Behaviour & Information Technology
Vol. 30, No. 3, May–June 2011, 309–323

Taylor & Francis
Taylor & Francis Group

Participatory design in OSS development: interpretive case studies in company and community OSS development contexts

Netta Iivari*

Department of Information Processing Science, University of Oulu, PO Box 3000, 90014 Oulu, Finland

(*Received 27 July 2009; final version received 15 June 2010*)

This article examines distributed participatory design in open source software (OSS) development. User participation is becoming a relevant topic of research in the OSS development context. Though it has not been examined much to date, the OSS development context has been argued to advocate a particular type of participatory design, which can now be scrutinised in its natural setting as it evolves. Two interpretive case studies on user participation in OSS development are included in this article. The first examines a traditional community OSS development project; the second concentrates on the company OSS development context, the case being a software development unit of a global corporation involved in OSS development. Through analysis of the cases, different forms of participatory design (PD), especially of distributed PD, are identified. Distributed PD is interpreted to include gaining an understanding of users' current practices, redesigning them together with users and gathering feedback from users related to the solutions. Different kinds of roles are available to users, as well as to for intermediaries 'representing users'. Especially, the importance of online forum-based and intermediary-driven PD is emphasised in this article. Implications for PD and OSS research and practice are considered.

Keywords: participatory design; user participation; open source software development; distributed development; interpretive case study

1. Introduction

This article empirically examines practices that can be characterised as participatory design (PD) in the open source software (OSS) development context – as they evolve in their natural setting. The specific research question of this study is the following: 'What kind of PD practices have emerged and evolved in distributed OSS development?' OSS development is a current object of study in information systems (IS), human computer interaction (HCI) and PD research communities (Binder *et al*. 2002, Frishberg *et al*. 2002, Nichols and Twidale 2003, Benson *et al*. 2004, Clement and van den Besselaar 2004, Luke *et al*. 2004, Fitzgerald 2006, Niederman *et al*. 2006, Zhao and Deek 2006, Cetin *et al*. 2007). Generally, it necessitates the source code to be 'available for anyone who wants to use or modify it' (Niederman *et al*. 2006, p. 131). Developers are typically also users of the OSS they develop; that is, they scratch their own itch.[1] Therefore, the distinction between user and developer is unclear in OSS development, and all users should be considered potential developers. However, it has been reported that up to 90% of users might be 'passive users' who want only to use the OSS without taking any part in programming (Ye and Kishida 2003, Nichols and Twidale 2006, Barcellini *et al*. 2009). In this article, these 'passive users' are called 'non-developer users'.

In this article, it is acknowledged that, in addition to the earlier-described model of community OSS development, companies are getting involved as well (Fitzgerald 2006, Niederman *et al*. 2006, Iivari *et al*. 2008), utilising OSS solutions as parts of their products or as development tools, and potentially also releasing the source code of their products to OSS communities for future development (Fitzgerald 2006, Niederman *et al*. 2006, Iivari *et al*. 2008). It has been argued that this kind of new, 'commercially viable OSS development' will be highly influential in the future software landscape, and currently there is a lack of research examining this emerging phenomenon (Fitzgerald 2006). This article will focus both on traditional community OSS development and on company OSS development; in both situations, it examines the development of solutions targeted at a mass of users who do not all have technical background or interest in developing solutions. In this situation, it is critical that the developers take the needs of non-developer users into account.

Generally, user participation in systems design is strongly recommended in IS, HCI and PD research communities. OSS development has been argued to

*Email: netta.iivari@oulu.fi

ISSN 0144-929X print/ISSN 1362-3001 online
© 2011 Taylor & Francis
DOI: 10.1080/0144929X.2010.503351
http://www.informaworld.com

have a user-driven development model; user participation and PD have been emphasised as specific strengths of OSS development (Zhao and Deek 2005, 2006, Barcellini *et al*. 2008, 2009, Titlestad *et al*. 2009). However, these studies discuss participation of developer users. Nowadays, on the other hand, certain OSS solutions have become very popular, and the non-developer user population is constantly growing (Frishberg *et al*. 2002, Scacchi 2002, Franke and von Hippel 2003, Ye and Kishida 2003, Nichols and Twidale, 2006, Niederman *et al*. 2006, Cetin *et al*. 2007, Viorres *et al*. 2007). For these users, usability of OSS tends to be poor, because developers traditionally scratched their own itches and usability was not a major concern. When the developers no longer equal users, usability and the needs of non-developer users become important issues to be addressed (Nichols and Twidale 2003, 2006, Benson *et al*. 2004, Zhao and Deek 2005, 2006, Andreasen *et al*. 2006, Bødker *et al*. 2007, Cetin *et al*. 2007, Bach *et al*. 2009). Therefore, this article addresses a significant and topical problem related to enabling non-developer user participation in OSS development.

It has been argued that companies involved in OSS development could take responsibility to ensure user participation (Frishberg *et al*. 2002, Nichols and Twidale 2003, 2006, Benson *et al*. 2004, Andreasen *et al*. 2006, Iivari *et al*. 2008); because companies face pressures to deliver useful, usable solutions for end users, it might be natural for them to take care of user contact during development (Markus and Mao 2004, Iivari 2006). However, there is a lack of empirical research related to the feasibility of and complexities imposed by company involvement in OSS development. Researchers have already argued that this might be quite challenging (Heikinheimo and Kuusisto 2004, Fitzgerald 2006, Iivari *et al*. 2008). Some research addressing PD in OSS development already exists as well, where researchers have suggested solutions or experimented with PD in OSS development (Nichols *et al*. 2003, Nichols and Twidale 2003, 2006, Luke *et al*. 2004, Zhao and Deek 2005, 2006, Gumm *et al*. 2006, Bødker *et al*. 2007, Cetin *et al*. 2007, Obendorf *et al*. 2009, Titlestad *et al*. 2009). Contrary to those studies, this article focuses only on interpretive understanding of the dynamics between OSS and PD, providing insights into how distributed PD practices evolve in natural OSS settings without researcher intervention. Specific to this study is also a focus on the participation of non-developer users.

HCI and PD research has created a number of methods and tools supporting PD, but these methods and tools typically assume colocated participants, which is not the case in OSS development (Nichols and Twidale 2003, 2006, Gumm *et al*. 2006, Cetin *et al*. 2007, Obendorf *et al*. 2009). OSS development is a form of distributed development, which poses additional challenges to user participation. OSS projects are typically carried out by geographically distributed developers, who rely on the Internet for communication and cooperation: mailing lists, discussion forums, bug reporting, feature request and version control systems (Ljungberg 2000, Mockus *et al*. 2002, Scacchi 2002, Frank and von Hippel 2003, Lakhani and von Hippel 2003, Ye and Kishida 2003, Gumm 2006, Gumm *et al*. 2006, Nichols and Twidale 2006, Barcellini *et al*. 2008, 2009, Obendorf *et al*. 2009). OSS development may be distributed physically, organisationally and temporally: developers around the world may work in projects, sometimes several of them, on a voluntary basis without organisational ties or contracts (Gumm 2006).

This article is organised as follows: the next section characterises what is meant by PD practices and what kinds of PD practices have already been identified in OSS development. The third section presents the research method utilised, the cases involved in this study, and the procedures of data gathering and analysis. The fourth section outlines the results of the empirical inquiries. Thereafter, the article discusses the implications of the findings for PD and OSS development research and practice: what can be learned from these cases by PD and OSS researchers, how these PD practices could be further supported in OSS development and in other distributed contexts, and what could be provided by researchers to improve PD and OSS development practice. The final section summarises the results, identifies their limitations, and outlines paths for future work.

