Usability Evaluation in Task Orientated Collaborative Environments

Florian Urmetzer and Vassil Alexandrov

ACET Centre, The University of Reading, Reading, RG6 6AY
{f.urmutzer, v.n.alexandrov}@reading.ac.uk

Abstract. An evaluation of the usability is often neglected in the software development cycle, even that it was shown in the past that a careful look at the usability of a software product has impact on its adoption. With the recent arising of software supporting collaboration between multiple users, it is obvious that usability testing will get more complex because of the multiplication of user interfaces and their physical distribution. The need for new usability testing methodologies and tools for their support under these circumstances is one consequence. This paper widens the methodologies of usability evaluation to computing systems supporting the solving of a collaborative work task. Additionally details of a distributed screen recording tool are described that can be used to support usability evaluation in a collaborative context.

Keywords: collaboration, groupware, evaluation, usability, testing, HCI.

1 Introduction

Collaborations are well known to better the outcome of projects in industry as in academia. The theory is based on the availability of specialist knowledge and workforce through different collaborators [1]. Software tools to support any form of collaboration are widely used in scientific as well as in business applications. These include tools to enhance communications between collaborators, like for example Access Grid [2] or portals enabling text based exchange of information [3]. Additionally there are tools to support virtual organizations in their work tasks. These tools have the aim to enable two or more individuals to work in distributed locations towards one work task using one computing infrastructure. An example is the Collaborative P-Grade portal [4] or distributed group drawing tools as described in [5]. The P-Grade portal enables distributed users to make contributions to one workflow at the same moment in time. Therefore users can actively change, build and enhance the same workflow in an informed fashion using advantages of collaborative working models. Similarly the distributed group drawing tool enables multiple users to draw and share the drawing via networks.

In this paper, first the state of the art in usability evaluation methods will be looked at, detailing the current methods for usability testing. Then an example of a collaborative system will be given to provide an understanding of the terminology

collaborative software. The authors will argue that current methods have to be revised under the new collaborative software paradigm. Finally the authors will introduce a multiple screen recorder, which is currently under development. This recorder may help to enable usability evaluation in collaborative environments and therefore enable a revision of the current methodologies, namely observational studies, under the new collaborative software paradigm.

2 Methods for Usability Evaluation and Testing

Usability testing has been defined as the systematic way to observe real users handling a product and gathering data about the interaction with the product. The tests can determine if the product is easy or difficult to use for the participants [6]. Therefore the aim of enquiries into usability is to uncover problems concerning the usage of existing interfaces by users in realistic settings. The outcomes of these tests are suggestions to better the usability and therefore recommendations to change the interface design [7].

Historically usability tests have been at the end of software projects to enable the measurement of usability of the outcome of the project and to better the interface. One example is automatic user interface testing, where software evaluates with the help of an algorithm a programmed specification on a user interface. These automatic methods have however been described to have scaling problems when evaluating complex and highly interactive interfaces [7].

In recent years usability testing has changed to the user centred paradigm and is therefore implemented more and more into the software design cycle. The testing involving target users however is still an important part of the enquiry [8], [9], [10]. Two examples are the use of design mock-ups and the use of walkthroughs.

For example Holzinger [11] used drawn paper mock-ups of interfaces to elicit information from potential users of a virtual campus interface. Paper mock-ups were described to be used by the researchers to allow the user to interact in single and group interviews before the interface was programmed. The researchers described the process and methods applied in the project as useful for identifying the exact look and feel of an interface and the requirements of the users' usability needs.

A second example for a user centric design approach is the usability walkthrough. Using focus groups, the walkthrough ensures a quality interface and the efficiency of the software production. This process is indicated to include users as well as product developers and usability experts in a group setting during the design of the software. This means that all attendees discuss the functionality and the design of the software before the programming has started [12], [13].

The two example methods in use to define the design of interfaces are however stressed to be additions to tests with users using the interfaces after the programming was done [7]. In usability evaluations, after the software is programmed, most commonly observational methods are chosen to determine the use of interfaces. For example the researcher would observe chosen participants performing tasks while using the tested interface. The tasks should be related to the everyday jobs that need to be performed by the users when using the software, for example the question, 'can

a user send e-mails using newly developed e-mail software?’ However as Dumas [6] points out the tasks to be performed should be a set of problem tasks, instead of tasks that are seen to be rather easy. This is seen as a way of making sure that the interface is not verified, but critically analysed.

