Migrating a Development Project to Open Source Software Development

Wolf-Gideon Bleek, Matthias Finck
Department of Computer Science
University of Hamburg, Germany
{bleek, finck}@informatik.uni-hamburg.de

Abstract

The CommSy-system is a web-based community system, which has been in development since 1999 at the University of Hamburg. It has initially been developed by students and researchers in their spare time. Its last organizational setting was a publicly funded research project, which allowed for full-time and part-time developers. As that project has come to an end, we are aiming at an open source project to ensure continuity by providing a frame for people from different organizations. In this paper we discuss the characteristics of this specific project and of other open source projects to identify a strategy for migrating that particular project to open source. We outline the actions taken to migrate the existing project to open source software development and raise questions concerning the necessary characteristics of an open source project as well as whether the actions will suffice or not.

1. Introduction

Open source software is an increasingly popular form of software used as well as developed. The open source label is becoming a synonym for good quality on many levels. We feel that some of the promises of open source software development may be suitable to help sustain an existing project. In our case (study) we are discussing the conversion of our project, which started in 1999 and is now at the verge of becoming an open source project. This paper describes the setting that has been established over the years and how the transformation process has been initiated. First, we will present our understanding of an open source project and software development process and draw on essential characteristics that might help employing an open source migration. We will then describe the CommSy-project so far, followed by actions taken to migrate the existing project to an open source project. We state the goals connected with these actions and sketch further measures necessary and outline challenges as well as problems that have arisen so far or are likely to come

up in the near future. We conclude with an outlook on future work and possible research questions are stated.

2. Key Characteristics of an Open Source Project

Open source projects have become very popular in recent years. Even formerly clearly commercial software is developed in open source projects these days [8], [9]. Current literature presents different views on open source software development [12], [2], [13].

Common to all open source software development projects is that the latest source code of the software is available for free to anybody. A license specifies in which way the software and the source code may be used. Enhancements or corrections to the current version are considered "contributions" to the project and are incorporated coordinately. Becoming a member of an open source project is based on the fact that one contributes actively to the project by e.g. fixing a known bug or implementing a feature presented on a "roadmap".

An open source project is usually managed by volunteers. Most of the coordination and communication work is done with electronic media, e.g. email, online discussions forums, and web-based bug-tracking systems [6], [6].

From the process point of view open source software development is regarded as agile software development [1]. This is mainly because of four reasons: First, the amount of documents produced in advance of and during development is relatively small. Code and prototypes need to speak for themselves. Second, the team structure is flat, and even though the number of participants may be large, the actual number of people working on the same area of the software is fairly small. Third, the assignment of tasks is ad-hoc (on a favor basis) and the tasks are small. Fourth, the release cycles are short (less than half a year, or even shorter than 3 months).

To characterize a software development as an open source process we feel that it must fulfill three essential characteristics:

Openness: The project must be open to new participants, e.g. new developers can get involved. Furthermore, openness means that anybody can use the product by simply installing it. It also means that the process itself is open to changes, which leads to the next point:

Agility: The development process itself must be agile in the sense that the process is carried out in short cycles and may be changed as needed.

Distributed: The participants of the development process are not all located at the same place.

3. The CommSy Project

CommSy stands for community system and is a web-based groupware system to support the communication and coordination in working and learning groups, which has been developed at the University of Hamburg.

In the last three years CommSy has been used predominantly in university contexts to support project-based learning. It supports communication (for example news and discussion forums) and the exchange of working materials (with e.g. file uploads and online documents) as well as organizing the project (aided by dates and groups). CommSy has been used in a variety of teaching fields including history, languages, education, economics, and informatics and it consists of three key features [10]:

(1) CommSy Project Rooms are designed for closed learning groups of approximately 10 to 30 members. These groups normally work for a limited period of time. To support learning group activities, CommSy offers several groupware functionalities like news or events that can be announced, discussion forums, personal home-pages, where members are able to present themselves to the group and materials that can be written in a cooperative way, collected and classified by the users. A Project Room is an integral part of the CommSy Common Room.

(2) The CommSy Common Room is an archive "in progress," designed to support teaching and learning individuals and groups over a longer period of time (cf. [11]). The Common Room offers a listing of courses and extracurricular activities in current and earlier semesters, including information about contact persons, a list of Project Rooms and archived study material. Such material can be, for example, the result of a project, a research paper, a book reference, and it can be stored as file attachments.

(3) To enter either a Project- or the Common Room, CommSy offers a single entry point, the so-called CommSy Portal. In addition to its function as a convenient access to the system, the portal provides information for inexperienced users.

Starting in 1999 the CommSy system has been developed in funded and unfunded student projects and by researchers. Due to the characteristic of the development process as a student project, the process was designed to be open to any person who was interested in co-operation. Main parts of the source code were programmed with agile software development methods. For example, in 2000 the development team used the sprint technique of the extreme programming method "scrum" ([1] p. 32).

Since 2001 a publicly funded research project called WissPro[1] residing at three universities has formed the organizational frame for CommSy development and provision. The software development process has been established as an inherent part of the research project, and all members of the development team were paid by WissPro funds. The CommSy development process is based on a participatory and evolutionary design process, which is similar to the STEPS model (cf. [4], [3]).

