

Usability in Company Open Source Software Context - Initial Findings from an Empirical Case Study

Netta Iivari¹, Henrik Hedberg¹, and Tanja Kirves¹

¹ University of Oulu, Department of Information Processing Science
P.O.Box 3000, FIN-90014 University of Oulu, Finland
{netta.iivari, henrik.hedberg}@oulu.fi, tanjakir@mail.student.oulu.fi

Abstract. In the company open source software (OSS) development context, usability is becoming an important issue due to a growing user population, who is only interested in usable applications, not in their development. Companies try to gain a competitive advantage of OSS by utilizing available components, but the openness is difficult to achieve in the business world with licenses, patents and intellectual property rights. This paper analyses usability and user-interface (UI) development in the company OSS context through an interpretive case study in a software development unit of a large, global corporation. Our initial findings suggest that there is a need for human computer interaction specialists in the OSS context. It is revealed that with the software based on OSS, more time can be spent on usability and UI design. In the company's viewpoint, there are still many issues involved in deciding what parts of the product will be open source. Especially UI code may remain closed due to competitive advantage and patents.

1 Introduction

This paper analyzes how usability and user-interface (UI) development are dealt with in company open source software (OSS) development context. Company OSS development context refers to 'OSS 2.0', i.e. to the new, commercially viable OSS development, in which software (SW) development companies try to gain a competitive advantage of OSS [7]. SW development companies release the source code of their products and produce OS compliant licenses, and OSS companies develop their business around raising money through licensing [7], [15]. In OSS development, the source code needs to be 'available for anyone who wants to use or modify it', but in the company OSS context, there actually is a 'continuum of openness' [16: 151]. This paper focuses on commercial SW development utilizing OSS as part of the product, and potentially also releasing the source code for the OSS community to develop it further.

Nowadays OSS is targeted at a mass of users, not only for developers developing it for their own use. The user population of OSS is becoming larger, including a growing number of non-technical users, who are not interested in OSS development, but only in the resulting solution [4], [8], [14], [15], [18]. Therefore, usability and

high quality UI are becoming an important issues also in OSS context [1], [14], [19], [20]. This paper reports initial findings of an interpretive case study designed to analyze usability and UI development in company OSS development context.

2 Usability and User Interface Development

The importance of usability and UI development are emphasized especially within the field of human computer interaction (HCI). Usability refers to 'the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use' [10]. Another widely cited definition defines usability to consist of learnability, efficiency, memorability, errors and satisfaction; it should be easy to learn to use the system, its use should be efficient, it should be easy to remember how to use it, the users should not make errors or they should easily recover from them, and the use should be satisfactory. [16]

Different user-centered design (UCD) and usability engineering (UE) methodologies offer guidance for usability and UI development. Many differences can be identified in them, but they all emphasize the importance of gaining a detailed understanding the user, his/her tasks or work practices and the context of use as a basis. Afterwards, one should carefully redesign the tasks or work practices based on that understanding. User involvement in the design process is argued for. User feedback should be gathered early, and the design solution should be iterated based on the user feedback. [2], [12], [16], [17]

To carry out these activities, the HCI literature suggests hiring a group of specialists, variably labeled e.g. as usability, UCD or UE specialists. They are to carry out or facilitate the analysis of the users, their current tasks and work practices and the context of use, the user task redesign and the iterative usability evaluations. They are also to ensure active user involvement. [2], [9], [12], [16], [17]

Many challenges involved with usability and UI development in the OSS development context have already been revealed. In OSS development, the developers typically produce the SW for themselves. From the viewpoint of non technical users, usability of the OSS tends to be poor, and the development process anything but 'user centered' [1], [3], [4], [14], [19], [20]. Therefore, usability is becoming a relevant topic of research in the OSS context, even though, to date, it has not been examined much [1], [4], [14], [19].

Typically, the OSS developers do not have knowledge about the non technical users, their tasks and the context of use. In addition, no UCD/UE methodology is typically employed. Extensive gathering of user feedback and bug reporting have been highlighted as solutions in OSS context. However, it has also been reported that non-technical users may be intimidated or incapable to report bugs. [1], [3], [4], [13], [14], [19], [20] HCI specialists do not typically participate in the OSS development, and the OSS developers do not normally have the HCI knowledge and skills needed. The users can not replace HCI specialists either, because they are not trained for

developing and ensuring usability, even though they encounter the usability problems while using the system. [1], [4], [8], [13], [14], [19] [20].

