Gaming on the Edge: Using Seams in Pervasive Games

Matthew Chalmers, Louise Barkhuus, Marek Bell, Barry Brown, Malcolm Hall, Scott Sherwood & Paul Tennent

Department of Computing Science
University of Glasgow
Glasgow, UK

matthew@dcs.gla.ac.uk

ABSTRACT

Outdoor multi-player games are an increasingly popular application area for pervasive computing, supporting experimentation both with new technologies and new user experiences. This paper presents a set of experiments with an outdoor pervasive game that exploits the gaps or seams that exist in complex computer systems. The Bill game is designed so that players move in and out of areas of wireless network coverage, taking advantage of the connectivity within a wireless 'hotspot' and also of the lack of connectivity outside it. We draw lessons for how such games can successfully encourage social interaction between players, discuss the interaction between the game and the local environment, and describe our approach to recording and 'replaying' such games. More broadly, this paper discusses how the notion of *seamful design* can be a source of design ideas for such games.

1. INTRODUCTION

Games have long been one of the most popular applications of technology, both in terms of their impact on culture and their significant financial success [8]. Games have also been a key motivator in the development of many new technologies and techniques, particularly in the areas of computer graphics and artificial intelligence [3]. Yet games in their own right have only recently attracted sustained academic attention [9]. One avenue of recent investigation has been *mobile* and *ubiquitous* games [1, 6, 10] through which a number of broader research themes (such as mixed reality) have been investigated. Games such as 'Can You See Me Now?' [6] explore the incorporation of urban environments and digital systems, forming games to support unusual yet enjoyable connections between online players and players on the street.

In this paper we describe our experiences with an outdoor mobile game that exploits seams. A seam is break, gap or 'loss in translation' in a number of tools or media, designed for use together as a uniformly and unproblematically experienced whole. Our designs draw upon the concept of 'seamful design' put forward by Mark Weiser [2, 11], but echoing established media theory, in which designers take advantage of the physical limits and characteristics that constitute a design medium rather than smoothing them out or ignoring them.

For example, many applications for mobile computers may be built as if they could be used along with the features of the environment one travels through, e.g. to display web pages about nearby buildings and people. Such applications often assume constant network connectivity, and yet this is not always the case when mobile systems really are mobile; as one walks away from an access point, such systems often crash or become unusable as the wireless network drops off and then disappears. Applications may be built to be uniform and 'seamless', but the seams of their infrastructure often show through in interaction.

Inspired by the seamful design idea, we designed a game, *Bill,* in which each player uses a handheld PDA equipped with GPS and 802.11. Players collect virtual 'coins' from outside the wireless network, and then runs back into network range to 'upload' the coins to gain points. Game strategy is based on observing, understanding and taking advantage of where coins and players are, hotspots of network coverage and the 'cold spots' out beyond them, and the urban setting of the game.

This paper describes our iterative design process, in which we ran 14 observed trials of Bill, involving 46 players in all, and experimented with a number of different game features. Based on in-depth analysis of the game play, we focus on a number of design issues in our game that may be useful for designers of future mobile and pervasive games. Firstly, we explore social interaction in these games, and the way that players achieve much of their enjoyment through collaboration and competition. Second, we discuss the management of interaction between the digital game and the urban environment, in particular the importance of the environment in affecting gameplay. We also discuss the benefit of using replays of games by designers for analysis and also by players for understanding and developing tactics. We also present some of our ongoing and future work on newer games that also use seams. Overall, we suggest that this concept can be a productive resource for designing games and, potentially, for the design of ubicomp systems.

2. MOTIVATION AND RELATED WORK

While games offer considerable value and enjoyment for players, they also have the potential to be used by designers and researchers as experimental platforms for new technologies and design concepts [4]. Along with our desire to create a successful game, we were also interested in using a game to experiment with Weiser's concept of seamful system design. In particular, we were interested in how the use of infrastructure such as 802.11 wireless networks can be very apparent in people's interaction with mobile computers. What may be 'infrastructure' to a system designer may be something that users have to understand, handle or react to—which perhaps then should be designed for in the interface and interaction design, rather than ignored or suppressed.

Wireless networks have distinct physical characteristics such as a tendency to be absorbed by metal, water and other conductive materials, and a pattern of coverage that makes for a limited area of usable network connectivity. Usable network coverage, therefore, rarely covers all the areas one moves through and spends time in during a day of work or leisure.

Furthermore, when one takes a mobile computer into network coverage, one can receive and transmit information to other machines and other players but one might also be tracked, recorded or spied upon electronically. Out of network coverage one can, then, be more private or 'safe' from these negative aspects of network use, and do parts on one's work or leisure that do not rely on network access—deferring or avoiding parts of one's work or leisure that do rely on network access, of course. Depending on one's context, one might wish to be in a network hotspot—but at other times one might prefer to be outside the network.

