

Recording and Understanding Mobile People and Mobile Technology

Paul Tennent and Matthew Chalmers

Computing Science, University of Glasgow

pt@dcs.gla.ac.uk, matthew@dcs.gla.ac.uk

Abstract. We present an approach to recording and understanding the activity of people moving and interacting with each other via technologies such as mobile phones and handheld computers. Our focus is the combination of observational techniques, usually based on video recordings, and system-based techniques that log or instrument the technologies in use. At a higher level, we explore tools to allow sociologists and computer scientists to interact around a coherent visualisation that coupled resources usually associated with just one of these two communities of research practice. The *Replayer* system supports the creation of system logs, and the visualisation of the results in a display that is synchronised with video and audio recordings. We present a case study showing how Replayer was used in the evaluation of a mobile multi-user system called *Treasure*, highlighting evaluation results that would not easily have been discovered by more traditional means.

Introduction

In observing and understanding people's activity, a great variety of tools and techniques have been developed. Traditional methods are often strained when the subjects of a study are mobile, for example walking or even running through a city. Technologies such as mobile phones and handheld computers further complicate the work of those attempting to study and understand such situations, as people's use of mobile technology may be intimately related to and influenced by the activity of others far away. Several video cameras may be used to record activities in several locations set within some larger activity, but this brings the practical problem of synchronisation and the general problem of how to overview this material and combine it with other sources of informative material, such as data gathered automatically from the mobile technology itself.

As part of our research into how people use and interact through mobile technology, we have developed some techniques and tools to help with our observational studies of mobile technologies in use. Much of this work centres on combining the video often used by ethnographically inspired researchers with the system logs often used by technically motivated researchers. We have developed this within the Equator interdisciplinary research collaboration, in a group that combines sociologists and computer scientists. The *Replayer* system was developed to improve understanding and evaluation of the activity at hand, and also to assist the collaboration between researchers from different disciplines i.e. to give them a way to bridge between and understand each other's observations and methods. When used along with the *QCCI* tool, which helps to synchronise video of people using mobile computers, video and log data from a studied setting can be combined within a single synchronised visualisation or 'replay'.

We developed these tools in the course of a project exploring the use of technology in tourism, leisure and entertainment. We carried out short experimental trials of systems,

generally involving information about an urban area being presented on a map on each participant's mobile computer. Small markers on each map showed the locations of the trial participants. Their locations were tracked using technology such as GPS satellite positioning. Wireless networks were used to communicate between the computers. In such settings, video evidence alone often fails to provide a coherent view of the group's activity and interaction. Having several observers, each with a video camera, helps to cover a wide area of use but does not completely alleviate this problem. The software in the trial systems therefore had code added to log their use, so that each computer created and stored time-stamped logs for each trial participant. Also, as is common with such technology, participants would often walk out of the wireless network 'hotspots', disconnecting in a way that often leads to inconsistency of information among the computers—and inconsistency in the information presented to the various participants of a trial.

Motivation and Related Work

Since the mid 1980s, sociology has had an increasing influence on computer science, particularly in the area of computer supported collaborative work (Crabtree, 2003). Perhaps in contrast to its 'left field' status within the wider field of sociology, ethnomethodology (or ethnomethodologically-informed ethnography) has become established as a means to observe and discuss social interaction, and to inform the design of collaborative systems (Garfinkel, 1967). The initial focus was workplace studies, and the development of system design requirements (Hughes, 1995; Sommerville, 1993). Examples of such studies include those of news journalists (Fagrell et al., 1999), architects (Suchman et al., 2002) and bank workers (Harper et al., 2000; Randall & Hughes, 1995). More recently, ethnography has been used to study technologies in use in other settings, such as virtual environments (Hindmarsh et al., 2000), the home (Crabtree et al., 2003), community care (Cheverst et al., 2003), tourism (Brown et al., 2003 A; Brown et al., 2003 B), education (Randell et al. 2004), mobile phones (Eldridge & Grinter, 2001) and mobile games (Crabtree, 2004). While many of these studies were of one fixed locale, making traditional forms of observational analysis relatively easy to apply, there is an increasing amount of attention given to mobile use of technology. The cultural and economic significance of mobile technology is something that we are all increasingly aware of. Studies of technology and society have begun to respond but, in simple practical terms, traditional observational techniques become more difficult to apply. Keeping up with the activity is difficult, and even then small devices such as mobile phones and PDAs can easily be occluded from view, so that much interaction with devices cannot easily be observed.

These shortcomings suggest a need for the recording of additional data. While the focus of observation should not be shifted away from the people in the setting, it can be expanded to include more systemic data. The recording and playback of systemic data is a well-travelled path, with systems such as (Gray et al., 1996; Hilbert & Redmiles, 2000; Hochheiser & Shneiderman, 2001; McLeod et al. 2004) instrumenting code and providing some replay of data. Indeed some systems such as (Orso et al., 2004) provide visualisations that are so complex, they cannot easily be understood by other than trained information visualisation specialists. While such visualisations may be useful in their hands, they are far from ideal as a tool to support people beyond computer science, such as sociologists.

