Mobslinger: The Fastest Mobile in the West

Helen Clemson
Informatics
Infolab21
Lancaster University
Lancaster, LA1 4WA, UK
h.clemson@lancaster.ac.uk

Paul Coulton
Informatics
Infolab21
Lancaster University
Lancaster, LA1 4WA, UK
p.coulton@lancaster.ac.uk

Reuben Edwards
Informatics
Infolab21
Lancaster University
Lancaster, LA1 4WA, UK
r.edwards@lancaster.ac.uk

Fadi Chehimi
Informatics
Infolab21
Lancaster University
Lancaster, LA1 4WA, UK
f.chehimi@lancaster.ac.uk

FNG2006, June 26–28, 2006, Preston, England.
ACM 1-xxxxxxxxxxxxxxxxxx.

Abstract

Whilst there is a number of location sensing games emerging for mobile phones, from both commercial and academic sectors, there are few examples of social proximity based games that are effectively position independent. Bluetooth would seem an obvious choice for proximity based games, although the majority of games produced to-date simply uses it to provide a quasi peer to peer connection between users of multiplayer games. This is no-doubt due to the fact that proximity can often be implied from other location sensing technologies and that Bluetooth is often perceived as difficult to employ. In this paper we will show that Bluetooth can provide exciting game scenarios that can enable spontaneous stimulated social interaction using only proximity information. We illustrate this through the design rationale and subsequent implementation of 'mobslinger' which is a wild west, quick draw, 'shoot-em-up' game using mobile phones.

Keywords

mobile, multiplayer, game design, social interaction, proximity sensing, Bluetooth collaborative

ACM Classification Keywords

J. Computer Applications, J.7 Computers in Other Systems-Consumer.

Introduction

With mobile phone subscriptions growing from just over 11million in 1990 to approximately 2 billion today there can be little doubt that mobile phones are becoming part of the fabric of our daily lives. However, the majority of phone users are still voice centric and the relatively slow adoption of data services means that manufacturers and operators are increasingly trying to identify application and services that excite the user. Mobile Games are already a success story with revenues exceeding € 1.7 billion in 2005, although, traditional games publishers have displayed a relative weakness in leveraging their success from the console market [7]. This is no doubt due to the fact that games on mobile phones appeal to a much wider consumer demographic, who have shown little interest in the driving and first person shoot-em-ups titles that dominate the console market, and have more in common with those utilizing games online or through their TV[2]. This opens the door to games that utilize the benefits of the mobile phone platform such as location based games, which despite initial skepticism, are increasingly being seen as an important new genre. We have already seen a number of location based games for mobile phones [6] and the number is increasing all the time. Some of these games use proximity, in the sense that the user's location is close to either another player or a real or virtual artifact within in the game. However, none rely solely upon the proximity between either a player and another player, or the player and a game artifact, irrespective of the physical location. The most notable example of this type of game play, although not mobile phone based, is that of Pirates [3] which was an adventure game using Personal Digital Assistants and RF proximity detection. Players completed piratical missions by interacting with other players and game artifacts using RF proximity detection. One of the interesting aspects that emerged was the stimulated and spontaneous social interaction between the players. One of the research motivators for this project was to produce a game that provided the opportunity to bring this type of spontaneous social interaction to mobile phone users. We believe it is best achieved by removing the requirement for a central game server, as utilized in Pirates, and utilize a proximity detection scheme that initiates a dynamic peer to peer connection.