2. Participatory design and OSS development

2.1. *Participatory design*[2]

PD research is based on the ideal of empowerment of the skilled worker in systems design. The field has been greatly influenced by the earlier Scandinavian collective resources tradition, which posited workplace democracy as its goal and maintained that union involvement is necessary in the development of computer systems for work setting. The tradition assumed an inevitable conflict between capital and labour, strongly positioning itself on the latter side. The point was to take care of the rights of workers during systems design. However, the political emphasis started to decrease during the 1980s, and interest started to turn to supporting the PD process, which was now conceptualised as cooperative work where people with different competencies (workers, designers and researcher-designers) jointly create new work practices and technologies and appreciate each other's

expertise and skills as valuable input (Greenbaum and Kyng 1991, Schuler and Namioka 1993, Bjerknes and Bratteteig 1995, Kensing and Blomberg 1998, Asaro 2000, Iivari *et al*. 2009).

User participation in systems design is a fundamental condition in the PD tradition, but there is ambiguity in defining the aim of the participation. The goal may be democratic empowerment, which maintains that the workers have the right to participate in decision making related to their work (including systems design), or it may be functional empowerment, which maintains that the workers have the right be able to do their work effectively and efficiently (i.e. useful and usable systems are to be produced for them), and their participation in the design process is needed to achieve this (Clement 1994). Clearly, democratic empowerment of workers was the original goal of the PD tradition, but, as mentioned, political issues have decreased in importance during the last decades. Therefore, one can argue that in the earlier PD literature a participative role for the workers was strongly advocated: the workers (i.e. the intended users) were positioned as active participants in the design process, and it was maintained that they needed decision-making power regarding the solution (Damodaran 1996). Recently, however, functional empowerment of the workers and informative and consultative roles for them have gained a legitimate position: the intended users are positioned as providers of information and as objects of observation, or as commentators of predefined design solutions, without decision-making power regarding the solution (Damodaran 1996, Kujala 2003).

PD has had a strong design orientation, i.e. emphasis on cooperative design work where users and designers both contribute to the design of future technologies; however, criticism of this has also emerged. It has been pointed out that during this cooperative design process, explicit analysis and understanding of the users' current work practices and settings may be neglected (e.g. Karasti 2001). Researchers have suggested supplementing PD methods with insights developed in the Computer-Supported Cooperative Work (CSCW) research field, in which much effort has been put into gaining and articulating detailed, ethnographic understandings of users' work practices and contexts. Indeed, the field of CSCW has clearly contributed to the PD tradition in the past two decades, ethnography oriented researchers providing ways to gain thorough understandings of users' current practices and combine them with cooperative design (see e.g. Kensing and Blomberg 1998, Karasti 2001).

In addition to direct user participation in the design process, a number of kinds of intermediaries have been identified in the literature, starting from the unions representing the workers discussed earlier. One can generally say that these intermediates either represent the users in the design process or facilitate cooperation between users and designers. Many differences can be identified in relation to these intermediaries: for example they may be in formal or informal positions; they may be hired or have adopted the position in an ad hoc manner in situ (Iivari *et al*. 2009). The intermediaries are called user representatives or change agents in IS research, HCI/usability/user-centred design specialists in HCI research, researcher-designers in PD research and ethnographers in CSCW research (Iivari 2006, Tuovila and Iivari 2007, Iivari *et al*. 2009). HCI specialists are expected to 'represent the users' in the design process: that is they are charged with gaining an understanding of the users and their work practices, and to deliver this understanding to the design (Iivari 2006), just as ethnographers are expected to provide ethnographic data to inform systems design (Karasti 2001). The researcher-designers identified in the PD literature, on the other hand, are expected to facilitate user-designer cooperation; they are expected to enable user participation and ensure that everyone feels comfortable participating, everyone's expertise being equally appreciated (Karasti 2001, Tuovila and Iivari 2007, Iivari *et al*. 2009).

It may be that these intermediaries are only allowed to act as providers of information (i.e. ethnographers or HCI specialists delivering data to design based on their empirical inquiries) or as commentators of predefined design solutions (i.e. HCI specialists carrying out different kinds of usability evaluations), but they might also be allowed to take on participatory roles, actively taking part in the design process with decision-making power regarding design solutions (e.g. researcher-designers orchestrating design sessions with users and designers, each having a say regarding the design solution) (Iivari 2006, Tuovila and Iivari 2007). Regarding both users and the intermediaries 'representing' them, potential difficulties have been reported in them having any impact on the solution (see e.g. Borgholm and Madsen 1999, Bødker and Buur 2002, Hornbaek and Stage 2006, Iivari 2006). As a solution to this problem, both integration and separation approaches have been suggested (see Hornbaek and Stage 2006); generally, the literature seems to advocate integration, including the participative role for users and intermediaries who are thus expected to actively cooperate with designers to affect the solution (Borgholm and Madsen 1999, Bødker and Buur 2002, Hornbaek and Stage 2006, Iivari 2006).

Altogether, in this article, PD is understood to include both gaining an understanding of users'

current practices and collaborating with users to design new practices. Feedback on solutions made is to be gathered from the users; therefore, PD includes understanding, designing and evaluating activities. The aim of the activities is to improve the functionality and/or usability of the solution. All the roles identified earlier are relevant for the users: they are to play an informative role, acting as objects of observation and providers of information, a commentator role, commenting on predefined design solutions, and a co-designer role, producing the solutions along with the designers, having decision-making power regarding the solution. Different kinds of intermediaries are potentially involved in the process as well, playing roles similar to those available to users: they are responsible for facilitating understanding of users' current practices, designing new practices and technologies, and gathering user feedback. However, the ultimate goal of PD, empowerment of skilled users, emphasises the importance of participation in decision making, which makes the participative role for both users and intermediaries the ideal one.

2.2. *Participatory design in OSS development*

Although users and developers cannot be strictly separated in OSS development, the focus of this article will be on users who are not interested in developing the OSS. From their point of view, as mentioned, usability tends to be poor, and the development process anything but 'user-centred'; nevertheless, the OSS environment offers a multitude of means for user participation. OSS development strongly relies on Internet tools for communication and cooperation – e.g. mailing lists, discussion forums and bug-reporting and feature request systems, in which users can ask for help, request features and report problems, and in which developers can gather feedback and provide support (Scacchi 2002, Frank and von Hippel 2003, Lakhani and von Hippel 2003, Ye and Kishida 2003, Barcellini *et al*. 2008).

However, it has also been reported that users may be unable to fix or report bugs, and that communicating with developers through mailing lists and bug-reporting systems might be difficult and scary for non-developer users (Nichols *et al*. 2003, Nichols and Twidale 2003, 2006, Benson *et al*. 2004, Cetin *et al*. 2007). Usability problem reporting is especially challenging, because the problem (and the solution) might be complex and intertwined, and difficult to explain textually (Nichols and Twidale 2003, 2006, Zhao and Deek 2005, Bach *et al*. 2009). Usability evaluation has been indicated as one way to integrate non-developer users into development; remote usability testing and user data gathering, as well as usability inspections by the users themselves, have been recommended (Nichols *et al*. 2003, Nichols and Twidale 2003, 2006, Zhao and Deek 2005, 2006, Andreasen *et al*. 2006). Design and requirements negotiation and collaboration tend to be difficult in OSS development, especially when textual descriptions are not enough (Nichols and Twidale 2003, 2006, Cetin *et al*. 2007); design by blogs and discussion infrastructure for user- and usability-related issues have been suggested for OSS development (Nichols and Twidale 2003, 2006, Bødker *et al*. 2007, Bach *et al*. 2009).