Typically observational analysis involves the use of several media. These include, but are not limited to: video recordings of systems in use, audio recordings, notes taken in the field, questionnaires, interviews, and textual logs and statistical graphics derived from systemic

data. Each of these media provides different parts of a larger whole, but are often difficult to combine. The evaluation of systems such as (Benford et al., 2004; Benford et al. 2005; Brown et al. 2003; Flintham et al. 2003; Reeves et al., 2005) involved attempts to combine video analysis with systemic replays, but are dogged by issues such as understandable visualisation design, and synchronisation and cueing of data, making it virtually impossible to effectively use the two media in conjunction without expending considerable time and effort. Reflecting on this prior work, we were motivated to develop a toolkit, *Replayer*, to serve as a flexibly applicable system for the creation, collation, annotation, synchronisation and visualisation of a mixture of data for analysis.

Replayer

Replayer represents a hybrid approach to analysing activity and evaluating mobile technology. We take several of the methodologies described above and weave them together to create a cohesive approach. We see this as a pragmatic response to the way that recordings of mobile distributed experiences may consist of large heterogeneous data sets. Issues such as synchronisation and data volume can make a manual approach to constructing and analysing the relationships between different recordings infeasible. One approach to this issue is keying: manually examining a single data set in great detail, for example a video recording of a system trial, and then using it as a ‘key’ to select important or relevant periods of time in the experience to investigate with the help of relevant subsets of other data such as system logs.

Instead of using a single data set to cue the others, *Replayer* uses any or all the data sets as keys or indices. The principle is to present all the data together, in a coherent manner, using multiple synchronised tools, and then allow the evaluator to select in one tool what he or she believes is important at any given point—with the system offering support by presenting temporally corresponding data in other tools. By coherently presenting a variety of data for any time, it is our intention that *Replayer* will help provide a sense of the context of a selected event. In practice, and as shown in Figure 1 (below), this has resulted in four distinct types of tool in *Replayer*: video, audio, systemic data presented as text, and systemic data presented graphically.

Each type of data may have several instances, for example in the case study presented later there were typically two video data sets shown simultaneously in separate tools. We offer one central tool for the synchronised control of all tools, but also, as every tool has its own set of controls, each can be played independently of its peers for more focused analysis or to compare an episode in one tool with another episode from a different time. Additionally, each video and audio tool has an independent volume control, allowing the sound relevant to a particular piece of analysis to be selected.

The textual systemic data provides a particularly powerful key or index. It is easily searchable, in contrast to both audio and video, and so we can search for a logged event, potentially on a number of criteria, and find both the system-oriented information about it, and its social or personal context as gleaned from the video and audio recorded around the same time. Of course it is likely that there will not be complete video and audio data for every part of an experience involving multiple people distributed over a wide area. We may then search for an event, and be taken to that point in the video and audio streams, and then see at a glance whether we have relevant images or sound for that event. The systemic data further provides us with easy access to numerical or statistical data about the system’s use and the activity of those using the system, e.g. *Replayer*’s text search tool offers facilities for counts

and means of particular events. Additionally, the text search tools let us use the available temporal and spatial information to quickly generate spatial and temporal distributions of events to support analysis, popping up histograms and summary maps on demand.