In terms of proximity detection the obvious choice is Bluetooth which despite previous predictions of its demise is in fact increasing its growth, with Nokia predicting a year-on-year increase of 65% in 2006. In fact there are already a small number of mobile Bluetooth proximity applications which are often described as Mobile Social Software (MoSoSo) and can be viewed as evolutions of Bluejacking. This is a phenomenon where people exploit the contacts feature on their mobile phone to send messages to other Bluetooth enabled devices in their proximity [5]. Bluejacking evolved into dedicated software applications such as *Mobiluck* and Nokia Sensor which provided a simpler interface, and in the case of Nokia Sensor, individual profiles that could used to initiate a social introduction. The only gaming example of this type is *You-Know-Who* from the University of Plymouth which provides a simple game premise to help initiate a meeting. After scanning for other users running the application and 'inviting ' a person to play the game, the first player acts as a 'mystery person', who then provides clues about their appearance to the second player, who builds up a picture on their mobile phone screen. After a set number of clues have been given, the players' phones alert, revealing both players' locations and identities. Obviously, the game play is quite limited and effectively non-competitive, which is unlikely to result in repeated game play, therefore, it is closer to the other MoSoSo applications than a game.

The game described in this paper provides significant opportunities for both stimulated and spontaneous social interaction with additive and competitive game play. The game draws from the familiar, which is always a good way of gaining acceptance for a new game, using the concept of a Wild West quick-draw gunfighter which we have called 'Mobslinger', as shown in figure 1.

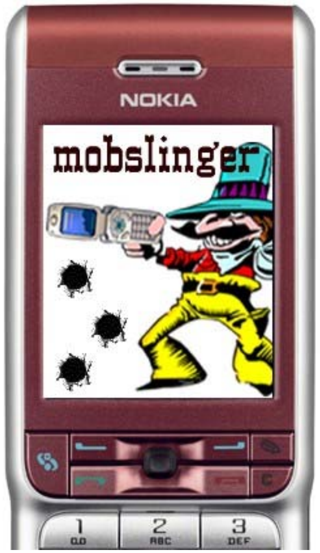

figure 1. Mobslinger splash screen on a Nokia 3230

In the forthcoming sections, we shall explore the design and implementation of the game, provide a rationale for the choices made and highlight the design challenges overcome. Finally, we shall define future enhancements that will facilitate large scale user experiences to be evaluated before drawing our overall conclusions.

Mobslinger Game Play

The basic game premise is simple to understand and operate, which is an essential feature in any game [1]. The relatively low cost of mobile phone games means they are often very quickly discarded by users if they cannot quickly engage with game-play. Mobslinger runs as a background application on Symbian Series 60 smartphone which periodically scans for other users in the vicinity who are also running the mobslinger application. Once detected, a countdown timer is initiated on both phones which alerts the user by sounding an alarm and vibrating the phone. The user then has to 'draw' their mobile and enter the randomly generated number which has appeared on the screen as quickly as possible. The person with the fastest time is the winner and the loser is 'killed', which means their application is locked out from game-play for a set period of time. The game is playable in a number of different modes which we will discuss in the following paragraphs both in terms of the operation and their relative merits.

Quick Draw

This is the basic mode of the game for two mobslingers at a time. Once the phones have detected each others presence the first mobslinger to 'draw' their mobile and enter the correct code wins, as shown in Figure 2. This mode is intended to promote spontaneous social interaction between two players, who would generally be unknown to each other, in the form of a ludic greeting. The loser is locked out of the game for a set period and unable to interact with other players. This may be a disadvantage for larger social groupings and therefore has been addressed in a subsequent mode. The optimum length of the lock out is something that can only truly be ascertained from the large scale user feedback and at present we have fixed this at 2 hours.

