

Out of the Dark: Adaptable Process Models for XP

Christian Bunse

Fraunhofer IESE
Sauerwiesen 6
67661 Kaiserslautern
Germany
+49-6301-707-222
Christian.Bunse@iese.fhg.de

Monica Pelayo

Western Illinois University
1 University Circle
Macomb, IL 61455-1390
USA
+1-309-298-1073
monixpelayo@yahoo.com

Jörg Zettel

Fraunhofer IESE
Sauerwiesen 6
67661 Kaiserslautern
Germany
+49-6301-707-222
Joerg.Zettel@iese.fhg.de

ABSTRACT

Lightweight methods such as extreme programming or agile computing are becoming increasingly important for the rapid development of software applications. However, there is a tradeoff in using lightweight methods. Often they lack in providing a process model (i.e., a sound description of roles, artifacts and activities), which allows easy adaptation for different organizations, as well as systematic guidance for developers. This paper describes a project, performed at Fraunhofer IESE, which addresses this problem by providing a systematically defined and easily adaptable process model for XP. This model is then used as basis for the generation of a web-accessible electronic process guide (EPG).

Keywords

XP, Process Model, Spearmint, Process Modeling, EPG, Guidance

INTRODUCTION

Extreme Programming (XP) [1,2] is an approach to software development which emphasizes a very tight cycle between code creation, testing and debugging through the principle of “lightweight traveling” (i.e., code is immediately integrated into the overall system after creation, and then subjected to serious testing). The expected benefits of such an approach include the rapid development of minimal systems, early creation of executable code, and low defect numbers.

Despite its undoubted strengths, however, the XP approach has one major drawback: lack of guidance. Although many books and web representations exist on XP, it is not easy for a developer or project manager to set-up and run an XP project. One reason for this is that a concise and precise model of the XP process is missing. Such a model does not only define activities, milestones, and roles, but it is also a major source of reference for every developer in a project.

A concise XP process model, which is (1) adaptable to a project and/or organization, (2) maintainable, and (3) easily accessible, would be the ideal complement for planning and running XP projects. Therefore, we identified and analyzed the available documentation on XP and modeled the XP process with SPEARMINT™ [6], a process modeling tool that has been developed by Fraunhofer IESE. The resulting process documentation can be easily modified and adapted to the needs of an

organization. Using SPEARMINT™’s ability to generate an Electronic Process Guide (EPG) [7] from a process model, a web-based process handbook for XP projects was created as well. This can easily be accessed by project members, keeping them always informed. Figure 1 shows the home page of the XP process guide.

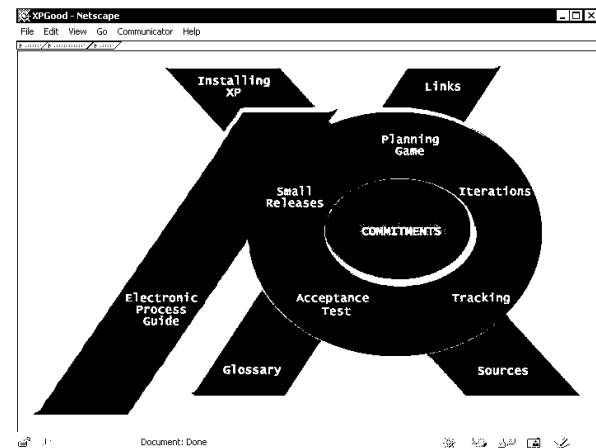


Figure 1: XP Process Guide

In the following we give a short overview on XP, process modeling, and SPEARMINT™/EPG before going into the details of the XP process guide.

BACKGROUND

XP

XP is a lightweight software development methodology that focuses on the principles of communication, simplicity, feedback and courage. The goal of XP is to deliver software products within time and cost constraints. XP is recommended for small projects where requirements change fast, or where there is a high risk of failure.