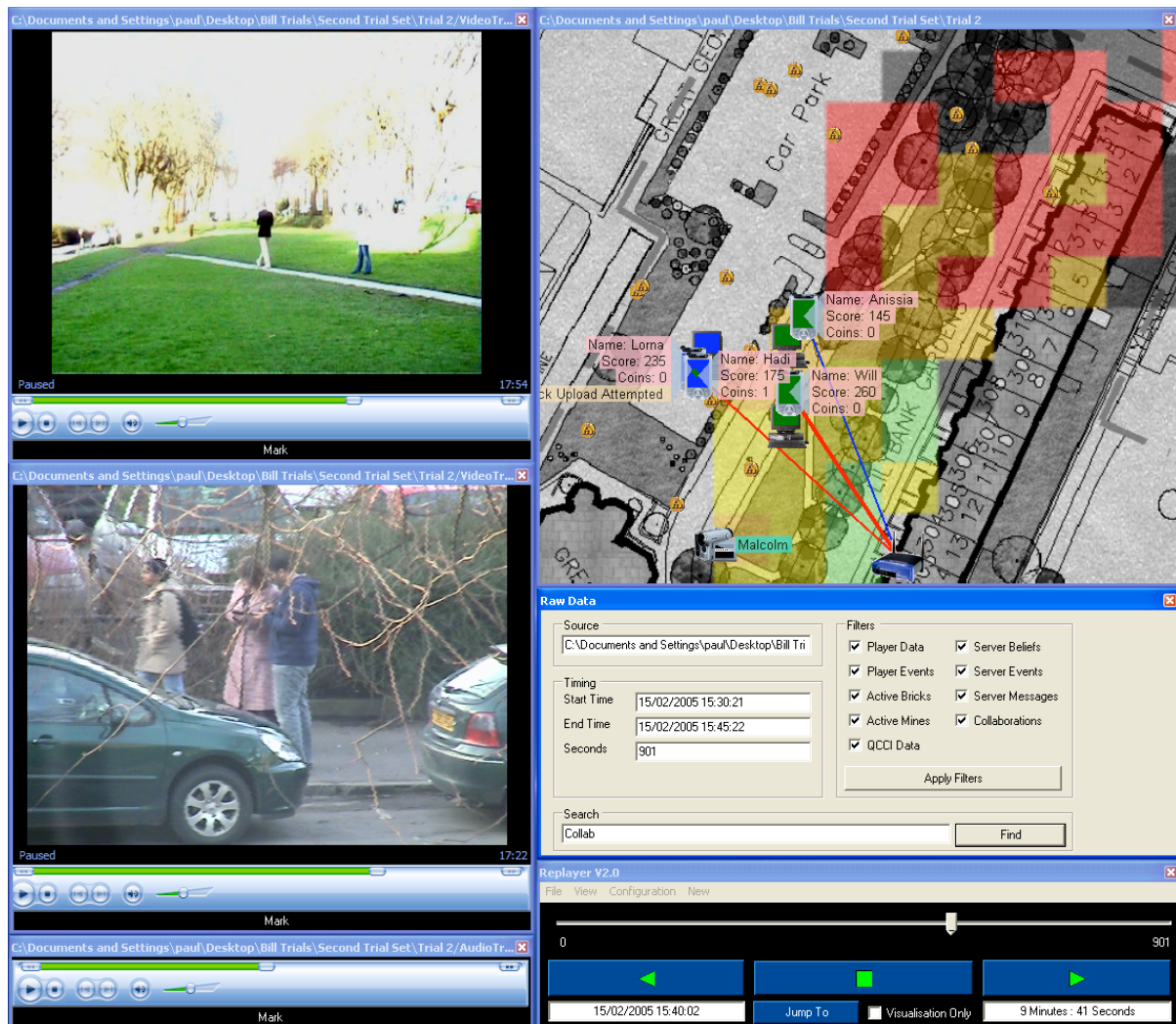


Figure 1. Replayer consists of a number of tools, in this case two video players (top left and centre left), an audio player (bottom left), an aggregate log visualisation (top right), a text search tool (centre right) and a playback control tool (bottom right).

One possibly under-emphasised aspect of creating systemic data is that the programmer's task of instrumenting code, to create system logs, can be error-prone. For example, minor variations in log data formats can cause unnecessary work to make data consistent and usable in analysis. Replayer allows easy instrumentation of code from within the Microsoft Visual Studio .NET integrated development environment. An add-in, a piece of software rather like a plug-in for a web browser, allows instrumentation to be directly applied to an individual program variable, method, class or solution with a single click, inserting code to record log data in each case. When the program is run, time-stamped and consistently structured log entries will be created by each of the instrumented parts of the program that are executed. More meaningful messages can be added to the output, making the logs more readable to those less familiar with the code—and searchable by them too. Replayer does not depend on this technique being used to create logs, requiring only that the logs it uses for input are in the correct format.

Another aspect of system logging that is often under-emphasised is inconsistency. In a multi-user system one can expect multiple devices. Therefore it is likely that no one device will be able to record all the required log data alone. Just as several video recordings may be required, systemic data capture must also be distributed, that is, each device must record its own independent logs. Logs may be inconsistent, in that different devices may have different data about supposedly shared system state and people's activity. For example, mobile devices may intermittently gain user location data from a central server, and so present different apparent locations for the same people. Since such inconsistency may have strong effects on the user experience, it is often enlightening to display and explore such conflicts. For this reason, Replayer collates all the devices' logged data within one temporally-based aggregate data structure, based on a regular subdivision of the time period of the user experience into a series of time-stamped 'frames'. Each frame reflects the distributed system at a given time and is examinable independently of those around it. This technique allows us to 'jump around' in the playback.

Visualisation

Replayer uses three main techniques for visualising the systemic log data. The first of these is a geographical representation of a frame. We run a map server that lets us locate and display a map covering the complete area covered by the users in the course of a user experience. This map can then be zoomed or panned to show increased detail for any specific area. State data that can be displayed by discrete 'zoning' is overlaid. An example of this is the coverage of a wireless network, as sampled by users' mobile devices. We split the map into a grid, and use alpha-blended coloured squares and a simple tripartite scale: red for low, yellow for medium and green for high values. The logged GPS locations of users are overlaid as icons on the map. We also add icons for geo-referenced artefacts such as wi-fi access points. This is important as systems using wi-fi only work within a relatively small distance of these access points, and radical changes in system behaviour (and the ability of people to communicate through the system) occur when people move away from these areas of wi-fi coverage. The network connections that PDAs make to wireless access points can also be displayed as lines drawn between the locations of the users and the locations of the access point each user's device is communicating with, if any. We represent the network signal strength by the line's thickness. Additional state data is then added textually, e.g. state data about a particular user or artefact is displayed beside it. Events are handled in the same way; they are displayed beside the relevant user or artefact. All this text can lead to an extremely cluttered screen, a problem addressed in two ways. First, a simple excentric labelling technique is applied to stop text from overlapping with icon it refers to. Secondly, text and icons are rendered at a constant size (in terms of pixels) so that zooming in separates them out.