The literature also brings up the importance of different intermediaries[3] to participate in OSS development and address user-developer communication (Nichols and Twidale 2003, Benson *et al*. 2004, Zhao and Deek 2005, Andreasen *et al*. 2006, Cetin *et al*. 2007, Gumm *et al*. 2006, Barcellini *et al*. 2008, 2009, Bach and Carroll 2009, Bach *et al*. 2009, Obendorf *et al*. 2009, Titlestad *et al*. 2009). Some of these studies focus on HCI specialists (Nichols and Twidale 2003, Benson *et al*. 2004, Zhao and Deek 2005, Andreasen *et al*. 2006, Cetin *et al*. 2007, Bach and Carroll 2009, Bach *et al*. 2009), pointing out that companies involved in OSS development can provide HCI resources and guidelines for OSS projects (Frishberg *et al*. 2002, Nichols and Twidale 2003, 2006, Benson *et al*. 2004, Zhao and Deek 2005, Andreasen *et al*. 2006, Iivari *et al*. 2008, Bach *et al*. 2009). HCI specialists could take responsibility not only for organising user evaluations but also for carrying out user studies and iterating early design solutions through paper prototyping (Cetin *et al*. 2007). However, HCI specialists do not typically participate in OSS development, and if they do they may still be isolated and without decision-making power, their work not necessarily having any effect on the solution (Nichols and Twidale 2003, 2006, Benson *et al*. 2004, Andreasen *et al*. 2006, Bødker *et al*. 2007, Cetin *et al*. 2007, Viorres *et al*. 2007).

One clear challenge for successful user and HCI specialist participation is the distributed nature of OSS development. This article will examine the different forms of distributed PD that have emerged in the OSS development context; this has already been acknowledged as a current and interesting research topic (see e.g. Nichols and Twidale 2003, 2006, Cetin *et al*. 2007, Barcellini *et al*. 2008, Obendorf *et al*. 2009), and this study addresses it through two case studies representing the context. In both cases, an interest in users and in the usability of their solution is evident, and the solutions are targeted at non-technical users as well.

3. Methodological considerations

This article relies on two interpretive case studies addressing distributed PD in the OSS development

context (Iivari *et al*. 2008, Iivari 2009). Generally, interpretive case studies are designed for understanding and for making sense of the world; thus, the goal is not to explain in the predictive sense. Instead, the research interest is in the meanings people attach to the phenomena studied. Theories may be used, but they act as sensitising devices; the aim is not to falsify them. The overall aim is to enquire from the native's point of view, to end up with thick descriptions and to arrive at thorough understandings of particular cases. (Klein and Myers 1999, Walsham 1995, Denzin and Lincoln 2000).

This article is based on the findings of two separate interpretive case studies, both examining PD practices in the OSS development context as they evolve in a natural setting, without researcher intervention. These divergent studies add variety to the analysis of PD in OSS development: the first addresses PD in a traditional community-based OSS development project, the second one from the viewpoint of a company cooperating with traditional community-based OSS development projects. The cases, the data gathering and analysis strategies involved, and the motivation for choosing these cases are presented next.

The first case is an OSS development project for a media application for users who do not necessarily have any technical background. The project is a small but active one, with nine developers listed on sourceforge.net. It has shown clear interest in improving the usability of the solution and in involving users to do so; it is listed on a website requesting usability support from HCI specialists for OSS projects. In addition, a usability discussion forum has been established on the project website, asking users of the OSS to take part in further improving the program by offering suggestions on how to improve usability and outlining annoying issues in the current user interface (UI). The discussion forum is the place where non-developer media application users are invited to take part in the project; therefore, this article examines the communication – altogether around 1600 posts, 400 topics and 600 message sender nicknames – taking place there.

All the posts from the forum were printed out for this analysis. The researcher (author of the article) focused on the identification of different PD practices in use in this OSS development project. All posts were read through and, through initial inductive analysis, those dealing with PD were selected for more detailed examination. Afterwards, the literature outlined in Section 2 was used to categorise the different practices and user/intermediary roles; the researcher enquired whether the PD practices were concerned with understanding the users and their current practices, with redesigning them, or with evaluating the solutions, and whether the users and possible intermediaries representing them played informative, consultative or participative roles.

It is already known that non-developer users are typically neglected in OSS projects, but in larger ones or ones where companies are involved usability resources might be available to 'represent the user' in the development (see Frishberg *et al*. 2002, Benson *et al*. 2004, Nichols and Twidale 2006, Iivari *et al*. 2008). The project in question is small and without corporate resources, thus providing interesting data on how an OSS project initially tries to deal with its non-developer users. All user-developer interaction takes place on the project website, thanks to which all PD practices can be considered as distributed PD.

The second case provides data related to company involvement in OSS development. The case is a software development unit of a large global corporation that has been involved with OSS development for a few years. The unit has carried out certain projects in cooperation with OSS projects; this analysis focuses on those that have developed applications for end-users who do not necessarily have programming skills or interest. These projects have also happened to develop media applications. This company develops products for mass consumer markets, thanks to which the potential user population might be quite large in the future. OSS solutions are used as a basis for application development; applications are sometimes closed-source and sometimes open source. The personnel of the projects analysed in this article consist of developers, testers, HCI specialists and managers, all hired by the company.

From a PD viewpoint, the second case provides data related to the intermediaries 'representing the users' – i.e. HCI specialists. The unit has a strong background in usability and UI development; the users involved in development have been invited to the working location or have been members of OSS communities, and are thus distributed physically, organisationally and temporally. The external OSS communities are utilised as providers of OSS components and as providers of feedback and design ideas for application in the later phase of the development. The case thus utilises OSS communities in roles other than those of mere technology providers, and enlightens us on the ways OSS communities can be used in PD. The project is carried out face-to-face in one location in this case, although this is not always the case in the company's projects. The project personnel communicate with OSS communities through email and discussion forums. The article focuses in particular on the interaction between the company and the distributed OSS users. This case was selected because it offers a rich setting in which to analyse the emerging

company OSS development phenomenon with particular emphasis on PD.

Different kinds of empirical data were gathered. The main method was theme interviews with people representing different professional backgrounds: a manager, two developers and two HCI specialists. All interviewees were selected by a high-level unit manager. The company is in a constant process of reorganisation, so people involved in the projects were moved to different organisational units after the projects ended; the manager found two knowledgeable developers, two knowledgeable HCI specialists and one knowledgeable manager to be interviewed. The researchers had no say in the selection process. All interviews were tape-recorded and the tapes were transcribed. More data were gathered through field notes after several meetings with the managers in the unit; additional material includes blogs written by unit personnel, the company's OSS-related websites, and those of the associated OSS communities. After all the material was rendered in written form, the author of the article focused on identifying the PD practices in these company-OSS collaboration projects. The material was read through and parts dealing with PD were selected for further examination; the literature outlined in Section 2 was again used as a sensitising device to categorise the different practices and user/intermediary roles, as described in connection with the first case.

4. Empirical insights

4.1. Participatory design in community OSS development

This section will illustrate how users and intermediaries 'representing other users' contribute to OSS development in informative, participative and especially consultative roles, taking part in understanding, designing, and especially evaluating activities. This case focuses on distributed PD practices, as shown in an OSS usability discussion forum inviting end users to help improve the program.

Though the forum name implies an aim of gathering usability feedback and improvement ideas, there are a variety of messages posted there. Many can be labelled as feature requests (499) or bug reports (300) (Iivari 2009), typical messages in OSS discussion forums (see e.g. Lakhani and von Hippel 2003, Ye and Kishida 2003). In the former, different kinds of features are requested, from issues of appearance (how it ought to look) to behaviour (how it ought to behave) and integration (with what it ought to operate). Options for certain appearance or behaviour customisations are also requested. In bug reports, on the other hand, different kinds of problems are expressed: the sender does not know how to use, or does not like, a particular feature of the OSS, or the OSS altogether. These messages seek improvements to functionality or usability.