This tension between the good and bad aspects of being either in or out of a network appears, to us, to be a seam in the sense that Weiser discussed. Where there is and is not coverage, and the context–specific choice about whether to be in network coverage or out of it, are aspects of network use that are common in use but under–represented in the design of ubicomp systems and user experiences. Instead, most such systems are built on the assumption that all use of the system happens within network coverage.

Bill's design used this seam in particular as a starting point. The setting of a mobile multiplayer game let us experiment without requiring long–term commitment from users (i.e. players), or demanding that they have any conceptual understanding of the notions of seams and seamful design. Instead, players would be using game limitations or constraints in a way that is commonplace in games, in the form of limits, boundaries and rules [9].

There are several research projects that have used outdoor gaming as a means of exploring new research ideas, and a number of games that work with similar design features and techniques. 'Can You See Me Now?' (CYSMN) [6] linked on-line and street players in a chase game. Street players (*runners)* moved around the game area covered by a game–specific wireless network, and had their positions tracked by GPS. On-line players used arrow keys to move themselves around a 3D view of the same streets, with icons showing the locations of runners. Similarly, online players' positions were shown on the mobile computers carried by runners. Runners chased on-line players through the city, making their GPS positions match the on–line players' positions i.e. 'catching' them. In playing CYSMN, the variable accuracy of GPS caused problems for street players when trying to catch players in areas of bad GPS coverage. However, as the game progressed, street players became more skilled at using their knowledge of good and bad GPS areas, luring online players into areas of good GPS where catching them was easier. In this way the players took advantage a limitation—a seam—of the game's construction to their advantage, but the game was not designed to make explicit use of this.

Another game that was influential in the design of Bill was NodeRunner (www.noderunner.com), which made use of the wireless network infrastructure existing in a city. Each team had a PDA equipped with 802.11 and a camera. Teams of players raced against time, logging as many wireless access points as they could and uploading photographic proof of each find to a central server. While NodeRunner made original use of the existing invisible wireless infrastructure, it made no use of the signal beyond the existence of access points.

The 'Pirates!' game [1, 5] used RF technology to determine the proximity of players to one another and specific resources. The game mapped an ocean environment on to the real world and players took the role of a ships commander travelling from island to island trading and fighting in order to gain wealth. The underlying RF infrastructure was mapped to specific game events so that when a player came close to a RF beacon representing an island, a game event was triggered. In particular, face–to–face interaction was a key part of the game, encouraging some of the social aspects of gaming that can be lost in some computer game designs.

3. OVERVIEW OF THE *BILL* GAME

The main aim of a Bill player is to collect 'coins' placed in areas of poor network coverage, and then bring these coins back into an area of good network coverage to gain points. By moving in and out of areas of network coverage, players also inadvertently survey the wireless network they are playing in, building up a changing map of network coverage that they all share.

At the beginning of the game, each player is given a PDA with GPS and 802.11 wireless capabilities. The PDA interface (Figure 1) consists of a map that supports panning and zooming, on which the player's location, the location of coins, and the location of other players are displayed. To pick up a coin, a player must walk or run to the location of the coin as indicated on the map, so that his or her GPS–tracked location is close to the coin's location, and then press the *Pickup* button. For the player to gain points for this coin he or she must then walk or run to an area of sufficiently high network signal strength and click *Upload* so as to send the coins they have collected to the game server.

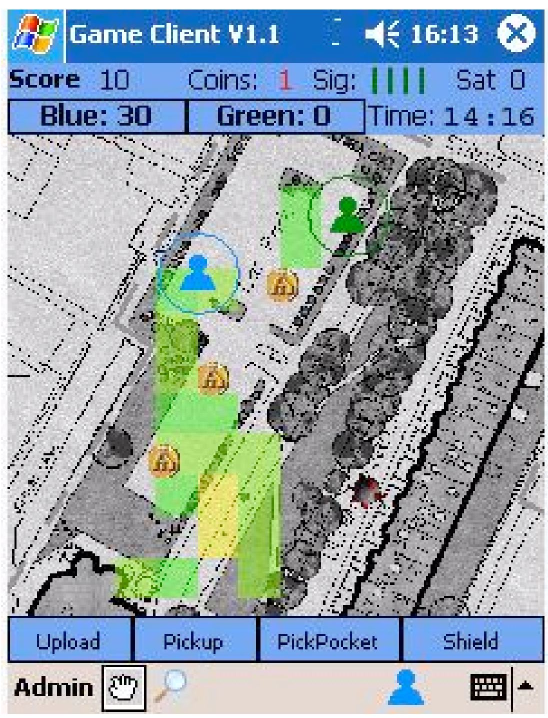


Figure 1: Bill's interface. The map shows players' locations, along with coins generally positioned outside the network. A map layer of green and yellow squares builds up as players move around, revealing network coverage.