Replayer's other visualisations are simpler temporal and spatial distributions respectively, and are used primarily with event data. In the former case, Replayer can show the temporal distribution of any selected event in a simple histogram, in which all the instances of selected event(s) are displayed along a temporal axis as in Figure 2. In the latter, the selected event(s) are displayed spatially based on the same map system mentioned earlier but showing aggregates or averages over all frames rather than data from one frame. As shown in Figure 3, additional state data can again be attached textually to each event, and a legend is set in one corner of the image.

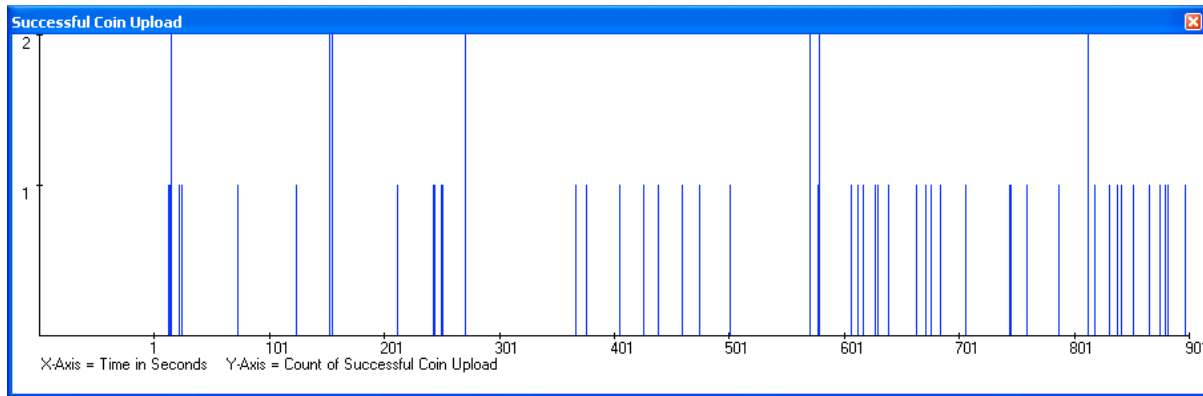


Figure 2. Temporal distribution of Upload events for all players in a single game of Treasure. The x-axis is time in seconds from the beginning of the game, and the y-axis is the number of such events in that second.