Related to company OSS development, it has already been argued that large corporations that participate in OSS development can provide professional usability resources and HCI guidelines for OSS development [1]. However, there clearly is a lack of research on company OSS development, and particularly on usability and UI development in this context. This paper provides insights into this phenomenon.

3 Empirical Case

In this research effort an interpretive case study method is utilized. Case studies are suggested, if little is known about the phenomenon or current perspectives have little empirical evidence or conflict with each other [6]. There is a lack of research on usability in company OSS development context. Interpretive case study method was selected, since through it one can focus on the meanings attached to the phenomenon studied from 'the native's point of view', produce thick descriptions, and gain thorough understandings. The focus is on understanding and making sense, not explaining in the predictive sense. Theories are used as sensitizing devices. [5], [11]

The case involved in this study is a SW development unit of a large, global corporation that has been involved with OSS development already for few years. It has a strong background in usability and UI development. However, now the company is facing challenges in combining the worlds of commercial SW development, OSS development and usability and UI work. The case was selected, since it offers a rich setting to analyze the emerging 'company OSS' phenomenon.

Different kinds of empirical data have been gathered from the unit. This paper relies on data gathered by interviewing the personnel and by keeping field notes after joint meetings organized in the unit. Five persons were interviewed, including a project manager (denoted as [a]), two developers ([b] and [c]) and two HCI specialists ([e] and [f]). In addition, field notes from three meetings organized with the management of the unit were included in the analysis. This paper reports our initial findings related to how usability and UI development were dealt with in this particular case of company OSS.

The OSS was seen as a basis to build up something new in the company: *"From the viewpoint of the firm it is fast, time is not spent in implementation, but one gets everything ready made. One can concentrate on doing new things, not things that already exist."* [c] Typically, available solutions were evaluated and an OSS alternative chosen that was further developed to fit the company's needs. That way the development time was reduced significantly: *"Without this communication and library, [an application] would not have been possible. There was 6 months time, and it took 3 months, because there was support from the community. Otherwise it would have taken over a year."* [c] In addition to the support for the implementation of the low-level library, the OSS community gave feedback that was important for

usability of the product: *"From the viewpoint of UI design there are more users, a social network of knowledge."* [e] Interestingly, OSS was seen as a source of knowledge: *"[OSS is] learning and collaboration, a place to learn from developers."* [e] *"One always learns something. One sees other people's code, their expertise."* [c]

Usability and UI design were emphasized in the case: *"[Usability is] while developing UI, one of the most essential things"* [a]. *"[The firm] makes devices for people. Why would anybody buy a product that was difficult to use, compared to competitors? So market is the reason [for usability]"* [b]. *"[Usability is] extremely important: competitive edge, brand, image."* [d] There have been recent changes to emphasize the importance of usability and UI design: *"Usability and user experience are the key, the company started to invest in them. (...) In the beginning, one person took care of everything. Now there are seven persons in the team, so the role is bigger. (...) Power has been given to designers, from developers. They have the power to figure out things, they are very concerned with usability."* [e]

The difficulties involved with usability and UI design in OSS development are acknowledged: *"Developers produced the UI. It was difficult to help them, if you were not there in the beginning. (...) [I] tried to change the code in an OSS project, but it can be bureaucratic in there, they get a lot of change requests, they do not necessarily want to implement them."* [e] *"Usability is hard in OSS projects. There are no usability professionals. (...) The main problem is the usability staff in OSS projects. Except when big companies support the OSS projects. Because the firms are interested in the end user and give help."* [c]

Therefore, the support gained from companies to OSS projects is highlighted. Especially the importance of professional HCI specialists' participation in OSS projects is emphasized: *"[Interaction designers] there should be more of them willing to help, UI designers to plan and design, and developers to sponsor them."* [e] *"Get a UI designer and a usability specialist"* [b]. There should be *"open usability work: this kind of persons could be there commenting and helping (...) Usability specialists could gain reputation, if it results in good products"* [d] *"One should attract also designers, artists, end users. All areas. Developers learn from these [other areas]"* [c].