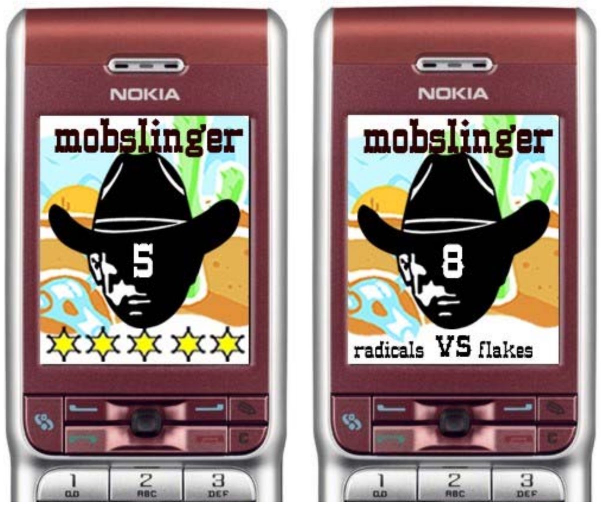

figure 2. Mobslinger 'Draw' screen and 'Draw' screen in Outlaws mode

Blood Bath

This is the large scale battle mode (often referred to by the design team as playground mode), in which two or more mobslingers score points by beating randomly selected targets to the draw over a set time period. After the time period has expired the game is ended and the mobslingers' high score is displayed on the phone. The mobslingers can then compare scores to ascertain the winner and allows them an element of *schadenfreude* (amusement at the misfortune of others). Because of the intensity of the game play the constant noise may be irritating in some social situations and this mode is probably best played outside or in a regimented social setting. This mode does not have the spontaneity of Quick Draw and needs to be initiated by a preformed social grouping.

Last Man Standing

This is a less intensive battle mode than Blood Bath and is for two or more mobslingers. The aim of the game is to be the 'last man standing' after a series of quick draw encounters within a group. Duels are triggered at randomly timed intervals, so are likely to prove less irritating for the general public, and create an air of anticipation amongst the mobslingers. The random time also means that the group can be engaged in the general social proceedings of the evening and the game effectively provides a humorous aside. The game play may be lengthy and players knocked out in the early stages may lose interest, although players in search of the more intense experience may opt for Blood Bath mode. As with Blood Bath mode, this requires an initiation by a preformed social grouping.

Outlaws

This is the team mode of the game where groups of mobslingers join forces to create 'Outlaw' gangs who then go out in the hope of combating other teams. The Outlaw gangs are generally comprised of two or more mobslingers and offer a mode of play where there is a greater opportunity for tactics to develop. The game play is similar to Last Man Standing, although the random interval between duels is shorter, and only mobslingers of an opposing set of Outlaws are selected to play against each other. In this mode, challenges are made and accepted between gangs. In figure two we show the 'Draw' screen for Outlaw mode where the player rating is replaced by the gang names. In essence this mode can be viewed as preformed social groups engaging in the spontaneous encounters of the Quick Draw mode.

Top Gun

We have also included a ranking system into the game to satisfy the desire of some experienced players being able to differentiate themselves from novices. The basic player starts out with a one star rating and this is increased in stages until they reach a 5 star rating as shown in figure 2. The levels are based upon games played, kills made and then their kill/die ratio. The levels are built up from satisfying a defined minimum for each of these criteria for each advancement level.

Mobslinger Implementation

In this section we will present a discussion of the implementation of mobslinger but rather than present a mere technical description we shall provide an overview of the design and the design challenges faced. Mobslinger differs from the vast majority of mobile applications in that it is a background application that, once activated, requires no user input. The application will continue executing in the background of the users normal phone activities until another phone, with mob-slinger running on it, is detected. Proximity detection is achieved using the Bluetooth discovery protocol to search for other Bluetooth devices in the near vicinity (approximately 10 meters). Although the proximity detection process sounds fairly simple from the game description, it is actually the most difficult part to implement. Bluetooth is based upon a client server architecture which means that it must also have a state where it allows itself to be discovered in addition to being the discoverer.

To achieve this operation the Bluetooth client server architecture has to be initiated, which involves setting one phone to discovery state (server) and the other to listening (client). One technical problem faced was how best to alternatively set the phones to each of the states, and determine the length of time in which the phone should remain in that state. The optimum length of time will be dependent upon the game mode. For example, in Quick Draw mode where you are looking for spontaneous interaction you will not want to operate the Bluetooth too often otherwise it could drain the mobile battery, alternatively, in Blood Bath mode where you are creating a social event and you will wish the events to occur more quickly. There is also a need for a random element to be included in the time between switching modes as a situation could occur where two phones keep missing each other.