While the user is performing the number of tasks provided, the tester would typically ask the participant to speak aloud what he is thinking during the execution of the task. This gives the tester an inside to the participants thoughts and problems, for example a participant would say ‘how do I send this e-mail now’ indicating that he is looking for the send button. This method is named think aloud [14], [16]. An extension of the think aloud is to make two people work on one interface and make them communicate about their actions. This is seen as more natural then just to talk to oneself [6].

A technology supported extension of usability testing is remote usability testing. This is described to enable testing participants in their natural surroundings and with the opportunity not to have to travel to the participants. The researchers in [16] have described their study to involve a video conferencing system, shared whiteboard and shared applications software on the participants’ side. This video conferencing software will then enable the observer on his side to see the image of the participant through the video conferencing system and to see the computer screen using the shared applications software. The shared whiteboard would be used to give instructions and tasks to the test participants.

In a more recent publication the researchers have used remote usability testing using Microsoft NetMeeting to share the test participants screen and a software called ‘snag-it’ to capture the shared content from the observers computer screen. In this particular case speaker phones on the participants’ side and a tape recorder on the observers side was used to capture think aloud comments from the participant. The tasks to be performed during the test were given to the user in written form via postal mail. The major outcome of the study was that they found that remote usability testing is as accurate as face to face testing [17].

Fig. 1. A remote usability test, using observational methods, the researcher is observing the participant using the software via the internet and a phone network. The phone is used for think aloud protocols and the participants screen is shared using screen sharing software.

2.1 The Collaborative Challenge

With the CoWebs (collaborative web spaces, e.g. wiki) it is possible to move more away from a text only media and to integrate collaborative multimedia allowing the group editing of, for example, formulas [18]. Supporting the CoWeb theory is the advent of virtual organisations in the Grid paradigm. Individuals can form virtual collaborations and therefore share information and resources in electronic means across organisations [19].

An example for a tool produced under these paradigms is the Collaborative P-Grade Portal [4]. This portal enables multiple users to make contributions to one workflow at the same moment in time. This is possible through a central server, where every participant in the collaboration logs in and receives an editable representation of the workflow. The workflow jobs are lockable by only one user in the collaboration. Through the locking, the job becomes editable for the user and is shown as locked to the other users (see fig. 3). This mechanism allows users to actively change, build and enhance the same workflow while being in different geographic locations.

Fig. 2. Example of the P-Grade Portals central server sharing a workflow. The locks can be seen in the interfaces indicating that the nodes are not editable.

This shift from single user applications to collaborative applications needs a reflection in the methodology of usability testing. At present it is not possible to run any form of usability tests in these collaborative settings. However it would be highly interesting to report such findings and therefore to gather more detailed information on the use and of the requirements of such systems.

3 Enabling Usability Evaluation in Collaborative Environments

The discussion above has shown that there are developed methods to enable usability evaluation when single interfaces are used. However there is need for tools to investigate multiple interfaces. This will have an important impact on the ability to conduct user interface evaluation and therefore to uncover problems concerning the usage of existing interfaces in realistic settings when using collaborative software. Therefore the researchers of this paper have proposed a software allowing distributed usability evaluation using observational methods.

The software allows the recording of chosen screen content and cameras, including the sound, of clients involved in a collaborative task. Therefore the clients screen content is captured and transferred over the network to the recording server where the video is stored to disc.

Fig. 3. Test computers are capturing screens and cameras and transfer the images and audio to a recording server. From there the usability tester can playback the recorded material.

The capturing of screens has been realized using the Java media framework. The area of the screen to be captured can be defined using x and y locations of the screen. Therefore not all the screen has to be captured, but only the part of the screen showing the application looked at by the researchers is captured.

Before the video is transferred, a timestamp is added, so that it is possible to synchronize the replay of the different clients at a later stage.

The transport is organized using RTP (Real Time Protocol) from the client to the server. Finally the video is written to disc on the server.

A researcher interested in analyzing the material gathered can then play the material back from the server to a single client. This will then transfer the video synchronized to one client.

4 Future Directions and Conclusion

The future tasks in the project will be grouped into three areas.

First more functionality will be added to the user interfaces of the clients as well as of the researchers interface. On the client side a choice of whether a camera and/or

audio is shared has to be added in the graphical user interface. These selections then have to be fed back to the user in a smaller window on the screen to enable the user to gain more control over the shared material. Additionally an option has to be created, where the user can choose the size of the transferred video. This will prevent the overload of networks and of the personal computer of the participant. The researcher has to have the option to choose the video feeds during playback and to see the video feeds during recording. This should enable the researcher to interfere in the test when needed.