The usage of the OSS gave more time for the usability and UI design: *"One could much earlier experiment with the finished software. One did not just use own simulation. This was a positive effect."* [d] *"Late changes are possible, because own process resembles OSS projects. OSS enables late changes. There is more design and evaluation. So, it helps usability that there is code ready-made. Time is not spent on that, you start building on top of that."* [e] *"[OSS development is] faster and closer to user"* [c]. In addition, OSS gave access to users: *"In OSS development one is always in contact with the user. One knows what they want and one knows if the wishes change."* [c] *"OSS community, users -> a rich place to gather user feedback. User-centered design in the community: wishes are ranked, [the users] are used as partners in the beginning and also during the design"* [d]. Being in contact with user is *"easier and cheaper"* [e].

UI design was a key to success: *"Designers (...) always thought of users, they thought of interaction, so that the users don't get tired. We have succeeded well. (...) Best thing of the whole project is the UI."* [c] Actually, the UI part was seen as a competitive edge for the company, and that was one factor affecting not to publish the source code: *"Usability and UI are so valuable that they are not released."* [e] The chosen model was to share improvements made into the OSS library and to keep the UI part of the application closed: *"[An application is] closed source, on top of open source."* [e] That fulfilled the license terms of the original source code also: *"Openness did not spread to the UI code."* [a] Despite the situation, the company continued to interact with the community: *"There is cooperation with the community, gathering of feedback, even if it was closed source. Bugs are fixed in the OSS."* [e]

The decision had negative impacts in the community: *"If the UI is closed that causes anger in some people"* [e], but the company had also other reasons for their decision: *"People asked for the source code, but it was not given. It was an IPR risk."* [a] Everything had to be analyzed keeping the company secrets in mind: *"In the beginning, sourceforge's cvs server was used, and everything was not thought of. It was not thought of that what is secret, proprietary (...) It was moved to the firms own server, the in-house made component was put in there also. Legal check must be made. This component is not in sourceforge."* [b] The most secure way to act is to develop the product in-house first, and then release it, if it is possible: *"Bug fixes are given back, but if there are risks, a legal check is done. Everything is not let out."* [e]

In general, the licenses were seen as an issue in company OSS development: *"License defines the openness. It is difficult"* [a] *"Sometimes there are difficulties due to the licenses. One has to know what kind of code is used, what libraries are used. Usually firms know these issues, they have to know. (...) There needs to be lawyers. One has to hire them."* [c] Sometimes the procedures needed, before the open source release can be made, were seen as too expensive and complicated, when compared to the benefits: *"[The company] has a lot of patents related to [issues related to the OSS and the application], there was no desire to carry out investigations."* [a] That encourages to keep the SW as closed as possible. The result is a hybrid SW with open and closed source parts.

4 Concluding Discussion

In the company OSS context, OSS solutions are used as part of the products, but there are many issues involved in deciding what parts of the product will be open source. Especially UI code tends to remain closed. Several reasons were identified related to that. Difficulties with licenses and patents were brought up. However, interestingly, UI was also seen as a competitive advantage, which was for that reason kept closed source. It was maintained that by putting effort into the UI design, the company can beat the competitors. However, for that reason, it is unlikely that the UI parts become open source in the company OSS context.

Regarding the field of HCI, the results of this study seem promising. If usability and UI design are becoming such central concerns in SW development as indicated by this study, the field of HCI will increase its importance. There probably will be a need for HCI methods and tools, which are particularly tailored to fit the OSS development context, as well as for HCI specialists contributing to the development. Regarding the relationship between HCI and OSS, two interesting observations are emphasized. First, the results confirm that there is a need of HCI specialists in the OSS context. The interviewees emphasized the need of professional HCI specialists (some specializing in usability, others in UI design), taking care of understanding the users, their tasks and the context use, and evaluating the solution with the users.

Another interesting issue was the benefits of OSS that can be gained by the HCI specialists in the company OSS context. It was emphasized that realistic user feedback can be gathered earlier than normally if part of the product is OSS. Therefore, OSS may reduce the development time, which enables early usability evaluation with the actual SW. In addition, it was emphasized that due to the reduced development time, more time can be spent on usability and UI design, which is also a clear benefit from the HCI viewpoint. Moreover, the possibility to concentrate on innovating new things, when utilizing OSS components as a basis, was brought up as a benefit of OSS in the company context.

This paper provided initial findings from our empirical examination. More thorough analyses will be carried out in the future. Paths for future work include also deepening the understanding of usability in the company OSS context, and devising procedures that enable HCI specialists to participate in OSS communities. In general, the effects of usability work on OSS products, communities and companies, and the means to achieve those effects are interesting, momentous research topics.

