Mohamedally, D., Edlich, S., Klaus, E., & Zaphiris, P. (2006). MIKEs PET A Participant-based Experiment Tracking Tool for HCI Practitioners using Mobile Devices? , In Proceedings of SPIE?s 18th Annual Symposium for Electronic Imaging 2006, San Jose, California, USA.

MIKE's PET: A Participant-based Experiment Tracking tool for HCI Practitioners using Mobile Devices

Dean Mohamedally[1], Stefan Edlich[2], Enrico Klaus[2], Panayiotis Zaphiris[1] ([1]Centre for HCI Design, City University London, Northampton Square, London EC1V 0HB, UK) ([2]TFH-Berlin, Luxemburger Str.10, 13353 Berlin)

ABSTRACT

Knowledge Elicitation (KE) methods are an integral part of Human Computer Interaction (HCI) practices. They are a key aspect to the synthesis of psychology empirical methods with requirements engineering, User Centred Design (UCD) and user evaluations. Examples of these methods include prototyping, focus groups, interviews, surveys and direct video observation. The MIKE project (*M*obile *I*nteractive *K*nowledge *E*licitation) at the Centre for HCI Design, City University London, UK provides mobile cyberscience capabilities for HCI practitioners conducting such research while at stakeholder locations. This paper reports on the design and development of a new MIKE based tool, named PET, a *P*articipant-based *E*xperiment *T*racking tool for HCI practitioners using Java-based (J2ME) mobile devices. PET integrates its user tracking techniques with the development of the second generation implementation of the CONKER (*CO*llaborative *N*on-linear *K*nowledge *E*licitation *R*epository) Web Service. We thus report further on CONKER v2.0's new capabilities developed to enable tighter collaboration and empirical data management between HCI practitioners, considering their UCD needs. The visualisation, tracking and recording of HCI participant-based datasets via PET is explored with close connectivity with the CONKER v2.0 Web Service, in order to provide mobile-web cyberscience for remote and local HCI practitioners.

Keywords: HCI, Mobile Interactive Knowledge Elicitation, Web-Portals, Mobile Cyberscience

1. INTRODUCTION

Our hypothesis introduces Knowledge Elicitation (KE) (Burge, 2001) in Human-Computer Interaction (HCI) to factors of mobility, asynchronicity, geographical dispersion and the synthesis of HCI formal practices with Cyberscience capabilities. The field of Cyberscience relates "to the use of Information and Communication Technologies (ICTs) for scientific purposes" (Nentwich, 2004). Nentwich asserts that ICTs will lead to qualitative changes in the content of research itself as well as the way research is conducted. With this in mind, this paper describes an environment for HCI practitioners to conduct Cyberscience-based HCI KE.

Recent advances in mobile computing technologies have demonstrated a potential for supporting HCI KE in the workplace. Therefore, an additional key aim is to provide the facilities for KE asset management for HCI practitioners that may be on site at stakeholder locations. For this aim, the scope of existing scientific rules in HCI knowledge elicitation practices have been simulated for adoption in HCI Cyberscience. This is achieved in the form of the second generation of the CONKER 2.0 Web Service (CONKER – Collaborative Non-linear Knowledge Elicitation Repository) (Mohamedally et al., 2005a) and PET – A Participant-based Experiment Tracking tool for HCI practitioners.

2. MIKE – MOBILE TOOLS FOR HCI KNOWLEDGE ELICITATION

MIKE (Mobile Interactive Knowledge Elicitation) is an open term we have given to define software and hardware configurations for a variety of mobile platforms technologies to specifically support and enhance the capabilities of Human-Computer Interaction (HCI) practitioners. Our research in this field is now in its third generation of HCI KE tools modelled for mobility under the various aspects provided by devices such as PocketPCs/Windows Mobiles, MPEG-4 based video handheld devices, Java (J2ME) enabled mobile phones, TabletPCs and PC Laptops. These MIKE software tools facilitate and simulate the criteria required by HCI practitioners for paper prototyping (Mohamedally et al., 2005b), remote user testing and observation (Mohamedally et al., 2005c), card sorting (Mohamedally et al., 2004) amongst other reported HCI KE techniques.