The existence of a discussion forum dedicated to usability issues, initiated by asking users to participate, indicates that developers have invited users into a consultative role (Damodaran 1996) on the project: to provide improvement ideas and feedback. All the feature requests and problem reports can be seen as evidence of this. The messages also offer some data on the users and their work practices: in some, the senders describe at length their needs, preferences, characteristics, usage habits and contexts, and steps they had taken in trying to use the OSS (see Iivari 2009). 'I truly hope this does not sound like a complaint, because this is not. However, I recently began using [the application], therefore I am here describing a painful process of how one is learning to use [the application]. Besides, I am devoted to help to produce a really great [application] for Linux. These comments are likely characteristic of the experience of several other people who try to use [the application] the first time. [A long description of the problems and of the steps the sender had taken in trying to accomplish his goals is offered.] I am going to continue to play with this beast and check what other things appear as confusing. Once more, this is a great piece of software, but the interface is just somewhat confusing for new users' (User). On the whole, however, there are few of this type of message.

In other messages, senders do not talk about their own characteristics or problems, but 'speak on behalf of' other user groups, thereby constructing 'novice,' 'non-technical' and 'typical' users (see Iivari 2009). In some cases, these user groups are discussed in the abstract, without any empirical evidence, offering only opinions or stereotypes: 'How about double clicking in the (...) window to select and to press ok? This would make it faster to use and also simpler for novice users I think' (User). Conversely, some senders do provide empirical evidence, describing how they have observed others or how others have described their behaviour, problems or needs. 'I have a couple of usability observations I'd like to introduce if that's ok? Honestly, this is the best [feature] but I can imagine some problems (I have seen in what kind of problems my luddite, Windows and Mac using friends have gotten themselves to in case they try to use [the application]; [a list of problems is outlined]' (User). Still, few messages provide detailed information about other users' characteristics, tasks, goals or usage contexts. It can nonetheless be argued that these senders, describing issues related to other users but not claiming to resemble them, have adopted the position of an intermediary delivering user data to the

developers, once again playing a consultative role (cf. Damodaran 1996).

Other users took a participatory role (Damodaran 1996) in the project. These users took active part in development, despite not having decision-making power regarding the solution: they contributed code, mock-ups and design suggestions in the discussion forum: 'Hello! I request two things to [the application]: [a feature] and [a feature]. (...). This can be done with [the application] by right clicking (...), but there are too many clicks from my point of view 😊. I have made a patch to handle it [a link is provided]' (User). 'I have created the following lousy mock-up: [a link to the mock up is provided]. Reasoning: the main thing people do with [the application] is [something], but it feels odd that the [...] button [...] is at the bottom of the screen' (User). This type of user participation has motivated the labelling of OSS development as 'user-driven' (cf. Zhao and Deek 2006). However, these users are not likely to aptly represent those non-developer users most in need of PD in OSS development. Probably rather technically competent users, they are capable of contributing by utilising the means already available; furthermore, not even these users are allowed to make decisions regarding the solution. The possible project leader and 'core team' of developers tend to make all decisions related to what to include in the code base in OSS projects (Ye and Kishida 2003). Such was the case in this project. The developers invited the users to contribute, but they alone decided what to include in the solution. The developers may reply to the users' design solutions or remarks that they 'simply don't like them' – because they are 'too cluttered' or they just 'suck'. On the other hand, they may implement certain ideas quickly, even in the same day, replying to the user simply that it was a good idea.

4.2. Participatory design in company OSS development

This section will illustrate how users contribute to OSS development in participative and consultative roles, taking part in designing and evaluating activities, and how HCI specialists 'representing' them contribute in informative, participative and consultative roles by taking part in understanding, designing and evaluating activities. This case study focuses on the ways PD is practised in a software development unit of a company cooperating with OSS communities; this specific unit is responsible for this cooperation from the company's perspective. Few company-OSS projects are carried out in the unit, and this analysis focuses on those with user populations including non-developer users. Two media application-development projects were particularly relevant from this viewpoint.

The unit utilises OSS solutions as part of the company's commercial products; OSS is used as a basis on which applications are developed. In one project, the UI level remained closed-source, while in the other the whole application's source code was eventually released. Keeping the UI closed-source is motivated by the fact that usable UIs are extremely important from a business viewpoint: '[Usability is] while developing the UI, it should be one of the most essential things' (Manager). '[The firm] makes devices for people. (...) Why would anybody buy a product (...) that costs as much as the competitors, but is 10 times more difficult to use? So maybe market is the answer [for usability]' (Developer). 'Usability and user experience are becoming (...) the key in today's market, the company started to invest a lot more in usability and user experience. (...) In the beginning, for a couple of years, there was just one guy taking care of everything of usability and user interaction and now there are seven' (HCI specialist). 'The user interface code was also not released. Actually the company decided that usability is so important (...) that the user interface code is actually the value of the whole thing, so that's why it was not released' (HCI specialist).

Development work is carried out similarly in the projects. First, a user group for which the solution was developed is settled: 'The first [phase] (...) tries to understand the people involved, the customers' needs and clients' needs. (...) We try to do some benchmarks and a persona, at least a simple persona, just to keep track of everything; we don't go too much deep in personas' (HCI specialist). Actual users are not contacted during this phase: 'Well, at first we sat down and thought that we are now making this kind [application] and thought that the display is this kind [display]. (...) So, this is the starting point and otherwise we have a lot of freedom to do what we want' (Manager). Afterwards, users' new practices and UI design solutions are produced by the HCI specialists: 'We started by building use cases in a user-centred way. We described the normal use situations that can be related to the software like this. After that we moved to producing rough UI' (HCI specialist). 'We sat there together for an hour and brainstormed and then they [the HCI specialists] left and came back within a couple of days and asked that would it be like this?' (Manager). The developers and the HCI make collaborative decisions about the design solutions: 'When we were able to produce something finished by our opinion, we went to present it to the developers and asked that is this possible?' (HCI specialist) 'It was with the implementation team like we just went there and asked. We did not produce a list to be discussed in a meeting. It was like active communication all the time' (HCI specialist).

Therefore, the HCI specialists play a participative role (cf. Damodaran 1996), 'representing the users' in development.

Both expert evaluations relying on the knowledge of the HCI specialists and empirical testing with real or potential users are carried out: 'First we produce rough UI (...) and afterwards (...) we carry out expert evaluation. In them, we use one or more usability specialists and modify the UI according to them, and then move it to a more detailed level. After that, we make a simulation of the UI and carry out a traditional usability test in a laboratory with a sufficient amount of users, from 6 to 10 per iteration' (HCI specialist). OSS users are now contacted to provide user feedback in a fast and easy way: 'What we did was: hi, let's do a prototype in, for example, couple of weeks and see how it goes. The user interface is very, very, very simple. What we did was to make it and release it. (...) Involvement was all through emailing community: what do you like to see, what do you like to have? (...) We did use the community to find out more and to find out improvements' (HCI specialist). 'We gained comments of it, when we read the Internet forums afterwards' (HCI specialist). The OSS projects' discussion forums are thus utilised in a similar way as in the community OSS project described earlier. However, it is also acknowledged that some kind of classifying and ranking of messages is needed: 'These open source software, they create a lot of communities, so it's a really, really rich place to gather user feedback. (...) So, we take this kind of user-centred design as much as we can into this user community-centred design. Of course, there are a lot of requests, sometimes it's very personal, so we try to rank and classify them, but what we try to do is to use open source users, even the developers, as partners in the beginning and in the design phase.' (HCI specialists). The users in this case are also positioned in a consultative role (Damodaran 1996), commenting on predefined design solutions: 'We do involve the end users, but only after prototype' (HCI specialist).