Process Modeling

Software processes are series of human actions to create software products. Process models are explicit representations of such processes that support communication, package experience, ease analysis, and provide guidance for process performance [8]. Process modeling is a software engineering discipline that focuses on how to create useful models of real-world processes.

SPEARMINT™

SPEARMINT™¹ is a process modeling tool that has been developed by Fraunhofer IESE over the last five years [6]. Its intention is to make complex software processes easily understandable, which is achieved by natural concepts and a graphical notation close to UML. To provide process guidance for project participants, SPEARMINT™ can generate an Electronic Process Guide (EPG) from any process model. This is a web-based process handbook, which provides dedicated means for quickly finding any relevant process information.

In this project, SPEARMINT™ has been used to model the XP process and to generate a web-based XP process guide.

THE XP PROCESS GUIDE

Extreme Programming (XP) is a recently developed, lightweight software development methodology, which becomes increasingly popular due to savings in resources and due to increased time-to-market. Although there are a large number of books and other publications on XP around, it is not easy to introduce it into existing software organizations. This is due to the fact that, up to now, a tailorable (i.e., specifically adapted for an organization) and easily accessible (i.e., online available for each developer) process description is missing. Thus, there is a need for a systematically developed process model (i.e., by using SPEARMINT™) which describes the basic roles, artifacts and activities involved in a XP project. This model can then be adapted to the different needs and can be used to create an organization-specific EPG.

Resources

An exhaustive study about XP was made in order to generate the EPG. Besides reading the XP book collection by Kent Beck, Ron Jeffries, Ken Auer, and Roy Miller, several websites were consulted and compared. Some interviews with XP practitioners were carried out on-line (through the XP discussion group) as well as in person at Fraunhofer IESE.

Appearance & Practical Usage

The XP process was divided into several abstraction levels according to the level of detail to be explained for every stage of the process. The roles used in the model are: developer, customer, manager, and tracker.

The first level of abstraction is shown in Figure 2, which is a graphical representation of activities, artifacts, and product flow among them. Everything starts with two activities: the customer describes the system, and the “rules of the game” are defined.

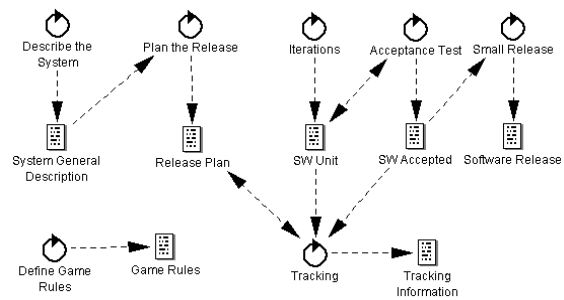


Figure 2 XP Process: Activity “XP Plan”

While the customer is describing the system, the developers write “user stories”. If for any reason a user story seems to be too big, it must be broken into two or more user stories.

After the system is described, the “tracker” must keep track of the level of progress in relation to the original plan and notify the “manager”. This way, the manager makes sure that he/she will count on reliable information at all times, in order to make decisions.

The second step is to plan the release of the unique software product by the process: according to the system requirements, to the level of expertise of the team members, tools available, etc. the development team will be able to give an estimate of the time and resources that it will take to successfully develop the user stories. If the customer agrees on time and cost, the team starts working; otherwise, they must negotiate, which features are to be developed for the first release of the system.

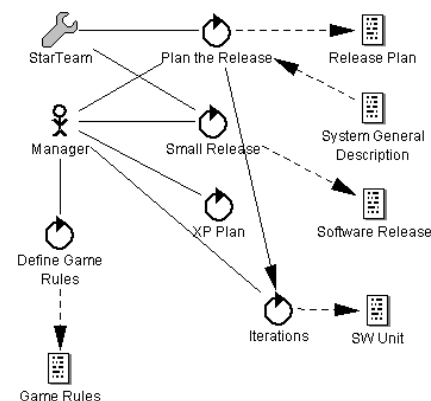


Figure 3 XP Process: Manager View

Then, the development team breaks the user stories into “tasks”, which are atomic units to code. The tasks may be further divided into more tasks or combined. This step corresponds to the next level of abstraction, as part of the activity called “Iterations”.