References

- [1] Benson C., Müller-Prove, M., Mzourek, J.(2004): Professional usability in open source projects: GNOME, OpenOffice.org, NetBeans. Extended Abstracts of the Conference on Human Factors in Computer Systems. New York, ACM Press. Pp. 1083-1084.
- [2] Beyer, H. & Holtzblatt, K. (1998). Contextual Design: Defining Customer-Centered Systems. San Francisco: Morgan Kaufmann Publishers.
- [3] Bødker, M., Nielsen, L. & Orngreen, R. (2007): Enabling User-Centered Design Processes in Open Source Communities. In N. Aykin (ed.): Proc. Human Computer Interaction International, Part I: Usability and Internationalization. LNCS 4559. Pp. 10-18.
- [4] Cetin, G., Verzulli, D. & Frings, S. (2007): An Analysis of Involvement of HCI Experts in Distributed Software Development: Practical Issues. In D. Schuler (ed.): Proc. Human Computer Interaction International: Online Communities and Social Computing. LNCS 4564. Pp. 32-40.

- [5] Denzin, N. & Lincoln, Y. (2000): Introduction: The Discipline and Practice of Qualitative Research. In N. Denzin, Y. Lincoln (Eds.): Handbook of Qualitative Research. 2nd edition. Thousand Oaks: Sage Publications.
- [6] Eisenhardt, K. (1989): Building Theories from Case Study Research. *Academy of Management Review* 14(4): 532-550.
- [7] Fitzgerald, B. (2006): The Transformation of Open Source Software. *MIS Quarterly* 30(3), 587-598.
- [8] Frishberg, N., Dirks, A., Benson, C., Nickel, S. & Smith, S. (2002): Getting to know you: open source development meets usability. *Extended Abstracts of the Conference on Human Factors in Computer Systems*. New York: ACM Press. Pp. 932-933.
- [9] Iivari, N. (2006): Discourses on 'culture' and 'usability work' in software product development. *Acta Universitatis Ouluensis, Series A, Scientiae rerum naturalium* nro 457.
- [10] ISO 9241-11 (1998): Ergonomic requirements for office work with visual display terminals (VDT)s - Part 11 Guidance on usability. International standard.
- [11] Klein, H. & Myers, M. (1999): A Set of Principles for Conducting and Evaluating Interpretive Field Studies in Information Systems. *MIS Quarterly* 23(1), 67-94.
- [12] Mayhew, D. (1999): *The Usability Engineering Lifecycle: A practitioner's handbook for user interface design*, San Francisco: Morgan Kaufmann Publishers.
- [13] Nichols, D., Thomson, K. & Yeates, S. (2001): Usability and Open Source Software Development. *Proc. Symposium on Computer Human Interaction*, Palmerston North, New Zealand, July 6. Pp. 49-54.
- [14] Nichols, D. & Twidale, M. (2006): Usability Processes in Open Source Projects. *Software Process Improvement and Practice* 11, 149-162.
- [15] Niederman, F., Davis, A. Greiner, M., Wynn, D. & York, P. (2006): A Research Agenda for Studying Open Source I: A Multilevel Framework. *Communication of the Association for Information Systems* 18, 129-149.
- [16] Nielsen, J. (1993): *Usability Engineering*, Boston: Academic Press.
- [17] Rosson, M. & Carroll, J. (2002): *Usability Engineering: Scenario-based Development of Human-Computer Interaction*, San Francisco: Morgan-Kaufman.
- [18] Viorres, N., Xenofon, P., Stavrakis, M., Vlanhogiannis, E., Koutsabasis, P. & Darzentas J. (2007): Major HCI Challenges for Open Source Software Adoption and Development. In D. Schuler (Ed.): *Proc. Human Computer Interaction International: Online Communities and Social Computing*. LNCS 4564. Pp. 455-464.
- [19] Zhao, L. & Deek, F. (2005): Improving Open Source Software Usability. *Proc. 11th Americas Conference on Information Systems*, Omaha, USA, August 11-14. Pp. 923-928.
- [20] Zhao, L. & Deek, F. (2006): Exploratory inspection: a learning model for improving open source software usability. *Extended Abstracts of the Conference on Human Factors in Computer Systems*. New York: ACM Press. Pp. 1589-1594.