Second it should be possible to add a tag to the recorded material. Tagging is the adding of information to the recorded material, which can be in text form as well as in graphical form. This should enable scientists to describe moments in time and code the material for further analysis. This tool is seen as important by the researchers, given that during the analysis it will be hard to keep track of where usability problems occur, because the attention of the researcher has to be multiplied by the number of participants. Similar, tagging mechanisms have been used in meeting replays [20].

Finally a messenger structure to allow textual communication between the tester and the participant has to be added. This may only be text based, however could be voice based or even video based. However intensive testing of network overload and other issues has to be done before.

In conclusion the researchers have shown in a prove of concept [21] that it is technically possible to enable the recording of multiple screens and cameras on one server. This prove of concept is now programmed in one server and multiple clients application. This application will enable researchers to conduct observational studies in distributed collaborative computing environments and therefore to evaluate user interfaces.

The current methods used for such interface evaluations have been described in this paper in great detail and the new development is described from the classical observational study of one user to the observation of groups of users and their screens. At this point it is taken as proved that the observation of the users via networked screen and camera recording is possible, however tests are being conducted at present to see how effective the new methodologies are from a human factor point of view.

References

1. Dyer, J.H., Collaborative advantage: winning through extended enterprise supplier networks. 2000, Oxford: Oxford University Press. xii, 209.
2. AccessGrid. Access Grid: a virtual community. [Internet] 2003 [cited 2006 13 March 2003]; <http://www-fp.mcs.anl.gov/fl/accessgrid/>.
3. Klobas, J.E. and A. Beesley, Wikis: tools for information work and collaboration. Chandos information professional series. 2006, Oxford: Chandos. xxi, 229 p.
4. Lewis, G.J., et al., The Collaborative P-GRADE Grid Portal. 2005.
5. Saul, G., et al., Human and technical factors of distributed group drawing tools. *Interact. Comput.*, 1992. 4(3): p. 364-392.
6. Dumas, J.S. and J. Redish, A practical guide to usability testing. Rev. ed. 1999, Exeter: Intellect Books. xxii, 404.
7. Mack, R.L. and J. Nielsen, Usability inspection methods. 1994, New York; Chichester: Wiley. xxiv, 413.

8. Carroll, C., et al., Involving users in the design and usability evaluation of a clinical decision support system. *Computer Methods and Programs in Biomedicine*, 2002. 69(2): p. 123.
9. Nielsen, J., *Usability engineering*. 1993, Boston; London: Academic Press. xiv, 358.
10. Faulkner, X., *Usability engineering*. 2000, Basingstoke: Macmillan. xii, 244.
11. Holzinger, A., Rapid prototyping for a virtual medical campus interface. *Software*, IEEE, 2004. 21(1): p. 92.
12. Bias, R., The Pluralistic Usability Walkthrough: Coordinated Emphaties, in *Usability Inspection Methods*, J. Nielsen and R.L. Mack, Editors. 1994, Wiley & Sons, Inc: New York.
13. Urmetzer, F., M. Baker, and V. Alexandrov. Research Methods for Eliciting e-Research User Requirements. in *Proceedings of the UK e-Science All Hands Meeting 2006*. 2006. Nottingham UK: National e-Science Centre.
14. Thompson, K.E., E.P. Rozanski, and A.R. Haake. Here, there, anywhere: remote usability testing that works. in *Proceedings of the 5th conference on Information technology education table of contents*. 2004. Salt Lake City, UT, USA: ACM Press, New York, NY, USA.
15. Monty, H., W. Paul, and N. Nandini, Remote usability testing. *interactions*, 1994. 1(3): p. 21-25.
16. Thompson, K.E., E.P. Rozanski, and A.R. Haake, Here, there, anywhere: remote usability testing that works, in *Proceedings of the 5th conference on Information technology education*. 2004, ACM Press: Salt Lake City, UT, USA.
17. Dieberger, A. and M. Guzdial, CoWeb - Experiences with Collaborative Web Spaces, in *From Usenet to CoWebs: interacting with social information spaces*, C. Lueg and D. Fisher, Editors. 2003, Springer: London. p. x, 262 p.
18. Foster, I., C. Kesselman, and S. Tuecke, The Anatomy of the Grid: Enabling Scalable Virtual Organizations. *International Journal on Supercomputing Applications*, 2001. 15(3).
19. Buckingham Shum, S., et al. Memetic: From Meeting Memory to Virtual Ethnography & Distributed Video Analysis. in *Proc. 2 nd International Conference on e-Social Science*. 2006. Manchester, UK: www.memetic-vre.net/publications/ICeSS2006_Memetic.pdf.
20. Urmetzer, F., et al. Testing Grid Software: The development of a distributed screen recorder to enable front end and usability testing. in *CHEP 06*. 2006. India, Mumbai.