With CONKER supporting these tools and managing the collaboration and sharing of such electronic HCI KE data, our attention was drawn to methods and techniques in HCI KE whereby tracking the progress of HCI sessions and collecting supporting visual and audible evidence electronically would support the HCI practitioners' analysis of the test subject's data.

3. PET REQUIREMENTS

The design of PET has three key requirements of importance to the HCI practitioner whilst at stakeholder locations. These are (A) tracking changes and data management of the HCI KE sessions (B) to provide informal capture and recording mechanisms for the human-computer interaction that takes place for the purpose of supporting the HCI data collection in sessions, and (C) visualization and review of HCI KE datasets captured. The capabilities provided by cross-platform Java in the form of J2ME MIDP 2.0 enables a wide range of mobile devices to facilitate these requirements of PET in software. The CONKER v2.0 environment assists with providing a back-engine for storage, processing and collaborative sharing of these requirements.

3.1 Tracking Experiments with PET

Managing HCI sessions and tracking the participants' progress can be a time-consuming and paper-driven problem for HCI practitioners. In recent times, HCI practitioners have found ways to utilize Microsoft Excel and Word templates to fill in repetitive form-based information on the participants' profiles and manage the progress of their participatory design sessions. However retrieving and observing the history of particular users has not been facilitated in mobile HCI theory before. Thus PET was designed in mind with a database solution for tracking session information, chronological activity, and test subject (participant users) profiles. This data is collated and transferred into the CONKER 2.0 web service under managed projects.

3.2 KE Media Capture Facilities for PET

The low-fidelity nature of decision making in paper-based KE methods in HCI user design and evaluation is a significant aspect of HCI data collection. PET includes a series of media capture facilities for imaging and audio (see figure 1). These are done to support the practitioners in reviewing prior sessions, and provide valid comparative elements when analysing the data within sessions.

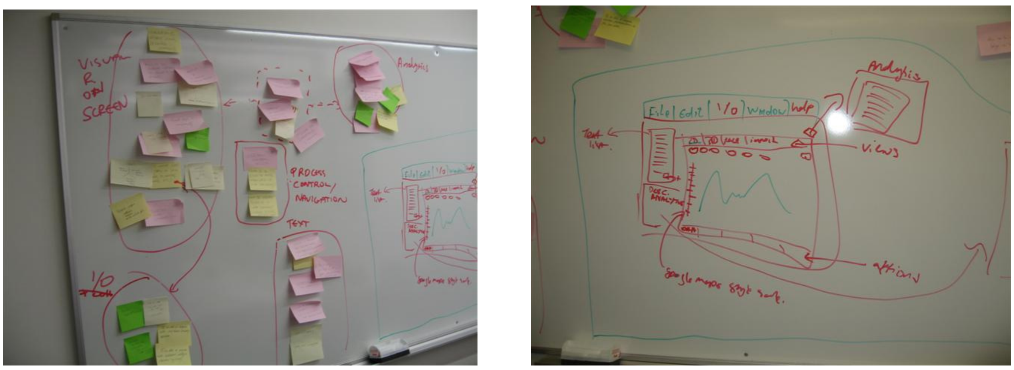

Figure 1: Many HCI techniques are low fidelity e.g. Affinity Diagramming and Paper Prototyping.
- PET provides smartphone data capture, tracking and session integration with CONKER 2.0

Image capture facilities (e.g. for paper prototyping, affinity diagramming, storyboarding and card sorting sessions) are provided with the MIDP 2.0 MMAPI classes. The image resolution captured is dependant on the free heap memory available on the host mobile device, thus the higher the resolution of the camera and memory available, the higher the quality of imaging for storage in CONKER 2.0.