Here, it can be emphasised that OSS usage facilitates involving the actual users earlier in development: 'In the open source project, you actually get the final users much faster for you from all around the world. For usability perspective, I would say that it's simpler and cheaper to do in the open source' (HCI specialist). 'We were able during much earlier phase to handle the finished software. (...) Usually, it takes pretty long time to make a simulation of it. (...) Here, we were able to experiment in practice what it looks like and how it feels like in real use' (HCI specialist). Some users are also allowed a participative role (Damodaran 1996) in development, some even contributing code to the application: 'We gave all the code to the community and gained their acceptance. [The application] has been made expandable. Users can make plug-ins. The infrastructure has been built' (Developer). On the other hand, the developers contribute back to the OSS communities: 'In practice, we evaluated it and checked it and spotted pure bugs and of course we fixed them. In the beginning we did it, the project was in source forge and we asked for developer rights and were able to contribute' (Manager). 'Sometimes we do closed source projects with open source components, but sometimes we are also helping a lot the communities by fixing the bugs and not putting it in our products only.' (HCI specialist)

The communities are not necessarily enthusiastic about the developers' contributions. '[An OSS project] is not very open. There is a maintainer, it is his personal project and he decides what he wants. If you want a feature and start doing it, he does not necessarily let you. It is not open for new contributions. It is a bunch of friends who do it together. I sent patches to there but it is not open for everybody, not interested in new ideas or features. The maintainer makes the decisions. If he does not want it, he does not allow it, like just forget it. (...) I took patches and applied them, made a lot of changes, made it different, but only one was accepted, not the others to the official tree' (Developer). The same applies to HCI specialists' contributions to OSS projects: 'I think that usability is very hard when we talk about open source. (...) A lot of open source projects suffer from issue of not having good professionals related to usability and sometimes to UI. (...) I tried like entering some projects that I use their software: hi, I think this software needs a little bit of love. But, it is quite, how can I say, bureaucratic sometimes and quite political. (...) Of course they receive a lot of suggestions. (...) I tried to come up with some solution for something and they said it's not going to be done' (HCI specialist). Therefore, allowing users (or their representatives) to take part in the development is one thing, while making the OSS projects account for the contributions of the company's developers and HCI specialists is another. Within the company, the decision-making power is in the hands of the developers and HCI specialists; in the OSS projects, it is in the hands of the OSS developers.

5. Discussion

5.1. Characterising Participatory design in OSS development

This article empirically examined PD in the OSS development context. It was acknowledged that PD includes understanding, designing and evaluating activities, aiming to improve the functionality and usability of the solution. The activities may entail

informative, consultative or participative roles for the users, who may also be represented by different kinds of intermediaries playing informative, consultative or participative roles. The empirical findings of the PD practices identified in OSS development are summarised in Table 1.

As Table 1 illustrates, users participated in OSS development in many different ways. In the OSS project, they took on informative, consultative and participative roles, though the emphasis was clearly on the consultative. The users rarely offered insights into their characteristics or current practices, nor did they have any decision-making power regarding the solution (cf. Ye and Kishida 2003), due to which a participative role was not fully realised. On the other hand, there were also intermediaries 'representing the users' in the project; some users represented others, particularly the novice, non-technical and typical users, in their messages in the discussion forum. These intermediaries acquired the consultative role, commenting on predefined design solutions from the viewpoint of these represented users but not producing any design solutions and very rarely discussing their characteristics or current practices.

In the company OSS development context, on the other hand, the users took almost solely the consultative role, providing feedback on existing solutions. At the same time, the intermediaries 'representing the users', professional HCI specialists, held a very important position, taking on informative, consultative and participative roles in development. In the informative role, they helped settle the target user group for whom the solution was developed, based on their (HCI, previously acquired domain) expertise; no empirical user studies were carried out. The HCI specialists also took active part in decision-making in the design of new practices and technological solutions. They also acted in the consultative role by commenting on predefined design solutions, relying either on general HCI knowledge in their expert evaluations or on the empirical user testing data they had gathered.

Table 1 illustrates that understanding, designing and evaluating activities could be identified from the OSS development context, but that they had many shortcomings. In the OSS development project there seemed to be no explicit effort to understand or design activities ('design' in this case referring to the redesign of users' practices). Technical solutions were designed

Table 1. Participatory design in OSS development.

	Community OSS development	Company OSS development
Understanding the users	No explicit effort to understand the users, few mentions of users' characteristics or practices provided by users and the intermediaries 'representing' them.	HCI specialists 'representing the users' provide HCI knowledge on development.
Designing new practices and technologies	No explicit design of users' new practices but the 'technically capable users' take part in the technology design process, but without decision-making power.	HCI specialists 'representing the users' design users' new practices and technological solutions, and have decision-making power regarding the solution.
Evaluating design solutions	Users and the intermediaries 'representing' them provide empirical user feedback.	Users provide empirical feedback, HCI specialists 'representing the users' provide feedback based on their HCI expertise and empirical user feedback.
Users in informative role	Few users provide some data related to their characteristics or practices.	–
Users in consultative role	Users provide feedback on predefined solutions.	Users provide feedback on predefined solutions.
Users in participative role	Users provide design solutions but do not have decision-making power.	Users provide design solutions but do not have decision-making power.
Users represented by intermediaries	Users are represented by other users in the discussion forum.	HCI specialists represent users in development.
Intermediaries in informative role	Users provide very little mention of other users' characteristics or practices.	HCI specialists identify target users based on their (HCI, domain) expertise.
Intermediaries in consultative role	Users provide feedback from the viewpoint of other users.	HCI specialists provide feedback based on their HCI expertise and user contact.
Intermediaries in participative role	–	HCI specialists take part in the design process with developers and have decision-making power regarding the solution

by some users (cf. Zhao and Deek 2005, 2006, Barcellini *et al*. 2008, 2009, Titlestad *et al*. 2009), probably rather technically competent ones, but whether these solutions were accepted was totally dependent on the opinion of the developer (cf. Ye and Kishida 2003). No empirical inquiries into the user population were carried out. However, HCI specialists took active part in the design process, with decision-making power regarding the solution, and took particular care to redesign users' practices. Finally, evaluation was evident in both OSS development contexts. Both users and their intermediaries were involved in this activity.

5.2. *Characterising distributed Participatory design*

The people involved in OSS development tend to be distributed physically, organisationally and temporally. Work tends to be carried out remotely, using the Internet for communication and coordination. This was the case in the analysed OSS development project. In the company context, development is often distributed as well (e.g. Erickson and Evaristo 2006, Gumm 2006, Levina 2006), but this was not the case in the unit analysed here, in which the personnel happened to be co-located. Users were invited to the project location or contacted through means in the OSS environment, in which case the parties were again distributed physically, organisationally and temporally. Table 2 outlines the different forms of distributed PD identified in the OSS development context.

Table 2 illustrates that the traditional community OSS project analysed here included distributed PD. There was direct user participation as well as representation by intermediaries. However, the understanding activity was neglected, and the designing activity realised in a limited sense. Distributed PD in this case took place mainly through users and people 'representing' them by providing feedback in the discussion forum. Also in the second case, distributed PD was related mainly to evaluation. OSS communities were utilised in gathering user feedback; the distributed users were invited to propose design solutions as well. This bears a clear resemblance to the gathering of user feedback and improvement of ideas in the OSS project analysed here. The HCI specialists reported attempting to 'classify' and 'rank' the data, which can be assumed to be necessary in OSS projects without company involvement as well. Finally, they acted as intermediaries between distributed users and developers by delivering user feedback to development personnel; this was also carried out by utilising the OSS communities and their existing means (mailing lists and discussion forums) for communication.