The chances of a successful coin upload increase the deeper a player is inside wireless network coverage. However, a key competitive game feature is 'pickpocketing'. When players are close to each other they can use the *Pickpocket* button to steal coins that are being carried by other players. However, for a pickpocket to work, both players need to be within network range. Players can also protect themselves from such attacks by deploying a *Shield*, preventing other players from stealing coins that they have collected. Players bringing coins into network coverage have to be aware of where their opponents are, keeping a distance so as to upload coins before they get stolen.

To be successful in the game, players must therefore learn which areas are covered by wireless network and which are not. In other words, they have to learn and use the 'seams' of the infrastructure over which the game is played. Of course, tactics are used to avoid pickpockets, such as looking for an empty area of wireless coverage and then sprinting into it to do the upload. When attacked by a pickpocket, one might even cover the PDA's GPS antenna with one's hand so as to prevent location updates—but this is a risky strategy as location data might get through nevertheless. The latter strategy might be considered cheating, but we would have been pleased to see such expert manipulation had we ever seen it occur in game play.

In addition to coins, mines are occasionally placed in random locations on the map. When a player walks over a mine their PDA vibrates and is disabled for sixty seconds. This causes the player to drop all coins he or she was carrying and prevents the player from playing or observing what is going on in the game via the game interface. A player who has walked on a mine can still see other players, and can get into a good position to wait for their PDA to become active again.

In our development of the game we have trialed two distinct versions. The first version was a two–player game in which one player competed 'head to head' against another player. Feedback from this early pilot led us to increase the team size from one to two, and to add more features for social interaction between players. We maintained the essential competition between players in different teams, but we also added a feature involving collaboration between players in the same team: *collaborative uploads*. If teammates upload their coins to the same access point within two seconds of each other, they gain double the normal number of points.

In the first version of the game, players' PDAs collected data on the distribution of wi–fi signal strength as they moved around. At the end of the game, the server made a map from the data collected from all the players, which could be shown to players after their game. Forcing each player to discover network coverage individually, and to build up their own mental model of coverage without system assistance, was frustrating to players. In the second version of the game we made this data available during game play. A wireless coverage map was constructed dynamically by the server, and was regularly broadcast as part of the game state for display as a semi–transparent map layer on users' PDAs. Green squares show areas of high sampled signal strength, and yellow squares show areas of weak coverage. These maps provide players with additional awareness of the network strength in the game environment, and also reveal where players have been. These collaboratively constructed maps can be used to decide places to upload coins and areas where pickpocketing should work.

4. ARCHITECTURE & DEPLOYMENT

Our game was initially designed for play over our own wireless network set up at the University of Glasgow's campus. The Bill network consisted of five VPN-connected fixed wireless access points, arranged to create three distinct areas of network coverage. The map in Figure 1 shows part of this area. We also trialled the team version of Bill in the course of demonstrations at the MobileHCI 2004 and Ubicomp 2004 conferences, and the user trial findings reported in the next section of this paper stem from these trials.

At Mobile HCI we set up our own temporary wireless network that spanned much of a park–like area outside the conference venue. At Ubicomp we made use of the network set up by the conference organisers, one of whom kindly set up an access point in a window of the conference centre. The game was played in an area of grass and trees just outside this window. Each player in the game was provided with an HP 5550 iPAQ with built–in 802.11b wireless and a compact flash GPS unit. The game client itself was written in C# and makes use of the .Net Compact Framework.

Designing a system that would work reliably across the 'seams' of the wireless network proved challenging. We needed an appropriate networking system that could handle disconnection and reconnection, and would also work in areas of patchy signal strength. To this end, we developed both a custom wireless driver and messaging system.

The standard iPAQ PocketPC wireless driver automatically connects to the strongest network signal if it is not currently connected to a network. Unfortunately, this meant that the iPAQ would connect to non-game wireless networks when outside the range of the game network. Our replacement wireless driver allowed us to lock the iPAQ onto one network, disconnecting and reconnecting to that one network as needed without user intervention.

On top of this we implemented a simple UDP based messaging system. Since connections to the network are constantly being made and broken, TCP would have added considerable setup overhead. UDP allows us to use a 'one shot' messaging system without a guaranteed transaction mechanism. A central server periodically broadcasts notifications of all new game events (new coins and mines, scores, player positions), to all clients who are within range. Player's PDAs in turn send information over UDP to the server about their position and player events, such as coin pickups and uploads. While this introduced a time overhead in that events were only periodically updated, and only scales to a moderate number of PDAs, it removed the need for guaranteed messages.

Due to the limits in wireless networks, UDP packets are often (or usually) not received by all devices. The game was designed to work with these technical problems and drop outs, and display them as features of the game. So, for example, if one is outside network range then a user will not be given updates on other player's positions. This could be seen as a disadvantage to our system; but instead we have designed this as a game feature. For example, one can ambush other players by 'hiding' out of network range and then running into the network to pickpocket them .