As a consequence of modifying the organizational frame, key characteristics changed. Due to the growing team size, the stronger bonding, and the larger amount of available time for developing CommSy, the development process changed as follows:

- The openness of the development process for participants who were not involved in the WissPro-project decreased as a result of the larger programming tasks and the existence of full-time developers.
- Due to the close collaboration of the development team (located in three neighboring rooms), most design decisions were made in face-to-face meetings without (extensive) documentation.
- The agility of the process decreased and the time span of developing new releases increased because CommSy development became a paid full-time job, bigger releases were planned.

However, some characteristics of the CommSy development process stayed the same in different organizational settings:

First, there were always people with broad range of skills in the development team including social sciences and education science as well as computer science.

Second, the development process was characterized by an extensive feedback loop to ensure a co-operative learning and design process.

Third, many design decision were made in team discussions without using extensive kinds of documentation. The development team tested different techniques related to UML and tools for automatically documenting its source code, i.e. phpdoc, but for the technical support of the coordination and the documentation of the development process only CommSy itself was used in the majority of cases. The CommSy development team established

---

[1] WissPro stands for (German) "Wissenprojekt", which was a publicly funded research project by the Department of Education and Research (BMBF).

a project room called CommSy-CommSy that was used to upload important documents or to fix a date for example.

As we were forced to adjust the project's setting we intended to employ an open source strategy while at the same time preserving key characteristics of our development process. Important cornerstones of our development process were the interdisciplinary team structure, the various levels of openness, and the collaborative development.

4. Actions Taken to Migrate the Project to Open Source

To better understand the migration process, we will first lay out the reasons for switching to open source. Then we point out the goals connected with our migration process. Third, we describe in detail the actions taken especially new ways of communication that were established. Furthermore, we discuss possible role changes that have occurred.

Our reasons for changing a closed-source-project to an open-source-project were located on different levels. As pointed out in section 3, the WissPro-Project constituted only a limited organizational frame due to the limited funding of the BMBF. By the end of 2003, more than 6 people lost their employment and funding for student contracts was also no longer available. People were urged to work in new jobs. However, everybody involved in the project was eager to continue developing the system. This changed the structure of the team significantly because not everybody was able to work primarily on the development of the system during work time. Furthermore, people were scattered now in different organizational frames. We therefore had to open the development process to allow for people from different organizations to participate. In addition to that, the support for the vast amount of users had to be reorganized because there was no organization left to offer a single point of reference. On top of that, the former research project demanded a sustainable solution for providing the system permanently at least at the University of Hamburg.

Besides considering other options, we have decided to follow the path of open source software development. Our expectations regarding that change are as follows:

- Sustaining the coordinated development after the end of the research project.
- Embracing all former members of the development team even though new organizational boundaries exist.
- Introducing (further) applicants to the development team, which might not be collocated.
- Enlarging the development team through publicity.

- Bundling activities from different projects concerning CommSy development.
- Ensuring quality of the development process to create quality software.
- Incorporating new requirements.

To achieve an open source process model as described in section 2, we took a set of actions. First, we evaluated a number of open source license models and chose the GPL, effective April 1st 2003, as the underlying license. Second, we moved the development platform to the freely available SourceForge network to document the change from closed source to open source even though only internal developers were active at that time.

The next steps in our migrating process were to move components of our development process from our local platform to SourceForge successively:

- Using the bug-tracker at the SourceForge platform,
- Establishing the feature request and the task tracker at the source forge platform, and
- Shifting the project documentation to SourceForge

Then we started to separate the presentation of CommSy from WissPro by focusing on the software and its status as open source software. During WissPro we always presented CommSy and its development process as a part of the WissPro project. The nearer the expiration date of WissPro came, the development team presented CommSy as a development project of its own.

All the precedent steps were taken within the organizational frame of the WissPro research project. The following steps were taken after the WissPro project had ended. One of our main goals was to open the development process for former members working in new organizational frames on one hand, and on the other hand for new members being interested in the CommSy development. The motivation opening the development process for new team members was triggered by:

- The motivation from inside the team to enlarge the number of developers and;
- Requests of users to take part in the development process.

To support the team opening process we have shifted our process communication from predominantly direct communication in meetings to communication via electronic channels. Examples of actions taken are:

- The establishing of public accessible mailing lists like commsy.discussion@lists.sourceforge.net.
- The opening of the CommSy-CommSy to all people who are interested in the CommSy development

- The reduction of personal meetings to one meeting in two months.

5. Challenges and Problems

So far, we have discovered the following areas of challenges and problems in the migration process:

- Organizational frame;
- Team structure;
- Culture, and;
- Coordination work.

On the organizational level, we now have the situation that most contributions to the project are not paid. However, some work is paid due to partial funding or people employed by the university doing the work. The challenge lies in explaining to all participants why some work is paid while other work is not. Additionally, time management can no longer rely on standard work weeks, as people's main occupation is no longer CommSy development. The new challenge is to find and communicate a new development rhythm that is fast enough to meet outside expectations and not too fast to include everybody willing to contribute.