¹ SPEARMINT™ is a registered trademark of Fraunhofer IESE. More information and a demo version of the tool can be found at: “http://www.iese.fhg.de/Spearmint_EPG/”.

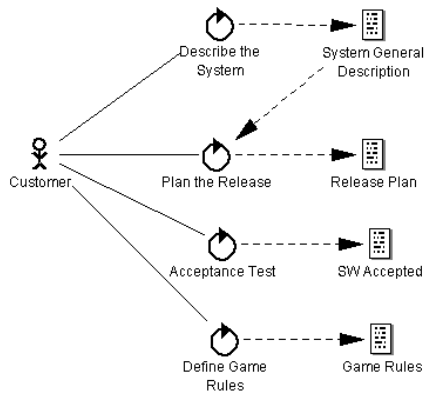


Figure 4 XP Process: Customer View

Once the user stories are implemented and internally tested, the customer is in charge of an “Acceptance Test”. Upon customer approval, a small release is made and the team is ready to go for the second release, where the customer is expected to include all the features that couldn’t be included in the current release.

This same abstraction level, from the point of view of the manager would look as in Figure 3, and with the customer perspective as in Figure 4.

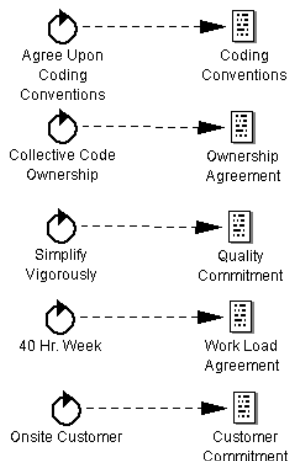


Figure 5 Game Rules

The “Game Rules” are the core of XP. These rules have been around for a long time, however, they were changed for more elaborate practices that led to some of the major problems that the software industry faces today, such as the inability to provide the customer with reliable time and cost estimates. Some of the artifacts produced by this activity are contained in Figure 5.

The development team must agree on the coding conventions to be used, such as naming protocols. The developers own all the code: they are free to make changes to any piece of code that they think can be improved, as well as to simplify anything that seems redundant.

One of the features that make SPEARMINT™ a powerful tool, is the flexibility to let the user get different views of the process: one view per role, besides the general view of the process. Since the roles perform different activities, there is no need to show all the activities that a particular role is not performing. The user also gets the graphical version of the process model in the EPG: it is possible to view the interconnected activities and the artifacts produced by them, and from that view access the information regarding each item.

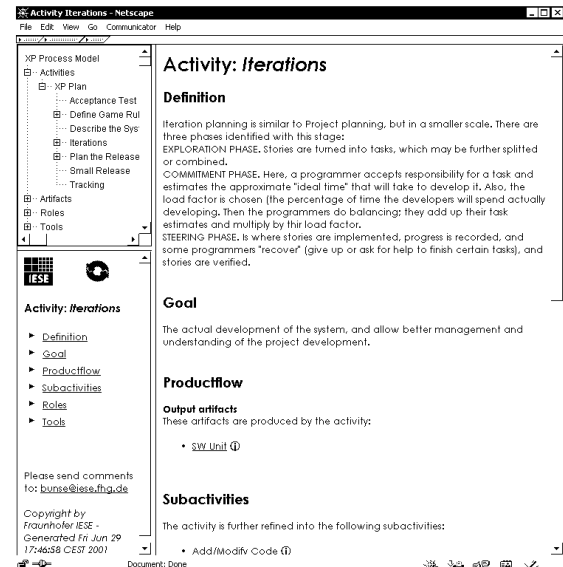


Figure 6: XP Process Guide: Activity “Iterations”

Of course, the XP process model is much more exhaustive than it can be described here. Figure 6 gives an impression of what the XP process guide (i.e., the generated EPG) looks like. Here, the guidance on activity “Iterations” is shown.