Audio is equally important data capture for HCI practitioners. For example the participant's comments made during the prototyping, affinity diagramming and card sorting sessions, as well the responses made during interviews and focus groups. These forms of data capture with PET enable HCI practitioners to log and highlight important interviews, focus group sessions, and user testing observations.

3.3 Visualisation of HCI KE using PET

PET provides an interface to transmit and request data to and from the CONKER 2.0 Web Service. A HCI practitioner can log into a mobile CONKER 2.0 interface (XHTML based) via the PET application. From this they can select a HCI session to retrieve participant results from and review within PET e.g. paper prototype images collected from sessions and comments on sessions from CONKER.

The visualization aspect of PET is also important to the practitioner as it will tell them who else in their team is tracking and managing a HCI session. In large dispersed teams of practitioners, this can be a key asset to organizing comparative group studies and facilitating organized session changes in large HCI studies.

4. RAISING PET

PET 1.0 is designed for and tested with series 60 Nokia Mobile Phones (e.g. Nokia 6680, N70) and MIDP 2.0 compliant SonyEricsson Mobile Phones (K750, W800). These devices have a dynamic heap stack for their Java Virtual Machine implementation, and use memory cards which enable significant imaging data to be transferred to and from the CONKER 2.0 Web Service. PET was built using NetBeans 4.1 with the MIDP 2.0 API and the MMAPI extension classes for the multimedia capture requirements.

For devices with MMAPI support and camera imaging capabilities, A video capture `Player` object is instantiated by passing the URI locator `capture://video` to the `Manager.createPlayer()` method. A `VideoControl` object enables PET to display the current camera stream to the device screen, and utilizes a `VideoControl.getSnapshot()` method to take a JPG snapshot which can be stored for transmitting to CONKER. Similarly, audio capture in PET is achieved similarly using the `capture://audio` URI locator.

5. CONKER v2.0

The purpose of CONKER as outlined in prior work is to support HCI practitioners with managing their experimental user design and testing work, with the potential benefits that MIKE tools have in terms of mobility and collaboration. Using *Sourceforge.net* as an example, the sharing of data and templates is a key priority but also the management of HCI projects and the data within projects and teams of practitioners. There are a variety of techniques to achieve this, and second generation CONKER 2.0 users can create projects which can be edited by multiple users at the same time. Figure 2 outlines our updated web portal architecture.

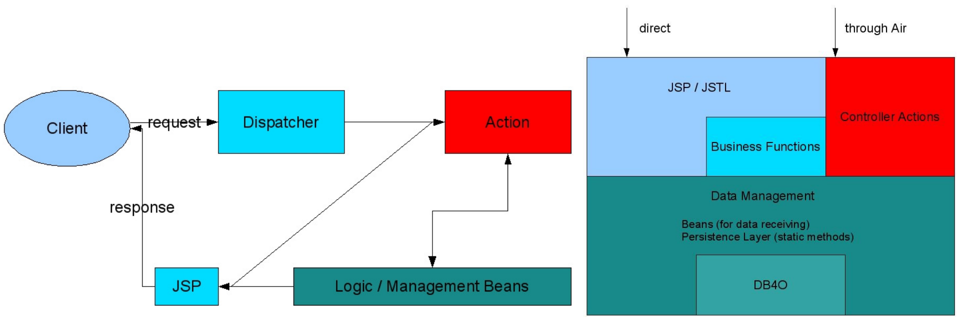

Figure 2: Portal architecture

Another requirement is framework that would support extensibility and scalability. In earlier work on CONKER (Mohamedally et al., 2005a) we reported on the framework Air *(www.air-framework.de)* which was designed with CONKER in mind.

5.1 Architecture Description

The core of the CONKER 2.0 system is again Air. Air utilises XSLT to separate the view from the scenario logic. The persistence layer is using db4o due to performance advantages and its' capabilities for deep object handling (e.g. projects having multiple sessions in experiments, single sessions per experiment, etc.).

We added an additional persistence layer to avoid side-effects when changing code later on, although we chose to utilise an object database here. Finally several distinct classes are needed due to the requirements of mobile devices and this followed through to the design of an architecture that would support graphics classes in research analysis.