Once a Bluetooth device is found, a socket is opened to search for the mobslinger game. This is known as device and service detection, which plays a fundamental part in Bluetooth communication [4]. When a service has been detected, a secure socket can be set up to transfer data between the two Bluetooth devices. The useful aspect of Bluetooth service detection is that it can be made application specific; in other words, we can isolate discovery to devices running mobslinger and ignore any other Bluetooth devices that may be in the vicinity. Once a secure connection has been established between the devices, the game can generate a random number that the user has to press, to 'shoot' the other person. This also activates a timer which starts recording when the random number is generated and is stopped when the user hits the correct key. The person with the fastest shoot-out time is the winner and is allowed to continue in the game, and in Blood Bath mode gain points. However, the loser dies and, as a consequence, is locked out of game play for a specified time, which is variable dependent upon the game mode. This operation is highlighted in the state diagram for the game shown in Figure 3.

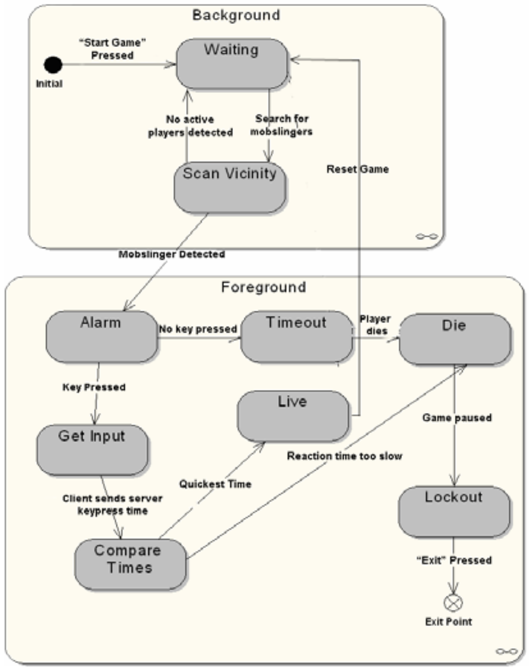

figure 3. Mobslinger State Diagram

The different modes of game play are implemented using the client/server Bluetooth architecture which is implemented with one client and one server even for the multiplayer modes of Blood Bath and Last Man Standing. Whilst it would be feasible to use one phone as the server and all other phones as clients, this piconet structure would limit game play to 8 players. In reality, we could only ever shoot one person at once, therefore, a single connection is only ever needed between two phones, thus allowing many pairs of phones to fight simultaneously.

The server controls the game play by requesting and sending data to the client devices. The Bluetooth interaction between the two devices can be seen in Figure 4. The game play only requires two messages (after detection) to be sent between the server and client. Firstly, the client sends a message to the server stating the time taken for the correct key to be pressed. The server then calculates the winner and returns a message with the result. The two devices are then disconnected to allow game play with other mobslingers in the area.

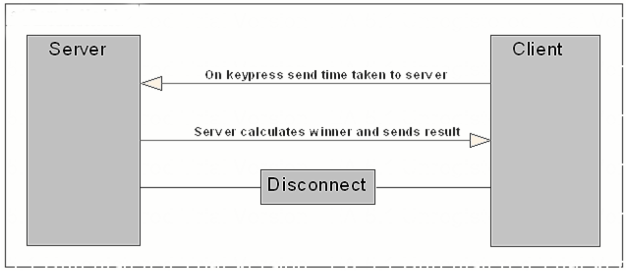
Server
Client
On keypress send time taken to server
Server calculates winner and sends result
Disconnect

figure 4. Mobslinger Bluetooth client/server transfer

User Interface

Mobslinger provides the user with more thrill and excitement than the average mobile application. Vibrating alerts and sounds are used to entice the user; such as gun shot sounds when the user clicks fire and vibrating alerts when a duel request is received. Graphics become more elaborate the higher the mobslinger progresses in the game, thus giving a ranking system and bragging rights to players with new shoot screens.

What next?