5. TRIALS

This section reports on three sets of game trials. The first set of trials used our specifically installed network, set up around the university campus, and the first version of the game, with individual players rather than multiplayer teams. Ten computer science students were recruited as pairs of friends who competed against each other. Eight of the players were male, and two were female. After an introduction to the game and the PDA interface, players were taken outside and played the game against each other for 25 minutes. Players were then given a 10 minute debriefing interview, and a small gift was awarded to the highest scoring player.

As mentioned earlier, we used the results from this trial to redesign the game, in particular to encourage more social interaction between players. This second version of the system was run with six trials at the Ubicomp 2004 conference in Nottingham, and three trials at the MobileHCI 2004 conference in Glasgow. Each of these games took place outdoors in small parks outside the conference venues. Participants were recruited from the conference attendees, with games lasting a shorter fifteen minutes. Each of these games had four players in each game, involving 24 males and 8 females altogether. In total across the Glasgow and Nottingham games we ran 14 trial games involving 46 players overall.

Our analysis of these trials draws on three sets of data. All three sets of trials were videotaped, with different players followed and videotaped during their play. Second, during the games themselves, we made observations of players' actions and their interactions during the game. Lastly, we collected extensive logs of game events such as player's scores, attempts and successes in picking up and uploading coins, and in collaborative and competitive events such as joint uploads and pickpockets. We also conducted post–game interviews.

Our overall impression of the game was that it was broadly successful. Players reported that they found the experience of playing the game enjoyable and engaging, to the point of returning from playing the game physically exhausted. As one trial participant commented: "That was fun […] least I don't need to go to the gym now".

In our analysis we focused on three main aspects of play—support for collaboration and competition between players, interaction between the game and the environment and lastly the recording and replay of games.

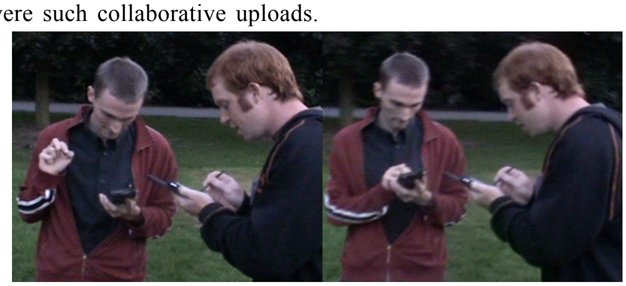
Figure 2: Two opponents out looking for coins and for network coverage in the first version of the game, walking side–by–side but with little or no social interaction.

5.1 Collaboration and competition

In the first set of trials, while the trials participants reported an enjoyable game, from our analysis and observations we could see that there was very limited social interaction between players. Indeed, in the post–game interviews a number of players commented that Bill could have been played as a single player game: there was hardly any interaction between players. At times players almost bumped into each other as they concentrated on playing the game and their iPAQs. Figure 2 shows one example of two players walking down the street side–by–side but almost completely ignoring each other, interacting instead with their PDAs. Without apparent network coverage, or information on each other's coins, they did not interact. While this shows something of the engrossment that the game could generate, we saw this as a generally negative feature of this game version, since interaction with others is so often a valuable part of the enjoyment of electronic as well as traditional games [7].

The main forum for social interaction in this version of the game was competition: the game was designed so that players would compete for a higher score. While there was evidence that players did put effort into competing for a higher score, this generated little social interaction during the game. The only specific game feature that required physical proximity between players was the Pickpocket, which allowed a player to steal coins from another. While players attempted to pickpocket a considerable number of times—in the first trial six times per player per trial, on average—there was only *one* successful pickpocket in the whole complete first trial. It appeared that there was insufficient support for awareness of the network coverage.

To address this, in the second version of the game we introduced the collaborative production and sharing of a network coverage map. We also changed the game to involve two teams of two players each, with a joint team score, and we added collaborative uploading of coins between members of the same team to achieve higher points. The second set of trials certainly shows more evidence of collaborative events, such as pickpockets, than trial one. Successful pickpockets jumped from negligible to an average of 1.6 per player per game, and there was considerably more interaction between players recorded in our videos and observations. Collaborative uploading—only available in the second set of trials—proved to be a focus for interaction, in that players would call each other over to both attempt to upload in the same spot, and verbally synchronise and confirm the success of the upload (e.g. Figure 3). Indeed, nearly half of all successful uploads were such collaborative uploads.



```
Player1: Right, OK, I have got six now=
Player2: Right, OK, are you ready=
Player1: 1,2,3 [5 second pause]
Player1: I got 120 points
```

Figure 3: Verbal interaction around a collaborative upload helped ensure that players would gain double points.