Experiences

Practical experience on using an electronic description of a software process [7] showed that developers prefer to have an information resource in form of an EPG on the actual process, which is always available and up to date. Furthermore they appreciate the guidance such an EPG provides for their daily work.

This positive experience has also been made with the presented XP process guide. In a student project the guide has been used in different ways. First, to educate a student developer in doing XP projects. Second, by using the EPG to guide the student in applying XP in a development project.

In summary, the slightly higher effort for creating a SPEARMINT™ model to create an EPG pays-off in later projects since developers know what they should do and how (i.e., guidance) and can always refer to the actual definition of their process.

Open Issues

Some problems detected for XP include the fact that

developers are not willing to do pair programming. The programming practices currently found in the market are not the best that could ensure “collective code ownership”, “simplify vigorously” or refactoring. This leads to a detected need for the market: to change the way people are taught to program, this is, to research and develop innovative teaching methods and tools.

Requirements elicitation is an area that despite of the specific needs that XP demands on this matter, that area itself demands more research. Team management, as well as other methodologies applied to the software development area, need to be adapted to the specific needs of this field. Teamwork is done differently; tracking and the relationship with the customer are different than in other types of projects, so managing needs to be adapted to this unique environment, as well as management training.

In order to keep improving the EPG, the following practices are recommended:

1. Implement an “in-house knowledge-gathering mechanism” that would help document in an organized fashion (through the EPG) the knowledge earned through the process implementation in the unique environment and nature of the project. This version of the EPG will serve as a “master knowledge base” for the organization where it is being used.
2. Every time a new project is started, generate a copy of the “master knowledge base” and modify it as the project demands, in order to immediately reflect changes as the project goes on.

SUMMARY

Within this paper we presented the results of a project on developing an electronic process guide for extreme programming. By using the SPEARMINT™ tool for systematically describing the XP process, based on available information resources, an electronic process guide (EPG) has been developed. This EPG can simply be installed within the Intranet of any software organization and then easily accessed by every project member of that organization. Experiences in using such an

EPG showed that developers appreciate online-available information as well as guidance on the next steps to be performed.

Another advantage of this approach is that the SPEARMINT™ model can easily be adapted towards the needs of different organizations. Thus, by using SPEARMINT™’s automatic EPG creation facilities the process guide always reflects the actual version of the process applied and helps developers to be informed about the current version as well as on the required activities.

INFORMATION AND QUESTIONS

For more information, contact: bunse@iese.fhg.de.

REFERENCES

1. Beck, K. *Extreme Programming Explained: Embrace Change*. Addison Wesley, 1999.
2. Beck, K., Fowler, M. *Planning Extreme Programming*, Addison Wesley, 2001.
3. Succi, G., Marchesi, M. *Extreme Programming Examined*, Addison Wesley, 2001
4. Newkirk, J., Martin, R.C., *Extreme Programming in Practice*, Addison Wesley, 2001
5. Jeffries, R., Anderson, A., *Extreme Programming Installed*, Addison Wesley, 2001
6. Becker-Kornstaedt, U., Hamann, D., Kempkens, R., Zettel, J., Support for the Process Engineer: The Spearmint Approach to Software Process Definition and Porcess Guidance, *Proceedings of the 11th Conference on Advanced Information Systems (CAISE'99)*, 1999
7. Becker-Kornstaedt, U., Verlage, M., *The V-Model Guide: Experience with a Web-Based Approach for Process Support*, *Proceedings of Software Technology and Engineering Practices*, 1999
8. Rombach, H.D., Verlage, M., *Directions in Software Process Research*, *Advances in Computers*, pp 1-63, Vol. 41, Academic Press, 1995