Regarding the team structure, we can now observe that external contributors demand their role in the development team and at the same time people working relatively close can provide different levels of intensity being engaged in the development. The challenge connected to this new team structure is to integrate different paces and equally valuing their levels of contribution.

Culture is an essential part of a development project. Part of the established culture was a clearly co-operative and discursive process. Long-term members fear that by integrating an uncontrolled new number of contributors some cultural values may no longer be shared. A natural divergence is created by paid and unpaid contributions. Justification and value of unpaid work needs to be clarified. How can co-operative development and discursive evolution of design decisions be preserved, even though the number of developers increases significantly and they are distributed?

On the level of coordination work the new challenge will be communicating with a large number of associated active developers and users. The need for coordination is much higher than before and transparency about what is going on in the development process is necessary for other developers as well as users.

Additional problems are as follows:

- How are new participants integrated in the existing team? How can they learn the project culture? How do they get information about team structure and established conventions?

- The development process slows down significantly due to the required amount of coordination and communication work. How can users and developers be satisfied with a slow development progress? How can one highlight the results and progress of the development?
- How does the development team ensure that no developer will be lost in the process?
- How does the development team ensure that new people are constantly attracted to the project?

6. Future Work and Research Questions

In the future a major task will be to observe and to analyze the consequences of our actions taken to migrate the CommSy project to an open source project. Due to the fact that the actions taken in the last year until today we only have little experience to estimate the success of these actions. We can only assume that the migrating process will succeed because we have some evidence for that. On the other hand, we also have evidence for missing special characteristics of an open source process we may never (want to) achieve.

For example, the openness of our project, meaning that anybody can use the product simply by installing it and that the process itself is open to changes seems to be fulfilled, but how far the openness for new participants can be assured, is still an open question. Our current team structure and our established ways of cooperation open up the project for further participants in general, but it will still be difficult for a new member who is in a remote location from the development team to participate in the process.

An important research question seems to be how these applicants for an open source development team could be involved and be integrated. An evaluation of some typical open source projects that are established for a longer period of time based on this question will be part of our future work. The results of this evaluation can be used to promote our migration process.

Another research question is how one can systematically migrate an existing software development process to an open source process. It seems to be an important part of our future work to analyze the actions we have taken to support the migration process to find out which are an appropriate instrument and which are not. We have to analyze in how far our project settings differ from other project settings and which consequences we may draw for the actions taken to support the migration process.

The result should answer the question what it takes to initiate an open source project.

7. References

[1] Abrahamsson, Pekka; Salo, Outi; Ronkainen, Jussi; Warsta, Juhani 2002. VTT Electronic, Espoo. 107 p. VTT Publications: 478, ISBN 951-38-6009-4; 951-38-6010-8

[2] Feller, J. and Fitzgerald, B. *A Framework Analysis of the Open Source Software Development Paradigm*. 21st Annual International Conference on Information Systems, Brisbane, Australia, 2000.

[3] Floyd, C. A. Systematic Look at Prototyping. In: Budde, R.; Kuhlenkamp, K.; Mathiassen, L.; Züllighoven, H. (Eds.): *Approaches to Prototyping*. Springer Verlag, Berlin, Heidelberg 1984, pp. 1-18

[4] Floyd, C., Reisin, F.-M. and Schmidt, G. *STEPS to software development with users*. In ESEC '89, Lecture Notes in Computer Science, no. 387, Berlin, Heidelberg. Springer, 1989, pp. 48 - 64.

[5] Hightower, R. and Lesiecki N. *Java Tools for Extreme Programming*. New York, Wiley Computer Publishing, 2002.

[6] http://sourceforge.net/ (last visited march, 8th 2004).

[7] http://www.bugzilla.net/ (last visited march, 8th 2004).

[8] http://www.eclipse.org/legal (last visited march, 4th 2004)

[9] http://www.linux.org/info/gnu.html (last visited march, 4th, 2004)

[10] Jackewitz, I., Janneck, M. and Strauss, M. CommSy: Softwareunterstützung für Wissensprojekte. In Pape, B.; Krause, D.; Oberquelle, H. (Eds) *Wissensprojekte - Gemeinschaftliches Lernen aus didaktischer, softwaretechnischer und organisatorischer Sicht*. Münster u.a.. Waxmann Verlag, 2004.

[11] Jackewitz, I., Janneck, M., Krause, D., Pape, B. and Strauss, M. Teaching Social Informatics as a Knowledge Project. In van Weert, T. J.; Munro, R. K. (Eds.) *Informatics and the Digital Society*. Boston u.a., Kluwer Academic Publishers, 2003, pp. 261 - 268.

[12] O'Reilly, T. *Lessons from Open Source Software Development*. Communications of the ACM Vol. 42 (4): 1999, pp. 32-37.

[13] Raymond, E., S. *The cathedral and the bazaar : musings on Linux and Open Source by an accidental revolutionary*, Bejing, 1999.

[14] Scacchi, W. *Understanding the Requirements of Open Source Software Development*. IEE Proceedings on Software, 149(1), 2002, pp. 24-39.