One complicating aspect of the change between trial one and trial two was that along with the changes to the system we also doubled the number of players in each game. This certainly had the effect of increasing the number of potential interactions between players, and in turn the amount of collaborative play generated. The reconfiguration of the game was therefore also a factor in the increased collaboration along with the changes to the system itself.

More broadly, these experiences show some of the value in supporting collaboration in outdoor games as well as competition. Our move to team games, and implementation of new features such as collaborative uploading, produced a second version of the game with more social interaction. While game designs often emphasise the importance of competition between players, our experience suggests the value of also supporting collaboration between players. Indeed, our game even supported a limited form of collaboration between competing players, through the way in which the shared map of wireless network coverage was built up during play.

5.2 Interactions between the game and the urban environment

Bill was trialled outdoors in three different environments: a section of a university campus, a park at the side of a university conference center and a city park landscaped on a steep slope. The trials were set in different landscapes, both in terms of the area covered by wireless network and in the topography of the local environment, and the local setting did have considerable effects on gameplay, e.g. the different environments had different configurations of potential visibility of players from each other. In the two later games, greater visibility contributed to more pickpocketing between players. The game areas in the first and last trials also had many more obstacles to moving around. In the first trial these took the form of cars and walls, and in the last trial hedges, bushes and trees covered part of the playing area. These obstacles changed the pace of the game in that they necessitated players walking larger distances to get coins.

It is common that the designer of an outdoor game has only limited control of specific environment in which the game is played. Indeed, part of the pleasure of an outdoor game is its setting within a less controlled environment or within an environment used for other purposes than just this game. Nevertheless, our findings underline the fact that the physical environment has an important influence on the play and experience of a game. The game experience in outdoor games is as much about the environment as it is about the computer game itself. This has led us to pay increasing attention to the environment in our design ideas for future games, to the extent of designing specific games for specific environments.

In the game players had to interact with aspects of the digital system of the game, such as network strength, coins and mines, as well as features of the environment around them such as cars, other players, trees and steep slopes. Indeed, one concern we had over the ethics of the first trial—and a common topic of questions from people we have discussed Bill with—was the possibility of players getting run over, even though most of the campus is pedestrianised. However, while players do spend considerable amounts of time focusing on the display on their PDA, they do not ignore or become unaware of the environment around them. Figure 4 shows a typical example.

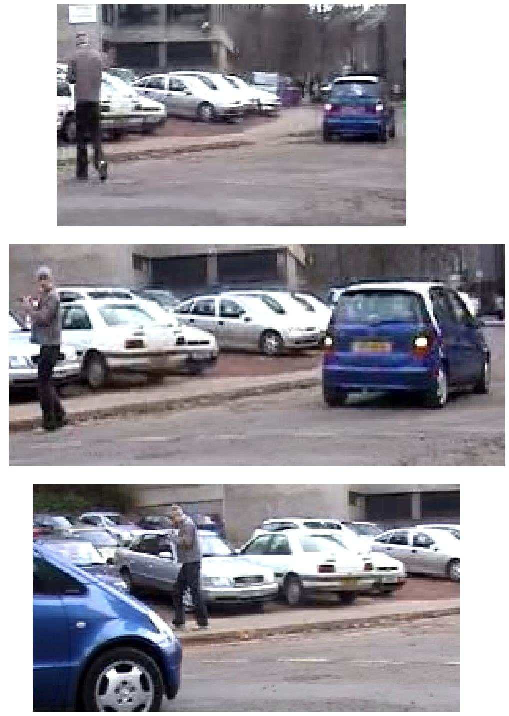


Figure 4: A player crossing a road while apparently engrossed in his PDA (top) is nevertheless aware of a car reversing behind him. He steps out of the way for a moment (middle) and then returns to checking where there is a strong enough signal to upload his coins (bottom).

A player crossing a road seemed to be engrossed in his PDA, but was aware of a car reversing behind him He moved and then looked to check it was past before going on to look for a good area to upload his coins.

One key skill in playing the game is simultaneously managing and relating interactions with the environment and with the PDA display, associating the objects presented on the PDA with features in the environment. Landmarks on the map have to be read and used in order to find where coins are, to find where teammates and opponents are, and to understand where one could successfully upload or steal coins. The world beyond the game is thus not ignored but rather is coupled to online information in play. Players did not, on playing our game, lose the ability to cross the road or walk around safely. Although they did spend considerable time concentrating on the information provided on the PDA, they also spent much time looking around, for other players, cars and landmarks to help them use the map.

One of the key elements of the game that we wished to understand was how players would interpret and use the distribution and variation of wireless network coverage. Although it is inevitable that there are many aspects of infrastructure that we did not intentionally reveal or use in our design, we did make a design resource of wi-fi network seams and so were interested to observe this aspect of game play.