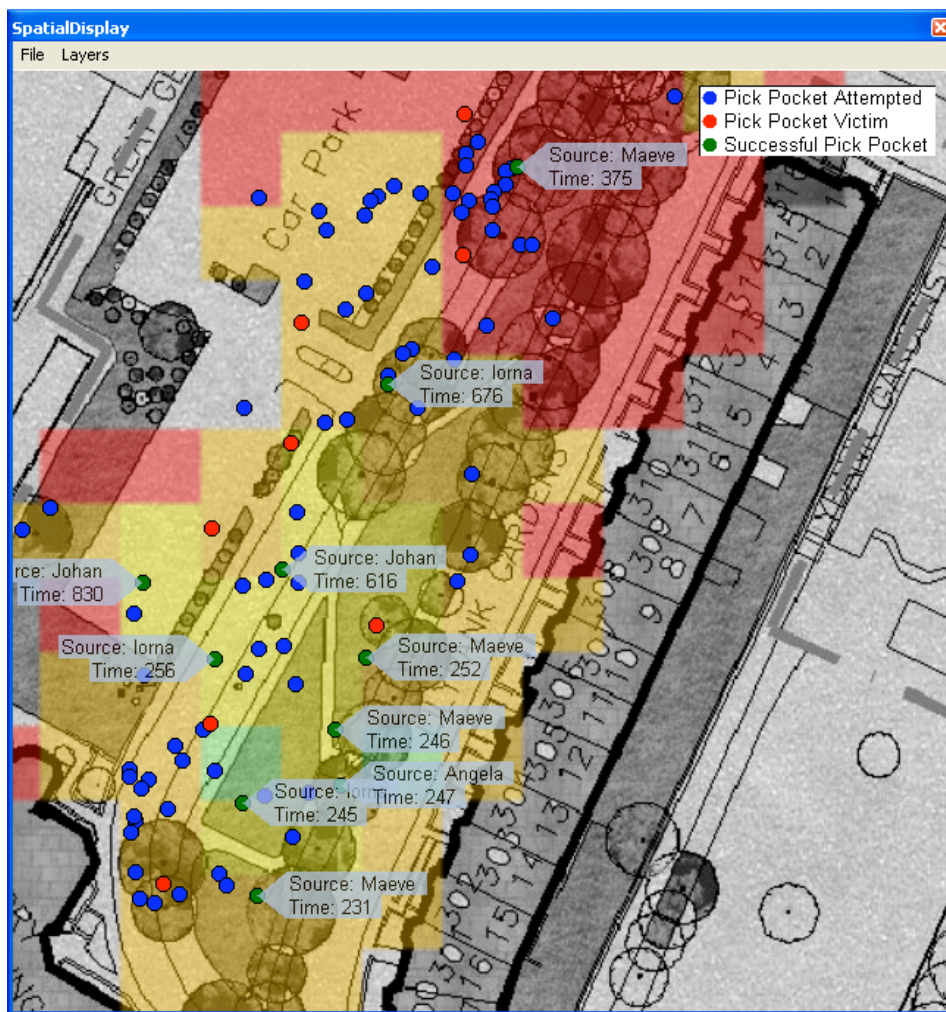


Figure 3. The spatial distribution of PickPocket events in one run of the mobile multiplayer game Treasure. For each successful PickPocket, the name (Source) and time (in seconds) of the event is shown in a label.

Distributions such as these are common in analysis (Benford et al., 2004; Benford et al., 2005; Brown et al., 2003; Brown et al., 2005), but can take considerable time and effort to create. Replayer aims to make their creation more lightweight, in that they can be quickly made, compared and adapted.

Synchronisation

Synchronisation is a fundamental feature of a system such as Replayer. Without accurate synchronisation, one cannot easily relate one data set to another. There are two areas where this is required. First, all the devices in a trial must be synchronised for support aggregation and collation of logs. Synchronisation can potentially be done *post hoc*, but this is a difficult process. It is preferable to do this either before or as part of the system's use. Replayer therefore provides a set of software classes for remote synchronisation of mobile devices over a wireless network. Synchronisation of these devices is therefore made relatively trivial. However, synchronising video and audio tracks with the system logs is more complex. One possibility is to set the clock on the video/audio recorder before the trial, however once the data is captured from the device and stored on a computer, the absolute start and end times are generally not stored. We propose an alternative technique based around a tool called QCCI (pronounced Quickie).

QCCI is an application designed to be run on the PDA given to each evaluator of a system in use, i.e. each observer who goes out into the 'field'. It solves the audio/video synchronisation problem by displaying a clapperboard image on the PDA screen (figure 4).



Figure 4. The QCCI clapperboard, used to synchronise observers' video.

The time on the PDA is synchronised to the other devices, either by the remote method described above, or by hand before the trial begins. To synchronise a video track, the observer's video camera is pointed at the screen of the QCCI device and the system time is then part of the film. Later, when opening that video stream in Replayer, the evaluator has only to find a frame where the clock is displayed and type the time into Replayer. Replayer will then jump to that point in the systemic data set, and store the offset between the start of the video and the start of the systemic data. Once this is stored the two are permanently synchronised and can be used to cue one another effectively. Audio synchronisation is handled similarly. At the press of a button QCCI makes a noise, which is recorded on the

audio track, and the time at which the button was pressed is logged by QCCI. The point at which the noise occurs in the audio track has to be found, and then the point in the systemic data where the button is registered, recording the offset in the same manner as for video. In addition to synchronisation, QCCI leverages the power of giving a PDA to each evaluator in additional ways. GPS is used to record the position of the evaluator, and spatially and temporally referenced audio annotations can be made. Recording the spatial and temporal location of the evaluators can add vital context to their field notes.

Case Study: Treasure

Treasure (Chalmers et al., 2005) is a mobile multiplayer game developed at the University of Glasgow. Played on an area of around 7000 square meters, the game consists of two teams of two players; each with a GPS and wi-fi enabled PDA. The PDA displays a map of the local area, and periodically coins appear on the map (see figure 5). Runners must move physically to the location where the coin is displayed on the map, then press the *Pickup* button to retrieve the coin.