5.3. *Implications for OSS and distributed PD research and practice*

The results of this study indicate that online forums have been used successfully for distributed PD in the OSS development context (see also Scacchi 2002, Frank and von Hippel 2003, Lakhani and von Hippel 2003, Ye and Kishida 2003, Barcellini *et al*. 2008). Extensive use of online discussion forums was evident in both cases; non-developer users also took part in online discussions concerning understanding, designing and evaluating activities. Though little support for design work or usability bug reporting and analysis is offered in online discussion forums (cf. Nichols and Twidale 2003, 2006, Zhao and Deek 2005, Bach *et al*. 2009), they evidently can be used for these activities too. In the traditional OSS project, none of the more advanced means, such as (remote) usability testing, co-design by blogs, or professional HCI resources were evident (see Frishberg *et al*. 2002, Nichols *et al*. 2003, Nichols and Twidale 2003, 2006, Benson *et al*. 2004, Zhao and Deek 2005, Andreasen *et al*. 2006, Cetin *et al*. 2007, Iivari *et al*. 2008, Bach and Carroll 2009, Bach *et al*. 2009). The HCI specialists in the company

Table 2. Distributed participatory design in OSS development.

Distributed PD	Community OSS development	Company OSS development
Understanding the users	Few mentions of users' characteristics or practices provided by distributed users and intermediaries.	–
Designing new practices and technologies	Distributed 'technically capable users' take active part in technology design, but without decision-making power.	Distributed 'technically capable users' take active part in technology design, but without decision-making power.
Evaluating the design solutions	Distributed users and intermediaries in consultative role provide empirical user feedback.	Distributed users in consultative role provide empirical user feedback; HCI specialists in consultative role provide empirical user feedback gathered from distributed users.

case did carry out use practice redesign and usability evaluations, but did so in their co-located setting, which resembles traditional PD in co-located settings. Regarding distributed PD, in both cases the OSS communities were used to gather informal ideas and usability and user feedback.

In addition to identifying online forum-based distributed PD, this study indicates the importance of intermediary-driven distributed PD. In both cases there were intermediaries 'representing the users'; in the traditional OSS development case, they were users speaking on behalf of other users, while in the company case they were professional HCI specialists hired by the company. Existing research has already indicated that different kinds of intermediaries mediating between developers and users are needed (see Nichols and Twidale 2003, Benson *et al*. 2004, Zhao and Deek 2005, Andreasen *et al*. 2006, Gumm *et al*. 2006, Cetin *et al*. 2007, Barcellini *et al*. 2008, 2009, Bach and Carroll 2009, Bach *et al*. 2009, Obendorf *et al*. 2009, Titlestad *et al*. 2009). In some cases, they may also deliver knowledge from design to inform use (e.g. Gumm *et al*. 2006, Barcellini *et al*. 2008, 2009, Obendorf *et al*. 2009, Titlestad *et al*. 2009), but here they focused specifically on delivering knowledge from use to inform design. They acted as spokespeople for users; some were amateur, informal intermediaries, while others had training and formal job roles related to the matter (cf. Iivari *et al*. 2009). In the company OSS case, the HCI specialists carried out understanding, designing and evaluating activities. In the distributed OSS development, they contributed mainly through evaluation activities. In the traditional OSS project, they contributed slightly to understanding and mostly to evaluating activities. The main findings of this study are given below in a few statements, which are to be further empirically examined by other researchers.

5.3.1. Online forum-based distributed PD is central in OSS development, and may prove to be valuable in other kinds of distributed development settings as well

In both cases, online discussion forums were used to gather data and design ideas from users. Forums were especially useful in gathering user feedback; this type of user data gathering could be experimented with in other distributed contexts, e.g. in web IS and product development contexts where, as in OSS development, the user population might be very large and distributed around the world, making it difficult to contact individuals face to face (Iivari 2006). Therefore, it might be useful to encourage users to spontaneously and voluntarily provide feedback and other data relevant to the development. This would necessitate setting up a forum and an active community around it, which is not an easy task. In addition, depending on the intended user population, not all people are able or willing to use such Internet communication tools (Benson *et al*. 2004, Nichols and Twidale 2003, 2006, Cetin *et al*. 2007). Because of these problems, future research is needed to examine the factors helping and hindering the use of discussion forums as PD tools in different kinds of distributed settings.

5.3.2. Distributed PD needs support particularly in 'understanding' and 'designing' activities

Clearly, evaluation with the distributed users is carried out in the OSS development context. Though additional support could be provided – a need for better (remote) usability reporting mechanisms has been identified (Nichols *et al*. 2003, Nichols and Twidale 2003, 2006, Cetin *et al*. 2007) – the other activities seem to have more acute needs. Designing with users is an integral element of the PD tradition, and numerous methods for it have been devised (see e.g. Greenbaum and Kyng 1991, Schuler and Namioka 1993); however, as mentioned, they tend to assume co-located participants, because of which they do not fit the distributed OSS development context. Design by blogs has been recommended as a quick and public way for non-co-located design work (Nichols and Twidale 2006), but it enables not user participation but distributed design work. Of course, technically skilled users can utilise it to communicate their solutions to the developers, along with other methods presented in this article (e.g. textual descriptions, mock-ups, patches). An open question, however, is how non-developer users can be invited to participate as co-designers to design their future practices and technologies. There may be a need for intermediaries orchestrating these sessions and making non-technical users comfortable participating. The usual PD tools and practices (prototypes, mock-ups and workshops) have proved useful in co-located settings, but their modification to fit the distributed PD setting is a topic for future research.

The same applies to the issue of gaining a detailed understanding of users' current work practices before redesigning them. One should not assume that OSS developers will take care of user studies themselves: either the users should be supported in articulating their expertise or intermediaries representing them should be acquired to do so. HCI specialists or ethnographers typically provide these kinds of data for design, but tend to be unavailable in the OSS development context. In the company OSS development context, however, they might be capable of contributing by gathering empirical data as well as ranking and classifying it. Where these professional

intermediaries are not available, amateur intermediaries could be invited to help. The existing tools for communicating the results of the understanding activity might be suitable for the distributed development context as well: i.e. different kinds of reports, scenarios, storyboards and personas can be delivered to interested parties almost as easily in distributed development as in a co-located setting.

5.3.3. *Amateur 'user representation work' carried out by users and intermediaries 'representing other users' provides useful data for OSS development*

An interesting innovation in the traditional OSS project was the user representation work carried out by users and the intermediaries 'representing' them. A similar kind of behaviour has been reported by Barcellini *et al*. (2008, 2009), though in that case the intermediaries seem to be mostly developer-users. Even though this study reveals that mainly informal feedback on predefined design solutions was gained, this type of voluntary user data delivery could be supported and made more systematic by providing advice on the kind of information the project needs on its users, their goals and contexts of use, and the OSS version already in use. One should inform the users and the intermediaries 'representing' them what kind of data, in what kind format, are valid and useful for design.

5.3.4. *Professional intermediaries are needed to ensure that the users' voices are really heard*

The HCI specialists and users 'representing other users' tried to give voice to the other, less technologically literate users. However, some of the representations produced by the users 'representing other users' seemed to be based only on opinions or stereotypes, due to which one should remain somewhat critical towards them. It is important to recall that the masses (the non-developer users) are never equally equipped to participate in the discourse on technological development, and that the technological elite always has the authority to produce the representations of the technologically illiterate, 'primitive,' 'exotic' other (cf. Asaro 2000, Iivari 2009), for which reason some sort of professional intermediaries are needed to ensure that the voices of the 'technologically illiterate masses' are heard. Professional HCI specialists have an understanding of what is needed from them, which might be difficult for OSS developers to articulate. Their expertise could initially be used for ranking and classifying the contributions already available in the OSS discussion forums; in addition, their education offers them the skills to contribute to understanding, designing and evaluating activities. However, as mentioned, it has been argued that HCI specialists tend to be isolated and that their work tends to have little effect on solutions in the OSS development context (Benson *et al*. 2004, Nichols and Twidale 2006, Bødker *et al*. 2007, Cetin *et al*. 2007, Viorres *et al*. 2007). The HCI specialists in the company case mentioned that OSS developers might be reluctant to make changes based on their feedback. Therefore, future research related to the complexities and challenges involved with HCI specialists gaining a participative position and decision-making power in OSS projects is needed.