To use this framework you include the jar and change the *web.xml* of the portal:

```
<servlet>
      <servlet-name>Dispatcher</servlet-name>
      <servlet-class>de.edlich.air.Dispatcher</servlet-class>
</servlet>
<servlet-mapping>
      <servlet-name>Dispatcher</servlet-name>
      <url-pattern>*.do</url-pattern>
</servlet-mapping>
```

Every request having the ending "*.do" is redirected to the dispatcher which redirects the response to the correct business classes. A parameter ACTIONCLASS describes this class. Business classes implement the Action-Interface and inherit from *ErrorUtils*. To achieve a more sophisticated error-handling, the portal now utilises a new abstract class *AbstractAction*:

```
public abstract class AbstractAction extends
            ErrorUtils implements Action {
public String target;
public void valiError(HttpServletRequest request, String error) {
      super.valiError(request, error);
      request.setAttribute(getClass().getSimpleName() +
                  "Error", error);
}
public void actionError(HttpServletRequest request, String error) {
      super.actionError(request, error);
      request.setAttribute(getClass().getSimpleName() +
                  "Error",    error);
      target = request.getQueryString();
}
public String getTarget(HttpServletRequest request) {
      if (target == null)
      target = request.getQueryString();
      return "index.jsp?" + target;

}
```

Every action within the portal will now be inherited from the class above. Hence every error message will get a name and if you have multiple forms on one page you now can assign the error messages to its origination. An additional parameter target allows handing over a target page for the successful processing of the action. Without this parameter the originating page will be taken. This also allows to set the page control flow and to control the error messaging.

We have been using JSLT which is using special JavaBeans based technology within the portal, to supply all the information and objects depending on the access rights. For special tasks and formatting requirements we added an additional custom tag library to CONKER 2.0.

Beside the actions as controller classes and the beans for the data transfer we are classing HCI scenario objects as formal business objects (BOs). Every entity of the system as (HCI) news, users, projects and experiments does exist as such a BO. They will be stored as they are and are entirely decoupled from the business logic.

The interface to the mobile clients is simply http. Special JSP pages in XML structure can be called from the mobile devices and being read with XML parser. Requests and data will be transmitted via HTTP requests (e.g. from forms) to the portal.

This partially sounds like the well known Struts or JSF usages but in fact it is much simpler and can be better adjusted to the needs of a HCI portal dealing with mobile devices.

5.2 CONKER 2.0 features

The portal already provides the following features:

- Project management and their experiments.
- Sharing of HCI data and experiment results.
- Tracking and monitoring during the experiments.
- Support the analyzing of the data with various mechanisms (like survey trend graphs, time-based polygraphs of participant activity, confidence dendograms from cluster analysis and more). Further research and investigations with analytical plug-ins will enhance the modelling capabilities of CONKER 2.0.
- Manage users' profiles and groups.
- News, blogs, and community forums.

These capabilities are tied to the everyday workflow model of HCI practitioners. As such, upcoming usability testing of the PET and CONKER 2.0 environments will validate the effectiveness of using such cyberscience techniques to aid and not hinder its real-world application to HCI practitioners, educators and researchers.

6. PET AND CONKER 2.0 TOGETHER IN ACTION

Using the CONKER 2.0 portal in collaboration with PET, you can acquire HCI datasets and validate them in a very short timeframe. Furthermore the capability to compare different HCI snapshots and gain some insights about trends enables a practitioner to take actions against negative evolutionary project paths. Mobile devices reduce the organisational overhead with test subjects at stakeholder locations. The interview results will be sent directly to the portal and can be used immediately.