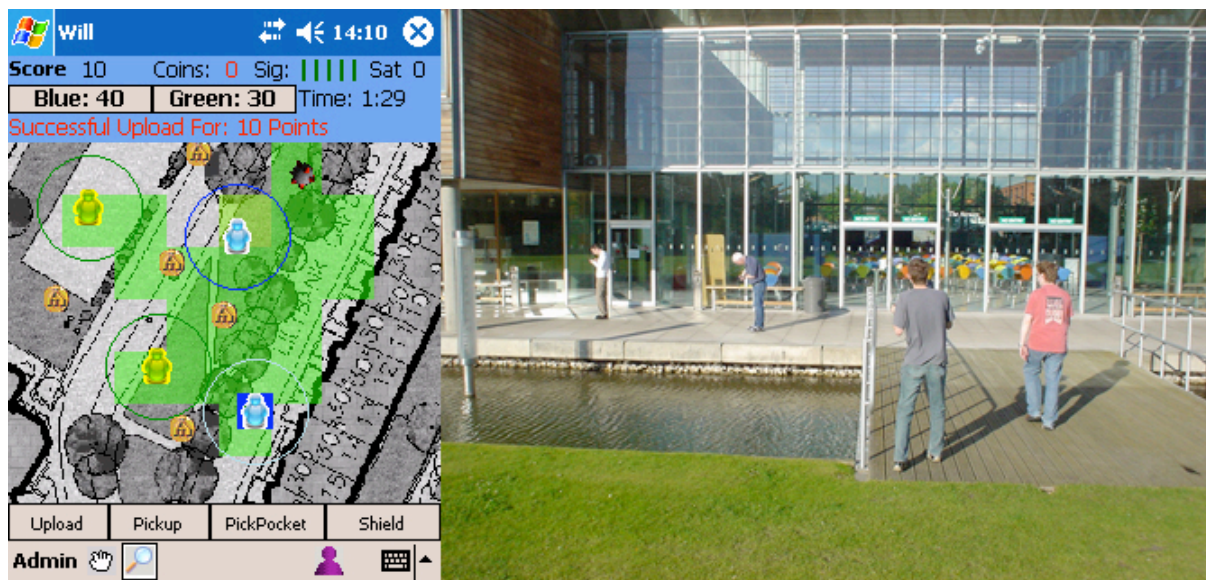


Figure 5. Treasure's PDA interface (left), four Treasure players (right). It is hard to determine what the system or the runners are doing from visual data like this, as we cannot see the individual runners' PDAs.

Once carrying coins, the runner must attempt to upload them over the wi-fi network to the game server to score points. Because the wi-fi network does not cover the whole area, runners are forced to survey the network as they play in order to locate 'safe' places to upload. In the course of this survey, runners build a shared wi-fi map of the area. Finally, interaction between runners is encouraged by two features: that of Pick Pocket and Collaborative Uploading. Pick Pocket allows runners to steal coins from their opponents which been picked up but have yet to be uploaded. When both players from the same team upload simultaneously, a collaborative upload is said to have occurred, and players are rewarded with double points.

The evaluation of treasure was concerned with examining how runners' strategies evolved as their experience in the game increased. Participants played between one and three games each, against different opponents. Each game lasted for 15 minutes, with a five minute 'warm

up' game to acclimatise players to the system. In order to conduct this style of evaluation, which relies on a mixture of numerical and contextual data, the Replayer system, described above was deployed to aid the analysis of collected data.

Treasure embodies many of the difficulties to be found when evaluating mobile multiplayer systems. It is fast moving, played over a reasonably large area, runners use PDAs with small screens, and at no time in the game does any one device - including the server - have complete information about the state of the system. This makes the system an ideal candidate for Replayer's aggregate logging technique. Each PDA therefore records its own logs, as does the server and these are subsequently combined by Replayer into a more usable format.

Because log data alone would lose much important social and contextual information, videos were also used. Two cameras were deployed for each session - one in the field and one from an upstairs window. Each of these camera positions has distinct advantages and disadvantages. The field camera is often close enough to record audio, however, it is typically unstable, as the operator is often forced to chase the players to keep them in shot. Additionally the field camera is unable to record the actions of all the players simultaneously due to the distributed nature of the game. Conversely, the overview camera is very steady, and the operator is able to get a better view of the game and can focus on players unseen by the field camera. However, because there is no audio, this camera does not provide the same level of contextual information as its peer.

Data analysis in Treasure

The analysis of the data collected during the trials can be broken down into three main stages. these are respectively: extracting general numerical data; looking for interesting games; and examining these games in more detail. Each game was loaded into Replayer so that numerical data could be quickly and automatically retrieved. For each game, final scores, numbers of uploads, pickpockets etc. per player were noted then Microsoft Excel was used to create graphs showing how these data changed over subsequent games. The numerical data served to give a feel for the style of a game, so it was from this indication that a subset of games was selected for further analysis. Once a game had been selected, the interview was listened to, giving an indication about how the runners felt about the game. The interview also suggested to the evaluator some key events to look for in the data. It is worth noting however, that the data often did not match these expectations. When examining a game the following process was used: Replayer's data search function allowed the evaluators to jump to each key event in the game, and examine the videos surrounding that event. Additionally, Replayer's spatial and temporal distributions were used to show more context information about events. Finally, the videos were watched from start to finish, with times of interest noted down. These times were then used to locate from the systemic data exactly what was happening. This last point is important as it revealed points of interaction that were not easily apparent in the systemic data alone.