6. Conclusions

This article examined distributed PD in the OSS development context, analysing empirical data from a 'traditional' OSS development project and from a company OSS development setting, the particular case being a software development unit of a large global corporation. PD was interpreted to include both gaining an understanding of users' current practices and redesigning those practices with users; furthermore, feedback of the solutions made is to be gathered from users. Different kinds of roles are available to users, as well as to intermediaries possibly taking part.

Through empirical examination, different forms of PD and distributed PD were identified in the OSS development context. All the roles identified for the users were evident in the empirical data, but the consultative role clearly dominated. The importance of intermediaries representing the user in development was emphasised: either HCI specialists or other users were 'representing users' in development. The HCI specialists were also taking active part in the design process, with decision-making power regarding the solution. However, distributed PD mainly emerged in relation to evaluating solutions, leaving the understanding and designing activities neglected. Practical questions related to ways of supporting distributed PD – especially understanding and designing activities – were discussed. Interestingly, highly similar findings concerning distributed PD were derived from these divergent OSS cases. Based on their similarities, the importance of online forum-based and intermediary-driven PD was emphasised: online forums enable the understanding, designing and evaluation activities in distributed development; intermediaries ensuring these activities are put into practice, and all parties' voices are heard and have influence. Other PD and OSS researchers, as well as practitioners, are encouraged to further examine how to enable this online forum-based and intermediary-driven PD in different kinds of distributed development environments.

This study is based on analysis of only two cases. In the future, more variety should be included in the cases analysed. Though the analysis includes cases representing a traditional community model and a company OSS development model, a number of distinctions related to OSS projects were not taken into account in this article: project size, length, application type, etc. These two case studies also rely on highly divergent data sets: the first examines a discussion forum established for end users to take part, while the second one relies on multiple empirical materials gathered mainly from the company's perspective. In both cases, interesting additional insights could be gained by interviewing OSS developers; however, both cases already provide plenty of findings from the viewpoint of end-user participation. The applications developed in both cases were aimed at a general public, not only at technically capable people, and the cases showed clear interest in users and their involvement. Therefore, the cases studied here provide fruitful settings in which to examine distributed PD in the OSS development context. The results can be better generalised by including two cases in the analysis. Generalisations in interpretive research are 'explanations of particular phenomena derived from empirical interpretive research in specific IS settings, which may be valuable in the future in other organizations and contexts' (Walsham 1995, p. 79). The identified PD practices may be valuable in other distributed IS settings as well and it is assumed that they are likely to be found in other cases representing the OSS development context.

A number of interesting paths for future work can be identified, including further empirical, interpretive analyses related to users' roles in different kinds of OSS development projects and in other distributed development contexts. Future research on appropriate ways of supporting the 'understanding' and 'designing' activities in distributed PD is particularly recommended; another path for future work is to consider appropriate ways to support and orchestrate participation of non-developer users, other users 'representing' them, and the HCI specialists 'representing' them in the OSS development context. Constructive research on suitable PD methods and tools in different kinds of distributed development settings is needed as well.

Acknowledgements

This research has been partly funded by the Academy of Finland.

Notes

1. Among other motivations, e.g. emancipation from large software companies, gaining reputation and career opportunities, learning, fun and altruism (see e.g. Bonaccorsi and Rossi 2006).
2. This study focuses on participatory design rather than on vaguely and diversely defined user-centred design (see e.g. Kujala 2003, Iivari and Iivari 2006), in which user participation is also an integral element, but which is defined from the viewpoint of the designer. Participatory design is based on the premise of cooperation between designers and users. This is evidently enabled and expected in OSS development, though many complexities are involved, because the distinction between designer and user is blurred and many users are not willing or able to adopt the co-designer role. Adopting the concept of user-centred design would require focusing on designers' actions (how they work with the users), while participatory design enables allowing agency for both parties.
3. There are a variety of intermediaries discussed in these studies: HCI specialists, local implementers, active users speaking on behalf of other users, communities of interest formed by representatives of divergent user groups, and developer-users taking part in both use and development communities mediating between them.

References

Andreasen, M., *et al*., 2006. Usability in open source software development: opinions and practice. *Information Technology and Control*, 35 (3A), 303–312.

Asaro, P., 2000. Transforming society by transforming technology: the science and politics of participatory design. *Accounting, Management and Information Technologies*, 10, 257–290.

Bach, P. and Carroll, J., 2009. FLOSS UX design: an analysis of user experience design in Firefox and OpenOffice.org. *In*: C. Boldyreff, K. Crowston, B. Lundell, and A. Wasserman, eds. *Open source ecosystems: diverse communities interacting, 5th international conference on open source systems*, 3–6 June 2009, Skövde, Sweden. Boston: Springer, 237–250.

Bach, P., DeLine, R., and Carroll, J., 2009. Designers wanted: participation and the user experience in open source software development. *In*: *Proceedings of CHI2009 conference on human factors in computing systems*. New York: ACM Press, 985–994.

Barcellini, F., Detienne, F., and Burkhardt, J., 2008. User and developer mediation in an open source software community: boundary spanning through cross participation in online discussions. *International Journal of Human-Computer Studies*, 66, 558–570.

Barcellini, F., Detienne, F., and Burkhardt, J., 2009. Participation in online interaction spaces: design-use mediation in an open source software community. *International Journal of Industrial Ergonomics*, 39, 533–540.

Benson, C., Müller-Prove, M., and Mzourek, J., 2004. Professional usability in open source projects: GNOME, OpenOffice.org, NetBeans. *In*: *Extended abstracts CHI2004 conference on human factors in computing systems*. New York: ACM Press, 1083–1084.

Binder, T., Gregory, J., and Wagner, I., eds., 2002. *In*: *Proceedings of participatory design conference*. CPSR: Palo Alto.

Bjerknes, G. and Bratteteig, T., 1995. User participation and democracy. A discussion of Scandinavian Research on System Development. *Scandinavian Journal of Information Systems*, 7 (1), 73–98.

Bonaccorsi, A. and Rossi, C., 2006. Comparing motivations of individual programmers and firms to take part in the open source movement: from community to business. *Knowledge, Technology, & Policy*, 18 (4), 40–64.

Borgholm, T. and Madsen, K., 1999. Cooperative usability practices. *Communications of the ACM*, 42 (5), 91–97.

Bødker, S. and Buur, J., 2002. The design collaboratorium – a place for usability design. *ACM Transactions on Computer-Human Interaction*, 9 (2), 152–169.

Bødker, M., Nielsen, L., and Orngreen, R., 2007. Enabling user-centered design processes in open source communities. *In*: N. Aykin, ed. *Proceedings of human computer interaction international: usability and internationalization*. *LNCS 4559*. Berlin: Springer, 10–18.

Cetin, G., Verzulli, D., and Frings, S., 2007. An analysis of involvement of HCI experts in distributed software development: practical issues. *In*: D. Schuler, ed. *Proceedings of human computer interaction international: online communities and social computing*. *LNCS 4564*. Berlin: Springer, 32–40.

Clement, A., 1994. Computing at work: empowering action by 'low-level users'. *Communications of the ACM*, 37 (1), 52–63.