A scenario-of-use example: A practitioner joins into an ongoing software development project for which they are responsible for pre-production designs. Initially we create this project in CONKER. Within this project we create sessions for experiments to be undertaken. PET will be used in participatory design sessions, and generate HCI KE data. PET will then require keys to upload them. The generation of keys acts as a security measure for each experiment, and will prohibit unauthorized data on private projects. This key also ensures that uploaded data will be assigned correctly to the right experiment and users. After enough keys have been generated, they can be easily distributed to each practitioner in the field. On the mobile client-side practitioners can then choose to utilise any of the MIKE tools, including PET (figure 3):

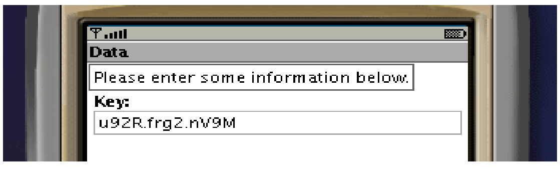

Figure 3: Providing keys for data assignment in the portal

After the key has been entered, related datasets are automatically linked to this key. Furthering this scenario, figure 4 shows a session involving a survey to elicit fixed-response answers.

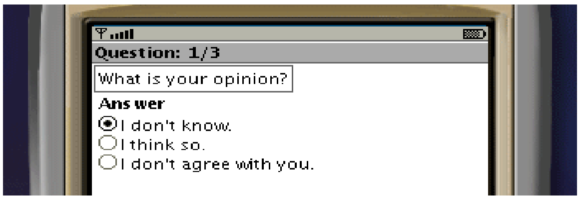

Figure 4: Perform the survey by questioning the user and selecting their responses

After the survey is completed, the data can be uploaded to the server (figure 5):

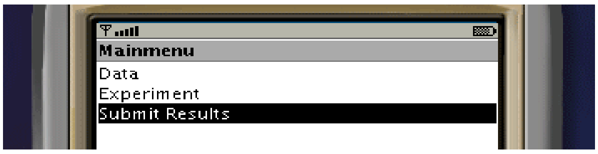

Figure 5: Submitting the data to the server

As soon as the data is on the server, it can be distributed to the rest of the practitioners assigned to that project, for later analysis (figure 6):

