviews supported these causes to a large extent. Naturally the issues were treated in more detail in the interviews.

Some issues surfaced during the interviews. The discussion about the organisation gave valuable new input. Also the discussion about the actual use of the current process provided new insight.

The interview process itself may also give some benefits besides new knowledge. For example the involvement will be increased. SPI cannot be driven by one man’s

initiative. Zahran ([16]) states that one of the most important tasks within SPI is institutionalisation of the process improvement. That is to pinpoint responsible resources and involving all levels in the organisation. In this regard it is necessary to discuss the selection of interviewees.

The rationale behind the interviewee selection was to find the smallest set of people that could be said to represent the company. Since SPI will affect the whole organisation, we wanted to involve as much of the organisation as possible. We therefore picked people as described in 3.1.1. The main problem with this selection is that it is quite difficult to generalise anything from the results.

The interviews provided deeper knowledge about the issues from the memo, and also surfaced some other issues. The interviewer should have used some technical aid, because the loss of information is large when both asking questions and writing the minutes. The number of interviews was rather small. The information was difficult to generalise. It is an open question whether we would have found more issues if the number of interviews had been larger.

5 Conclusions and future work

We have seen that multiple sources of information, and multiple ways of collecting it gives a better foundation for defining SPI goals and leverage points.

The next step is to extend the case study to the decision making and planning phase. An interesting aspect of this phase is the correlation between the information gathered and the improvement areas chosen. A part of this will be to test some qualitative analysis methods on the information. Seaman ([15]) discusses the use of qualitative methods in empirical studies of software engineering. The paper is written for researchers, but SPI work resembles research in many ways, so it should be relevant. The paper covers both data collection and data analysis and can be useful in both fact gathering and fact analysis.

Further the organisation will implement some process changes. We will continue to observe the process. It is possible that we can test different process changes in parallel and compare effect and success.

6 Acknowledgements

I would like to thank the interviewees and the quality manager for cooperation while working on this paper. I also would like to thank Erik Arisholm and Dag Sjøberg for valuable input and motivation.

The work is funded by The Research Council of Norway through INCO (Incremental and Component based software development).

7 References

1. Bennetts, P.D.C., Wood-Harper, A.T. and Mills, S. The Soft System Methodology as a Framework for Software Process Improvement. in McGuire, E.G. ed. *Software Process Improvement: Concepts and Practices*, Idea Group Publishing, Hershey, USA, 1999.
2. Boehm, B. and Abts, C. COTS Integration: Plug and Pray? *IEEE Computer*, 32 (1). 135-138.
3. Botten, N.A. Development process assessment toward leading edge quality. *IEEE journal on selected areas in communication*, 12 (2). 251-257.
4. DoD. Military Standard for Software Development and Documentation. MIL-STD-498, US Department of Defense, 1994.
5. Dybå, T.e., Wedde, K.J., Stålhane, T., Moe, N.B., Conradi, R., Dingsøy, T., Sjøberg, D.I.K. and Jørgensen, M. SPIQ Metodehåndbok, 2000.
6. El Emam, K., Drouin, J.-N. and Melo, W. SPICE: The Theory and Practice of Software Process Improvement and Capability Determination., IEEE, 1997.
7. Humphrey, W. *Managing the software process*. Addison-Wesley, 1989.
8. ISO. Information Technology - Software Life Cycle Processes. ISO/IEC 12207., International Standards Organisation, 1995.
9. ISO. ISO/IEC 15504 Draft Standard for Software Process Assessment (Parts 1-9), International Standards Organisation, 1997.
10. ISO. Quality Management and Quality Assurance Standards, Part 3: Guidelines for the ISO 9001 to the Development, Supply and Maintenance of Software . ISO 9000-3, International Standards Organisation, 1991.
11. ISO. Quality Systems: Model for Quality Assurance in Design/Development, Production, Installation and Servicing. ISO 9001, International Standards Organisation, 1987.
12. Kuvaja, P., Simila, J., Krzanik, L., Bicego, A., Soukkonen, S. and Koch, G. Software Process Assessment and Improvement: The BOOTSTRAP Approach., Blackwell, 1994.
13. Mackey, W., Organizing for innovation: focusing on the critical few R&D improvements that drive tangible results. in *IEEE International Engineering Management Conference*, (2000), 117-119.
14. Paulk, M., Curtis, B., Chrissis, M.B. and Weber, C.V. Capability Maturity Model for Software, SEI, 1991.
15. Seaman, C.B. Qualitative Methods in Empirical Studies of Software Engineering. *IEEE Transactions on Software Engineering*, 25 (4). 557-572.
16. Zahran, S. *Software Process Improvement. Practical Guidelines for Business Success*. Addison Wesley, Harlow, England, 1998.