Using the Replayer approach to evaluation provided us with information that would have been very time consuming to come by in more traditional data. An example might be one game in which the players claimed in the interview that the reason they had walked around together was that they had been collaboratively uploading, and indeed the video appeared to confirm this. However, the systemic data showed a different story—in fact they did not collaboratively upload once throughout the game. The players explained their strategy but their scores and the system logs did not reflect it. They were either following a different strategy to the one they claimed or they were following the strategy but unable to carry it out,

in technical terms. Another example stems from the nature of the game. Players move around erratically, starting and stopping, suddenly changing direction, etc. When watching the videos, it is difficult to establish the cause of this behaviour, indeed there are many possibilities; players may be moving between coins, they may be trying to understand the map (Brown & Laurier, 2005), or they may be participating in a Pick Pocket. Without the benefit of synchronised systemic data, it is very difficult to select between these possibilities.

Conclusion

The design of Replayer was motivated by the pragmatics of trying to observe and record a number of people moving and communicating. Also, however, it was our reaction to the tension between different styles of analysis and evaluation of mobile systems. On one side we have the more exploratory and qualitative assessment often favoured by ethnographers, and on the other is the more hypothesis-driven and quantitative approach often favoured by technologists. Replayer does not subscribe completely to either of these views instead seeking to enhance each with aspects of the other. As was shown in the case study, both these approaches contribute to a more complete picture. How Replayer is used—that is, which streams are primarily used to key the other—is up to the individual evaluator and one can shift from one choice of primary medium to another as the need arises. Replayer is not intended to present either technique as primary, but rather is intended to broaden and interconnect the range of views and options for understanding people's activity.

In its current state, Replayer highlights a number of challenges. Perhaps the most important of these is that knowledge of the log entries is generally required in order to search for particular events. Some kind of lexicon or legend is therefore required to show to the evaluator what text one might search for. A related issue is that the tools for searching are rather simple; in this initial version of Replayer, searches are limited to one event name. Another consideration is that of speed. An average desktop computer struggles to play back several video streams simultaneously, meaning that in practice one cannot realistically watch many streams in parallel. In time, commodity hardware will advance, but in the short term we plan to use two or more networked computers to share the load. Finally, there is a need for more integration with other tools. It would be helpful to provide links to statistical tools such as SPSS and Excel; indeed, Excel was used in the Treasure case study, to address something that Replayer currently cannot—comparing games played at different times. In effect, what one would have to do now is to run two independent copies of Replayer, each with a different data set. Replayer combines data from the same time, but we have yet to implement features that will support direct comparison of activity from different times.

Overall, our tools aim to simplify the analysis process by providing tools for programmers and sociologists to work together in creating and understanding the use of mobile technology and each other's work practices. It attempts to combine quantitative and qualitative materials, in a way that supports their comparison and combination without privileging one over the other. Our initial experience with applying Replayer to Treasure has been informative but, much like its use to study that game, we intend to use it to study a number of other systems, thereby affording understanding and redesign of Replayer as a piece of technology and as a focus for social interaction among technologists and sociologists.

Acknowledgements

This work was carried out within Equator, funded by the UK EPSRC (GR/N15986/01). The authors offer thanks to Equator colleagues past and present, especially Barry Brown, Louise Barkhuus, Marek Bell, Malcolm Hall, Scott Sherwood and Paul Rudman.

References

- Benford, S., Seagar, W., Flintham, M., Anastasi, R., Rowland, D., Humble, J., Stanton, D., Bowers, J., Tandavanitj, N., Adams, M., Row Farr, J., Oldroyd, A. and Sutton, J. (2004) *The error of our ways: the experience of self-reported position in a location-based game*, Proceedings of the 6th International Conference on Ubiquitous Computing, pp. 70-87, Nottingham, UK: Springer.
- Benford, S., Rowland, D., Flintham, M., Drozd, A., Hull, R., Reid, J., Morrison, J. and Facer, K. (2005) *Life on the edge: supporting collaboration in location-based experiences*, Proceedings of the 2005 CHI Conference on Human Factors in computing Systems, Portland, Oregon: ACM Press, pp. 721 - 730.
- Brown, B and E. Laurier (2005) *Maps & journeying : an ethnomethodological approach*. To appear in: Cartographica
- Brown, B., MacColl, I., Chalmers, M., Galani, A., Randell, C. and Steed, S. (2003) *Lessons from the Lighthouse: collaboration in a shared mixed reality system*, Proceedings of the 2003 CHI Conference on Human Factors in Computing Systems, pp. 577-585, Florida: ACM Press.
- Brown, B., and M. Chalmers (2003) *Tourism and mobile technology*, Proceedings of the 8th European Conference on Computer Supported Cooperative Work, pp. 335-355, Helsinki, Finland: Kluwer Academic Press.
- Brown B, Chalmers M, Bell, M, Hall, M, Rudman P, (2005) *Sharing the square: collaborative leisure in the city streets*, To Appear in the 9th European Conference on Computer-Supported Cooperative Work, Paris, France.
- Chalmers, M, Bell, M, Brown, B, Hall, M, Sherwood, S, Tennent P (2005) *Gaming on the Edge: Using Seams in Ubicomp Games*, To Appear in ACM SIGCHI International Conference on Advances in Computer Entertainment Technology, Valencia, Spain.
- Cheverst, K., Clarke, K., Fitton, D., Rouncefield, M., Crabtree, A. and Hemmings, T.(2003) *SPAM on the menu: the practical use of remote messaging in community care*, Proceedings of the 2003 ACM Conference on Universal Usability, pp. 23-29, Vancouver, Canada: ACM Press.
- Crabtree, A. (2003) *Designing Collaborative Systems: A Practical Guide to Ethnography*, London: Springer-Verlag
- Crabtree, A. (2004) *Design in the absence of practice: breaching experiments*, Proceedings of the 2004 ACM Symposium on Designing Interactive Systems, pp. 59-68, Cambridge, Massachusetts: ACM Press.
- Crabtree, A., Rodden, T., Hemmings, T. and Benford, S. (2003) *Finding a place for UbiComp in the home*, Proceedings of the 6th International Conference on Ubiquitous Computing, pp. 208-226, Seattle: Springer.
- Fagrell, H., Ljungberg, F. and Kristoffersen, S. (1999) *Exploring support for knowledge management in mobile work*. in Proceedings of European Conference on Computer Supported Cooperative Work 99, Kluwer Academic Press, Copenhagen, Denmark, pp. 259-269.
- Eldridge, M. and R. E. Grinter (2001) *Studying Text Messaging in Teenagers* Position Paper for ACM Conference on Human Factors in Computing Systems (CHI 2001) Workshop