Whilst many mobile games become uninspiring for the developers once the excitement of the design process is complete, this game offers a whole new set of research opportunities. While the game is popular amongst our research group there are a number of different aspects related to each of the modes that can only be answered by trials involving significant numbers of users, indeed, for the game to work on a commercial level the more players running the application the better. We are therefore trying to expand the implementation of mobslingers to other mobile platforms such as UIQ, J2ME, Windows Mobile, and Python so that we can deploy the software across a wide a range of users as possible. Once completed we can then investigate the following research questions for each of the game modes:

Quick Draw

- is the game played predominantly as a 'meet and greet' device between friends or to create spontaneous social interaction with unknown adversaries?
- Will the spontaneous interaction initiated by the game lead to further social interaction or do players prefer a degree of anonymity?
- Would the inclusion of a chat capability for use after the match be seen as beneficial?
- Does the shoot-em-up style of game play put off any users groups?
- What is an optimal scan time such that it doesn't impinge on the general battery life of the phone but is sufficient for users to experience a reasonable number of interactions?
- What is an acceptable lock-out period for losers?

Bloodbath

- Is the game played in existing social groups or do new groups come together to play the game?
- How quickly do we need to set the scan period to ensure a fast action game?
- What is an acceptable lock out period that does not cause a loss of interest for the loser but provides an advantage for the winner?
- How long should the game last so that it doesn't become boring?
- In what type of environment is the game best played?

Last Man Standing

- Does this mode have any perceived advantages/disadvantages over Blood Bath mode?
- As it is essentially Quick Draw mode with a faster scan time, what is the optimal time?
- Is it preferable to keep this mode within a defined group or should other mobslingers, who may be in the

vicinity but operating in Quick Draw mode, be included in the action?

- How long should the game last and should lock-outs be cleared at the end of the game?

Outlaws

- Do groups come together to play as teams and do these encounters tend to be spontaneous or contrived?
- Would some form of team communication offer a useful aspect to this mode?
- Do any form of tactics develop, and if so, what form do they take?
- If no tactics are evident, how can the mode be altered to facilitate their development?

The number of research questions highlighted, and no doubt others will become apparent, illustrate the fact that social proximity orientated game play using mobile phones is an area where there is a dearth of experience and therefore enormous potential.

Conclusions

Whilst the mobslinger concept may appear simple, particularly as it does not require significant user input or display complex or flashy graphics, it does provide a uniquely mobile experience. This mobile experience is not merely in the mobility of the device itself, but also encompasses the social aspect that mobile phones are most often associated with in the minds of the general public.

Mobslinger, and other games which are sure to follow, will provide an insight into spontaneous collaborative mobile games for an audience whose primary focus is the social.

Acknowledgements

Our thanks to Nokia for the general provision of software and hardware to the authors' paper which was used as part of the implementation of this project.

References

[1] Bateman, C., and Boon, R., 21st Century Game Design, Charles River Media, September, 2005.

[2] BBC News, Games firms face challenges ahead Published: 2006/01/09 08:11:18 GMT http://news.bbc.co.uk/go/pr/fr/-/1/hi/technology/4587902.stm.

[3] Björk, S., Falk, J., Hansson, R., and Ljungstrand, Pirates! - Using the physical world as a game Board, In Proceedings of Interact 2001, IFIP TC.13 Conference on Human-Computer Interaction, Tokyo, 2000.

[4] Bray,J., and Sturman,C., Bluetooth: connect without cables, Prentice Hall Inc, Upper Saddle River, NJ 07458, 2001.

[5] Jamaluddin J., Zotou N., Edwards R., and Coulton P., Mobile Phone Vulnerabilities: A New Generation of Malware, IEEE International Symposium on Consumer Electronics, , Reading, UK,ISBN 0-7803-8527-6, ISCE_04_124 pp 1-4, 2004.

[6] Rashid, O., Mullins, I., Coulton, P., and Edwards, R., Extending Cyberspace: Location Based Games Using Cellular Phones, ACM Computers in Entertainment, Vol 4, Issue 1, 2006.

[7] Screen Digest press release for "Wireless Gaming Business Models: Opportunities for publishers, developers and network operators." 7th December 2005, http://www.screendigest.com/.