All the Bill players played in ways that demonstrated a practical or tacit understanding of this particular kind of seam. The boundary between coverage and non–coverage acted as a divide in the game environment between where coins could and could not be uploaded, and where pickpockets could and could not happen. In general, players knew where they could and could not make these actions. However, this seam was dealt with in different ways. To some players there was only one location where they could upload, and they returned to that spot again and again. To other players, the signal strength meter was used to slowly 'scope' an area, to find the point of strongest signal to upload their coins. The boundary itself, an area of marginal or intermittent network connectivity that could be objectively measures, was not in itself a key feature of game play. Instead, what was significant was one's subjective understanding of the patterns of where one could reliably carry out or avoid particular game actions, based on one's experience and observations as a player.

5.3 Recording and replaying the game trials

We have found that mobile multiplayer games can be difficult to evaluate and analyse. Collecting and combining material from many moving players can be difficult. In response to this, we have focused on combining logs from multiple sources as well as video from several cameras, so as to create a coherent and synchronised visualisation or 'replay' of the game.

In the Bill trials, and in trials of other of our ubicomp systems, several observers each used a video camera to record system use. Of course, this only gave partial information on game play. The Bill software therefore had code added to log its use, so that each computer created and stored timestamped logs for each player. Generally, roughly accurate synchronisation of the computers in such trials is not a problem, as clock signals can be sent across the wireless network.

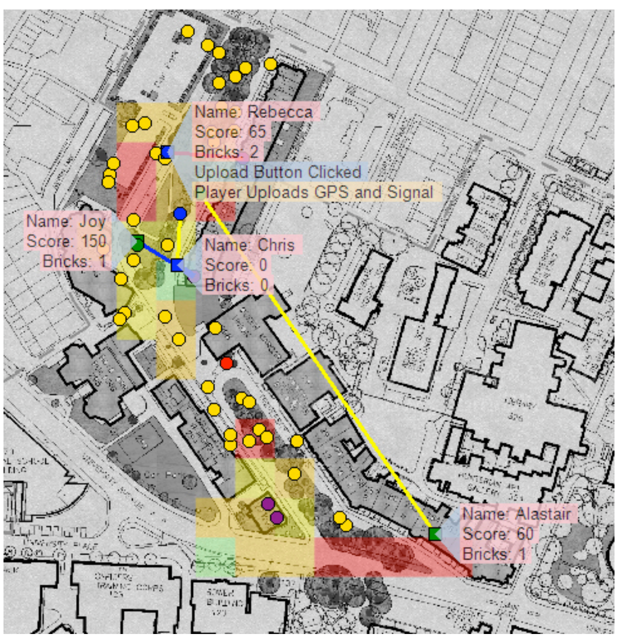


Figure 5. Replayer combines log information from players' PDAs and the game server log, and presents selected features of the movement and computer use of each player. It can be used in a synchronised playback along with video of the trial. Yellow lines highlight the difference between their current GPS positions and the positions last received by the server, with Alastair's positions showing an especially large disparity.

However, it can be more difficult to synchronise simultaneously recorded video (and audio) in order to carry out detailed analysis and transcription. Manually collecting and collating the recorded material from such trials is complex and error–prone. To help in this regard, we developed a tool, QCCI (Quickie), which runs on a mobile computer carried by each trial observer i.e. each observer has a mobile computer as well as a digital video camera. During the trial, QCCI logs the ongoing location of the observer via GPS, for use in subsequent replays. It also allows audio notes to be recorded, automatically labelled with the time and place of their creation—again for use in replays.

QCCI uses a low-tech method of synchronising trial videos and stills. It displays the network–synchronised system clock of the mobile computer in a large clear font. The observer then videos the screen of the computer at the start of the trial. This time can later be compared with the video camera's clock as shown on the video image itself, and the offset found. Audio synchronisation is similarly handled: QCCI plays a sound which can be recorded, and logs the time at which it is played, according to its own synchronised clock. This sound can then be used to correctly place the audio track in the replay.

Even the programmers' task of adding logging to their code can be error-prone. Replayer allows easy instrumentation of code from within the Microsoft Visual Studio .NET integrated development environment. Instrumentation can be directly applied to an individual program variable, method, class or solution with a single click, creating commands to record log data in each case. When the program is run, a timestamped and consistently structured log entry will be created each time any instrumented part of the program is used. If the programmer desires, it can be modified to suit specific situations because the instrumentation is inserted visibly into the code. This also allows meaningful messages to be added to the output, making the logs more readable to those less familiar with the code. Also, a toolset is also provided to allow remote FTP collection and organisation of logs, i.e. at the end of the trial, the log from each trial participant's computer is automatically collected and filed in a central repository.