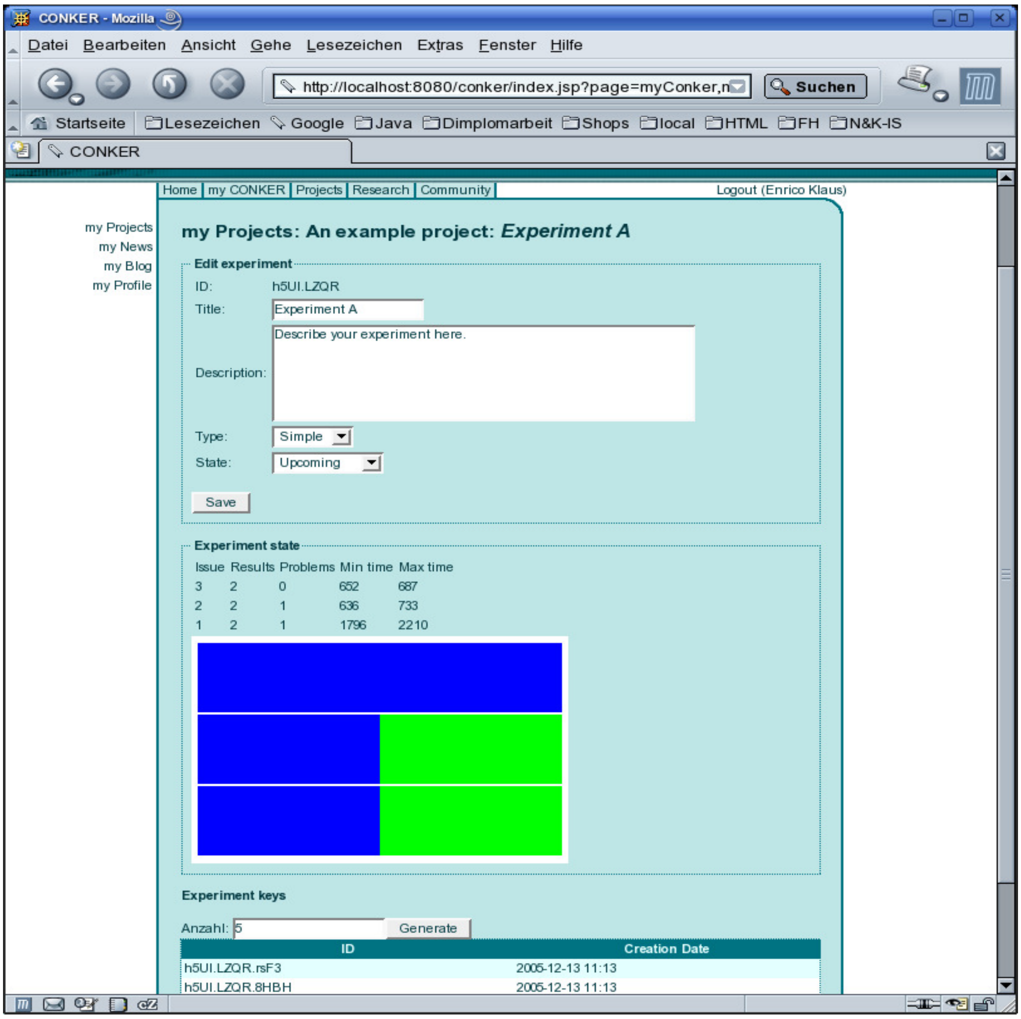

Figure 6: CONKER 2.0 demonstrating trends in the uploaded dataset from PET

PET will enable the practitioner to select between two transmission modes:

1. The “complete” mode gathers all experiment data and sends it entirely to the server.

2. The data is send to the server during the experiment (“live” mode). This allows “live” investigations and feedback by users on the server. Furthermore results from other clients can be sent back to logged on PET clients. One example would be to inform the practitioner that a particular question is inappropriate and request a change in a survey template.

7. CONCLUSION AND FUTURE WORK ON PET & CONKER v2.0

We have briefly outlined our design and development of HCI practitioner-orientated tools to support mobile cyberscience in the field of HCI Knowledge Elicitation. Although PET does not yet support many sophisticated (graphical) mechanisms to analyze the data, its pluggable architecture design allows for significant research expansion in this domain, as prior methodologies have mostly inferred low-fidelity analysis mechanisms. The opportunities for combining psychometrics with electronic tools in such circumstances by design is not yet fully realised by the electronic imaging community.

In addition, PET’s ability to handle the collection and upload of low-fidelity KE data is helpful to HCI practitioners in identifying key routes and significant outcomes of HCI participants’ decision making. Combined with the architecture of CONKER v2.0 and the expandable architecture of the Air Framework, there is significant opportunity for growth and analysis with large quantities of HCI KE data. The use of these tools and methods in the domain of cyberscience can and will aid other fields of information visualisation and management.

In forthcoming work on the PET tool, expert usability sessions are to be conducted with HCI specialists from our department, the Centre for HCI Design at City University London.

A pre-test questionnaire and real world field trials will be conducted to:

- Evaluate their current understanding of the methods available within PET.
- Present them with our mobile solution for PET and explore how they manage and track their HCI sessions in their experimental practices.
- Engage them in contributing ideas for enhancing the scope on any further requirements for use in their field operations.

User testing through remote and local observation in addition to a post-test Quality of User Interface Satisfaction questionnaire (QUIS) (Chin et al., 1988) will then be given indicate general impressions about the tools and any modifications that they think are necessary. From this data collected we hope to refine our solution into subsequent builds.

From a technology improvement perspective, we have understood that the subsequent generation of WiFi wireless enabled and high-resolution megapixel camera smartphones will be more suited to the activities outlined in this paper for the PET tool, and usability testing will be able to verify this claim. The J2ME aspects of the PET software will upwardly scale with advances in smartphone technology and networking along with CONKER 2.0 as its web service provider, to provide improved facilities as these technologies become more available to HCI practitioners.