Clement, A. and van den Besselaar, P., eds., 2004. *Proceedings of participatory design conference*. CPSR: Palo Alto.

Damodaran, L., 1996. User involvement in the systems design process – a practical guide for users. *Behaviour and Information Technology*, 15 (16), 363–377.

Denzin, N. and Lincoln, Y., 2000. Introduction: the discipline and practice of qualitative research. *In*: N. Denzin and Y. Lincoln, eds. *Handbook of qualitative research*. 2nd ed. Thousand Oaks, CA: Sage, 1–29.

Erickson, J. and Evaristo, R., 2006. Risk factors in distributed projects. *In*: *Proceedings of 39th annual Hawaii international conference on system sciences*. Washington, DC: IEEE, 10p.

Fitzgerald, B., 2006. The transformation of open source software. *MIS Quarterly*, 30 (3), 587–598.

Franke, N. and von Hippel, E., 2003. Satisfying heterogeneous user needs via innovation toolkits: the case of Apache security software. *Research Policy*, 32, 1199–1215.

Frishberg, N., *et al*., 2002. Getting to know you: open source development meets usability. *In*: *Extended abstracts CHI2002 conference on human factors in computing systems*New York: ACM Press, 932–933.

Greenbaum, J. and Kyng, M., eds., 1991. *Design at work*. *Cooperative design of computer systems*. New Jersey: Lawrence Erlbaum Associates.

Gumm, D., 2006. Distributed software development – a taxonomy. *IEEE Software*, 23 (5), 45–51.

Gumm, D., Janneck, M., and Finck, M., 2006. Distributed participatory design – a case study. *In*: *Proceedings of NordiCHI workshop on distributed participatory design*, 14 October 2006, Oslo, Norway, 5p.

Heikinheimo, H. and Kuusisto, T., 2004. The use of embedded open source software in commercial products. *In*: *Proceedings of the 13th European conference on information systems*, 14–16 June 2004, Turku, Finland, 8p.

Hornbaek, K. and Stage, J., 2006. The interplay between usability evaluation and user interface design. *International Journal of Human-Computer Interaction*, 21 (2), 117–123.

Iivari, N., 2006. Understanding the work of an HCI practitioner. *In*: A. Morch, K. Morgan, T. Bratteig, G. Ghosh, and D. Svanaes, eds. *Proceedings of 4th Nordic conference on human computer interaction*, 14–16 October 2006, Oslo, Norway. New York: ACM, 185–194.

Iivari, N., 2009. "Constructing the users" in open source software development – an interpretive case study of user participation. *Information Technology & People*, 22 (2), 132–156.

Iivari, N., Hedberg, H., and Kirves, T., 2008. Usability in company open source software context. Initial findings from an empirical case study. *In*: *Proceedings of 4th international conference on open source systems*, 7–10 September 2008, Milan, Italy, 359–365.

Iivari, J. and Iivari, N., 2006. Varieties of user-centeredness. *In*: *Proceedings of the 39th Annual Hawaii international conference on system sciences*. Washington, DC: IEEE Computer Society Press, 10.

Iivari, N., *et al*., 2009. Mediation between design and use – revisiting five empirical studies. *Human IT – Journal for Information Technology Studies as a Human Science*, 10 (2), 81–126.

Karasti, H., 2001. *Increasing sensitivity towards everyday work practice in system design*. Acta Universitatis Ouluensis, Scientiae Rerum Naturalium, A 362. Oulu: Oulu University Press.

Kensing, F. and Blomberg, J., 1998. Participatory design: issues and concerns. *Computer Supported Cooperative Work*, 7 (3–4), 167–185.

Klein, H. and Myers, M., 1999. A set of principles for conducting and evaluating interpretive field studies in information systems. *MIS Quarterly*, 23 (1), 67–94.

Kujala, S., 2003. User involvement: a review of the benefits and challenges. *Behaviour and Information Technology*, 22 (1), 1–16.

Lakhani, K. and von Hippel, E., 2003. How open source software works: "free" user-to-user assistance. *Research Policy*, 32, 923–943.

Levina, N., 2006. Collaborating on multiparty information systems development projects: a collective reflection-in-action view. *Information Systems Research*, 16 (2), 109–130.

Ljungberg, J., 2000. Open source movements as a model for organizing. *European Journal of Information Systems*, 9 (4), 208–216.

Luke, R., *et al*., 2004. The promise and perils of a participatory approach to developing an open source community learning network. *In*: *Proceedings of participatory design conference*. CPSR: Palo Alto, 11–19.

Markus, M. and Mao, Y., 2004. User participation in development and implementation: updating an old tired concept for today's contexts. *Journal of the Association for Information Systems*, 5 (11–12), 514–544.

Mockus, A., Fielding, R., and Herbsleb, J., 2002. Two case studies of open source software development: Apache and Mozilla. *ACM Transactions on Software Engineering and Methodology*, 11 (3), 309–346.

Nichols, D., McKay, D., and Twidale, M., 2003. Participatory usability: supporting proactive users. *In*: *Proceedings of the 4th annual conference of the ACM special interest group on computer human interaction – New Zealand Chapter*, 3–4 July 2003, Dunedin, New Zealand. New York: ACM, 63–68.

Nichols, D. and Twidale, M., 2003. The usability of open source software. *First Monday*, 8 (1), 21.

Nichols, D. and Twidale, M., 2006. Usability processes in open source projects. *Software Process Improvement and Practice*, 11, 149–162.

Niederman, F., *et al*., 2006. A research agenda for studying open source I: a multilevel framework. *Communication of the Association for Information Systems*, 18, 19–149.

Obendorf, H., Janneck, M., and Finck, M., 2009. Inter-contextual distributed participatory design. *Scandinavian Journal of Information Systems*, 21 (1), 51–76.

Scacchi, W., 2002. Understanding the requirements for developing open source software systems. *IEE Proceedings – Software*, 149 (1), 24–39.

Schuler, D. and Namioka, A., eds., 1993. *Participatory design: principles and practices*. New Jersey: Lawrence Erlbaum Associates.

Titlestad, O., Staring, K., and Braa, J., 2009. Distributed development to enable user participation. *Scandinavian Journal of Information Systems*, 21 (1), 27–50.

Tuovila, S. and Iivari, N., 2007. Bridge builders in IT artifact development. *In*: H. Österle, J. Schelp, and R. Winter, eds. *Proceedings of 15th European Conference on Information Systems*, 7–9 June 2007, St Gallen, Switzerland, 819–830.

Walsham, G., 1995. Interpretive case studies in IS research: nature and method. *European Journal of Information Systems*, 4, 74–81.

Viorres, N., *et al*., 2007. Major HCI challenges for open source software adoption and development. *In*: D. Schuler, ed. *Proceedings of human computer interaction international: online communities and social computing*. *LNCS 4564*. Berlin: Springer, 455–464.

Ye, Y. and Kishida, K., 2003. Toward an understanding of the motivation of open source software developers. *In*: *Proceedings 25th international conference on software engineering*. Washington: IEEE, 419–429.

Zhao, L. and Deek, F., 2005. Improving open source software usability. *In*: N. Romano, ed. *Proceedings of 11th Americas conference on information systems – a conference on a human scale*, 11–14 August, 2005, Omaha, NE, 923–928.

Zhao, L. and Deek, F., 2006. Exploratory inspection: a learning model for improving open source software usability. *In*: *Extended abstracts CHI2006 conference on human factors in computing systems*. New York: ACM Press, 1589–1594.

Copyright of Behaviour & Information Technology is the property of Taylor & Francis Ltd and its content may not be copied or emailed to multiple sites or posted to a listserv without the copyright holder's express written permission. However, users may print, download, or email articles for individual use.