After a trial, Replayer combines the logs gathered from participants' and from observers' computers. It then forms a unified visualisation or replay, to aid post hoc analysis. It presents a graphical user interface centred on a visualisation that can be played, paused and skipped through, much like a VCR, or queried like a database to locate specific events. A set of generic visualisation settings is offered, such as labels, lines and coloured regions to show logged information. Selected log events can be chosen via a simple graphical user interface for display, so as create an informative visualisation.

Replayer can export the visualisation as a movie file or a still image, for examination alongside trial video materials. It can also import video files, and play them in synchrony with the visualisation. Such combined visualisations can be used as a means of confirming hypotheses, searching for specific events, or establishing the system's state outwith the scope of the video. Other visualisations such as spatial distributions of sampled data are available, and statistical data can be easily specified and displayed by the system.

The example image shown in Figure 5 (above) shows a number of game features, including the logged state of key program variables and GPS locations for the four players *Rebecca, Joy, Chris* and *Alastair*. It has been set to use a yellow line to highlight, for each player, the difference between his or her current GPS position and the last position the server received for him or her. In the example shown in Figure 5, we see a clear example of a large disparity between these two positions. Alastair's current position is bottom right on the map, but the last recorded position was in an area in the top left. This meant that his position shown to the other players was far from his current position, which may have be a good or bad thing for a player depending on the state of the game and the player's awareness of how to take advantage of such inconsistency. Finer detail of the visualisation showed that some packets were getting through from his PDA to a nearby wireless access point, or from the AP to his PDA, but not enough to update the server.

We have found these tools to be very helpful in evaluating and redesigning Bill. We suggest that similar tools may be useful for designers and evaluators of mobile games and, more generally, of ubicomp systems. Much of the detail of the previous subsections stems from their use, and Replayer also revealed interesting features of the technology in use, for example the extent to which rain, snow and leaves on trees strongly affect 802.11 network coverage and GPS reception. Even the angle of a PDA with regard to the player's body affects its performance. Metal near to 802.11 access points also varies the distribution of RF, e.g. a truck parked in front of an access point was found to radically inhibit its coverage.

One system feature that we found particularly surprising was communication *gray zones*. These are created by the differences between the antennae of access points and mobile devices, and by the different bit rates used for broadcasts and for data transmissions. Transfer of packets to and from access points can show significant asymmetry, and high packet loss can occur despite apparent network reachability. Broadcasts are used to establish contact with a wireless network, and to set up data transmission. However, broadcasts can be done in the face of higher error rates than data transmissions, and so it is common to find that a mobile computer reports the availability of a wireless network that it is too far away from to use for data transmission.

At present, we are refining and extending Replayer so as to better integrate it into Visual Studio. We are also beginning to use it in the course of some new user trials, each of which consists of multiple Bill games played by the same players. We are exploring whether or how showing players replays of their past games can support the development of their techniques or tactics. In papers and presentations, we are using historical data from players' PDAs, the game server and video to show that the system and the users did what we said they did. In effect, we use these visualisations to testing and redesigning the system and refining its design. In future, we suggest that such visualisations may become a central part of the game itself, so that players use the same historical data to swap stories, see video and maps of what they did, showing they did what they say they did, and to test and redesign their *use* of the system.

6. DISCUSSION

The concept of seams has proven useful to us as a design concept, i.e. as a way of generating new design ideas for games. We would not argue that, in general, seamful design could or should replace more traditional seamless design. Instead, we suggest that the seams in our technical systems may be useful as resources for creating new design ideas and concepts. The notion of seamful design led us to emphasise features of our systems that we would otherwise have ignored. In particular, in the context of a game it led us to reveal to players a feature of a system that would normally be hidden: 802.11 signal strength. Making it a central feature of the game forced players to view and use signal strength in the game.

However, we consider that concepts such as seamful design are less important for players than for designers. Players can develop an understanding of how to play a 'seamful game' without entering into the philosophical analysis of design concepts, the physics of wireless networks, or the protocols of UDP. The way that the game reveals an aspect of ubicomp infrastructure is not meant to be part of a technical education, even though the technology of the game may be an element of its appeal.

In our ongoing work, we continue to develop games based on new seamful ideas. We are motivated by both technical and design issues. Firstly, we wish to explore larger game areas than we can cover with one wireless network, and games involving larger numbers of players than we could handle in Bill. Secondly, we are exploring the use of peer–to–peer ad hoc networks, which allow direct communication between players' PDAs as well as communication through fixed networks to central game servers. Peer–to–peer short–range wireless connections can often transfer information far more quickly than broadband connections, and mobile devices such as PDAs, music players and phones are appearing that have sizeable hard discs. This means that large amounts of information may be stored and shared among players, although the avoidance of a central server means there is a chance of inconsistency of information among different players'. This is usually considered to be a technical problem to be overcome but, in a game, keeping secrets or giving false information to others can be useful. We wish to explore ways in which designers can take advantage of such inconsistency, should they wish.