For HCI practitioners we expect cyberscience methods to play a key role in advancing the quality and management of HCI KE methods, especially given its nature as a traditionally paper based scientific field. Forthcoming results will be interesting to see how usable and how empirically accurate such methods will benefit the HCI scientific community at large.

REFERENCES

1. M. Nentwich. *Cyberscience: Research in the Age of the Internet*. Published by Austrian Academy of Sciences (2004)
2. J.E. Burge. *Knowledge Elicitation Tool Classification*. PhD Thesis, Worcester Polytechnic Institute. 2001.
3. D. Mohamedally, S. Edlich, P. Zaphiris & H. Petrie. *MIKE’s CONKER – A COllaborative Non-linear Knowledge Elicitation Repository for Mobile HCI Practitioners.* Multimedia on Movile Devices / Electronic Imaging, San Jose, p.72-83. (2005a).

4. D. Mohamedally, P. Zaphiris, H.L. Petrie. *PROTEUS: Artefact-Driven Constructionist Assessment within TabletPC-Based Low-Fidelity Prototyping.* In Proceedings of HCI 2005 Edinburgh, British HCI Group, British Computing Society, UK. (2005b).
5. D. Mohamedally, P. Zaphiris, H.L. Petrie. *Mobile Interactive Knowledge Elicitation (MIKE) – Mobile Cyberscience in HCI Methods*. In Proceedings of HCI International 2005, Las Vegas USA. (2005c).
6. D. Mohamedally., P. Zaphiris, H.L. Petrie. *Asynchronous Knowledge Elicitation Through Software Based Card Sorting in Collaborative and Geographically Dispersed Environments*. IEEE Prep 2004, Hertfordshire, UK. 2004.
7. J.P. Chin, V.A. Diehl, & K.L. Norman. *Development of an Instrument Measuring User Satisfaction of the Human-Computer Interface*. ACM CHI'88 Proceedings, 213-218 (1988)

BIOGRAPHIES

Dean Mohamedally (dean@mohamedally.com) is a PhD research student at London's largest academic HCI group, the Centre for HCI Design, at City University London. Prior to this he worked as a mobile software R&D consultant, and completed an internship at Philips Research Laboratories, UK. During his BSc in Computer Science at Kings College London, he developed multi-platform tools and software components for mobile systems. His research focus is on mobile and web software tools to support cyberscience in HCI methods. He is a registered community developer for several high-profile organisations and has academic ties with Sun Microsystems, Microsoft (.Net platform, Windows Mobile, TabletPC), Palm, IBM Websphere, Nokia, Motorola iDen, Sony Ericsson and Symbian.

Stefan Edlich (edlich@acm.org) holds a Diploma in Computer Science and in Economics. He received his Doctor in Engineering (Dr.-Ing.) in 1995 from Technische Universität Berlin and was Assistant Professor at the Institute for Communications and Operating Systems (Prof. Dr.-Ing. Schindler). From 1995 to 1998 he was the head of an IT-Department (Trade-Supervision) of a german top 10 Bank. In 1999 he joined one of the most successful OO Consulting-Companies in Germany. Simultaneously he worked for BEA-Systems as a professional trainer teaching the BEA product portfolio (weblogic et al) to some of the german top 100 companies. 2002 he was called for a professorship at the University of Brandenburg (App. Sc.) covering the area of software-engineering.

Panayiotis Zaphiris (zaphiri@soi.city.ac.uk) is a Senior Lecturer at the Centre for HCI Design, City University London. Before joining (in Spring of 2002), he was a researcher at the Institute of Gerontology at Wayne State University from where he also got his Ph.D. in Industrial Engineering specializing in Human Computer Interaction (HCI). His research interests lie in Human-Computer Interaction with an emphasis on issues related to the elderly and people with disabilities. He is also interested in internet related research (web usability, mathematical modelling of browsing behaviour in hierarchical online information systems, online communities, e-learning, Computer Aided Language Learning (CALL) and social network analysis of online human-to-human interactions).