on Mobile Communications: Understanding User Adoption and Design. Seattle, Washington.

Flintham, M., Anastasi, R., Benford, S., Hemmings, T., Crabtree, A., Greenalgh, C., Rodden, T., Tandavanitj, N., Adams, M. and Row-Farr, J. (2003) *Where on-line meets on-the-streets: experiences with mobile mixed reality games*, Proceedings of the 2003 CHI Conference on Human Factors in Computing Systems, pp. 569-576, Florida: ACM Press.

Garfinkel, H. (1967) *Studies in Ethnomethodology*, Englewood Cliffs, New Jersey: Prentice-Hall.

Gray, M; Badre, A; Guzdia, M. (1996) *Visualizing usability log data*, Proceedings of the 1996 IEEE Symposium on Information Visualization, pp. 93, Washington, DC, USA: IEEE Computer Society.

Harper, R., Randall, D. and Rouncefield, M. (2000) *Organizational Change in Retail Finance : An Ethnographic Perspective*. Routledge, London.

Hilbert, D.M. and Redmiles D.F. (2000) *Extracting Usability Information from User Interface Events*, ACM Computing Surveys, vol. 32, pp. 384-421.

Hindmarsh, J., Fraser, M., Heath, C., Benford, S. and Greenhalgh, C. (2000) *Object-focused interaction in collaborative virtual environments*, ACM Transactions on Computer-Human Interaction, vol. 7 (4), pp. 477-509.

Hughes, J., O'Brien, J., et al. (1995) *Presenting ethnography in the requirements process*. in Proceedings of second IEEE international symposium on requirements engineering, York, 27-35.

Hochheiser, H. and Shneiderman, B. (2001) *Using interactive visualizations of WWW log data to characterize access patterns and inform site design*, Journal of the American Society for Information Science and Technology, Vol. 52 (4), John Wiley & Sons Inc, New York, NY, USA, pp 331-343.

McLeod, I; Evans, H; Gray, P; and Mancy, R (2004) *Instrumenting Bytecode for the Production of Usage Data*, Proceedings of 5th International Conference on Computer-Aided Design of User Interfaces, pp. 185-196.

Orso, A; Jones, J A; Harrold, M J; Stasko, J. (2004) *Gammatella: Visualization of Program-Execution Data for Deployed Software*, presented at 26th International Conference on Software Engineering, pp. 699 - 700.

Randell, C., Price, S., Rogers, Y., Harris, E. and Fitzpatrick, G. (2004) *The ambient horn: designing a novel audio-based learning experience*, Personal and Ubiquitous Computing, vol. 8 (3), pp. 144-161.

Randall, D. and Hughes, J.A. (1995) *Sociology, CSCW and Working with Customers*. in Thomas, P. ed. The social and interactional dimensions of human-computer interaction, Cambridge University Press, 1995, 142-160.

Reeves, S; Fraser, M, Schnadelbach, H; O'Malley, C; Benford, S. (2005) *Engaging Augmented Reality in Public Places*, Adjunct proceedings of CHI Conference on Human Factors in Computing Systems, Portland, USA

Sommerville, I. (1993) *Integrating ethnography into the requirements engineering process*. in Proceedings of the international Press, Los Alamitos, California

Suchman, L., Trigg, R. and Blomberg, J. (2002) *Working Artefacts: Ethnomethods of the prototype*. British Journal of Sociology, Vol. 53 (2). 163-179.