We are now working on a game called *Feeding Yoshii*, in which we aim to take advantage of inconsistent state among mobile devices and also the variability of wi-fi networks (as was explored in Bill). Players discover and use net–connected wireless access points, which the game presents as the home of a Yoshi, an animal that likes certain fruit. Other wireless access points are presented as 'plantations', the sources of fruit to feed Yoshis. Players can also trade or steal fruit from each other, using ad hoc networks set up between players' PDAs. At the moment, Feeding Yoshii is a simple hunting and trading game, and we now starting user trials and 'playtesting'. We hope that lessons from these trials and from the trials of other seamful games such as Bill, and new technical developments and collaborations will help us up advance our systems, interaction designs and our design concepts.

7. CONCLUSION

Bill was a successful attempt to apply the concept of seamful systems to an urban ubicomp game. Infrastructure becomes a central feature of the game, rather than the peripheral technical context. The deliberate exposing of selected aspects of the infrastructure suggests something of how users could develop their own ways to take advantage of the limits, gaps and seams in technology.

The trials of Bill show it to be both an enjoyable and provoking game for players. In particular, our redesign of the game had some success in encouraging more social interaction—competition and collaboration—between players. Although there was concern about the interaction between the game and the urban environment, in particular about whether players would be too distracted from cars, roads, walls and so forth, we found this not to be the case. We suggest that this may be due to the way that the game already involves continual comparison and relating between features of the game interface on the PDA and features of the wider environment. This quality of such interrelation may in fact be one of the defining features of ubicomp games, since the design area of ubicomp centres on ways to richly interweave digital media with the other older media that make up our everyday environment.

Lastly, we found the evaluation of Bill to be a challenge, initially. We were aided by tools such as Replayer, which combines logs from players' PDAs, the game server, evaluators' PDAs and video cameras. Replayer's facilities for playback and analysis have been invaluable in our redesign of the game system and in our evaluation of its use. We also point out the potential value of using such replays as part of the user experience, i.e. of the game itself, in order to support players in understanding their own and others' play, and developing new ways of playing.

Making seamful systems can be a challenge. Indeed, much of our implementation work involved dealing with the ways in which systems are generally designed to fail on encountering seams, rather than to continue and to communicate those seams to users. Most systems (e.g. TCP) either succeed or fail when used on the boundary. However, as Bill shows, these boundaries can have value as positive system features, rather than simply as points of failure. Again, we emphasise that we do not see seamlessness as always bad and seamfulness as always good. Seams shown in an interface have to be chosen and designed well, and our experience of Bill has been that it opens up an area of design that so far has not been intensively explored. While design techniques and lessons from such research may be applicable to areas beyond games, we intend to continue using mobile games both as a design area to work in, understand and contribute to, and as a vehicle for developing new technologies, tools for analysis and design concepts.

8. 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 Steve Benford, Geraldine Fitzpatrick, Bill Gaver, Areti Galani, Ian MacColl, Tom Rodden, Paul Rudman and Anthony Steed.

9. REFERENCES

[1] Björk, S., Falk, J.*, et al.* Pirates! - Using the Physical World as a Game Board. in *Proceedings of Interact 2001*, Tokyo, Japan, 2001, 9-13.

[2] Chalmers, M. and Galani, A. Seamful Interweaving: Heterogeneity in the Theory and Design of Interactive Systems. *Proc. ACM DIS*, 2004.

[3] Cliff, D. and Grand, S. The Creatures Global Digital Ecosystem. *Artificial Life*, *5*. 77-94.

[4] Dyck, J., Pinelle, D.*, et al.* Learning from Games: HCI Design Innovations in Entertainment Software. *Proc. Graphics Interface*, 2003.

[5] Falk, J., Ljungstrand, P.*, et al.*, Pirates: Proximity-Triggered Interaction in a Multi-Player Game. *Proc. ACM CHI*, 2001.

[6] Flintham, M., Anastasi, R.*, et al.* Where On-Line Meets On-the-Streets: Experiences with Mobile Mixed Reality Games. *Proc. ACM CHI*, 2003, 569-576.

[7] Lundgren, S. Joining Bits and Pieces - How to Make Entirely New Board Games Using Embedded Computer Technology, *Computing Science*, IT University of Goteborg, Goteborg, 2002.

[8] Rutter, J. and Bryce, J. Editorial Comment. *Information, Communication and Society*, *6* (4). 5-11.

[9] Salen, K. and Zimmerman, E. *Rules of Play: Game Design Fundamentals*. MIT Press, 2004.

[10] Schneider, J. and Kortuem, G., How to Host a Pervasive Game: Supporting Face–to–Face Interactions in Live-Action Roleplaying. *UbiComp workshop on Designing Ubiquitous Computing Games, 2*001, 6.

[11] Weiser, M. Building Invisible Interfaces. Keynote talk, *Proc. ACM UIST*, 1994.