Author(s): Al-Zakwani, Abdullah
Title: Establishing a platform for networked games on mobile devices using SMS/GPRS
Year of publication: 2007
Citation: Al-Zakwani, A. (2007) 'Establishing a platform for networked games on mobile devices using SMS/GPRS' Proceedings of Advances in Computing and Technology, (AC&T) The School of Computing and Technology 2nd Annual Conference, University of East London, pp.128-134
Link to published version:
http://www.uel.ac.uk/act/proceedings/documents/ACT07.pdf

ESTABLISHING A PLATFORM FOR NETWORKED GAMES ON MOBILE DEVICES USING SMS/GPRS

Abdullah Al-Zakwani
A.Al-Zakwani@uel.ac.uk
University of East London

Abstract: The gaming industry has being exponentially expanding during the recent years. This expansion has seen game playing spreading from dedicated consoles to mobile communication devices. However, currently the main challenge is to establish two or more player games from handheld mobile devices over a network. This paper will define a possible way of achieving network games using Short Messaging Services (SMS) and/or GPRS. In this paper the outline of what Network Gaming is, how it is traditionally implemented (using desktop computers) and how it can be initiated into the mobile environment will be explored. The efficient data transfer over SMS or GPRS will require formatting and so it is compressed and made easily identifiable by the receiving device. The platform identified implemented and tested over a **Tic Tac Toe** game scenario.

1. Introduction to mobile games

During recent years one can see that there has been a surge in mobile communications. There has also been a focus on mobile software, mobile devices couple with an emphasis on quality of service and value for money by the companies providing the services (Le Grand, Ben-Othman and Horlait 2000). This era is being compared by many experts as the golden era of mobile computing similar to the desktop computer growth (Sydney 1999, Varshney and Vetter 2004, UPI 2005) in the late eighties and early nineties. With the growth of mobile devices, it is anticipated in the near future that almost every aspect of desktop computers will be ported onto mobile systems. Therefore there is a clear established demand for having the mechanism and achieving the performance we currently see on desktops for mobile systems. Thus major improvements are required to allow today's mobile devices to perform more complex functions and communication. In parallel to the trend described above, the gaming industry is witnessing a major shift towards network gaming (Van der Pool 2006) where players in multiple geographical locations could play against each other despite the distance. This movement is seen as integral to mobile devices especially as only few a specialised mobile devices (like the "Engage" by Nokia) can achieve this today. Currently, even a mere two-player mobile gaming capability is considered the maximum with a further constraint being the distance between players (which is usually **ten** meters and must be supported by Bluetooth communication) (Al-Zakwani 2006). Thus with the industry's plans to give mobile devices the ability of computers and the need of multiplayer-gaming capabilities it is essential that a technique is developed to facilitate a platform of network gaming on mobile devices.

2. The relation between console games and mobile games and their development

Games have arguably been in existence since the beginning of human life (Pelkonen 2004). This game existence phenomenon

can also be seen in the computing industry as computer games are arguably the starting point of the computer system as we know it today (Harnad 2006). However since the introduction of handheld telephones, much emphasis has been placed on introducing the console gaming capabilities into them. However the continuous game development has developed from static games where players have to be at the same geographical location to quite a formidable network structure where players from different geographical locations can play together (Quint and Shubik 2003).
Due to the steady popularity growth of mobile devices and applications and the resulting profit the industry has realised, gaming is seen to be the next important stage (Varshney and Vetter 2004, UPI 2005). Mobile applications and games have produced a profit of over a billion dollar 2004 for the industry and this is expected to double by the end of 2005 (Uhlir 2005). As mobile devices are small and usually able to support only a single person to interact with a single mobile it is important to establish a technique to allow more than one player to interact. This interaction can only be facilitated by network facilities like Bluetooth, GPRS, EDGE and other network communication capabilities. However this paper will focus on how an SMS can be used to initiate semi-dynamic two player games over a robust algorithm that can be ported onto any communication mechanism.

3. The game and platform

Digital communication is based entirely on transmitting binary data from one point to another (Sanneblad and Holmquist 2002). Algorithms are normally defined to facilitate the best data (as binaries) transportation mechanism according to the case at hand. For instance, playing a game over a broadband connection could allow the developer to initiate the moves by sending an image data over the network. However, due to the speed constraint a dial up connection have to communicate via strings that will need to be converted at the receiving end. This is the important issue during a game platform design.
In the mobile game platform the effects if these constraints are extreme as only very small data flow can be allowed over a network due to the cost and the processing capability of the sending and receiving devices.
In this paper we suggest a platform to initiate communication using characters which are based on a controlled number of binaries consisting of a very small number of bites. For minimising the transfer load the data is tokenised onto SMS which helps minimising the size.

This communication allows a stable controlled communication which can always be estimated for the process on a device and a flow over a network. To define and test this algorithm a game of *Tic Tac Toe* is developed using SMS to transport the binary (or characters) required to allow play to take place. The game play and platform architecture are defined bellow:

- The game play architecture: how the game play takes place
- The platform architecture: what objects are required to facilitate this gaming mechanism

4. The architecture

The game used for this experiment is Tic Tac Toe where players compete by filling a grid on nine boxes with the **symbols** O or X and the winner is determined by filling all the horizontal, vertical or diagonal boxes with his symbol. The architecture will be defined in two parts:

1. The game play architecture
2. The platform architecture

4.1. The Tic Tac Toe game play architecture

The game will be initiated by a request from player one who is responsible for opening the game Session. This message opens a specific communication port and keeps it open for listening to the response from player two. This port will be listening to a specific type of SMS (i.e. a data storage SMS that holds the required binaries (that form integers)). When this SMS arrives, the **Session** will recognise it according to its numerical structure. Then it will instantly process the current state of the images and manipulate the required image on the board. This mechanism is presented on Figure 4.1.1.

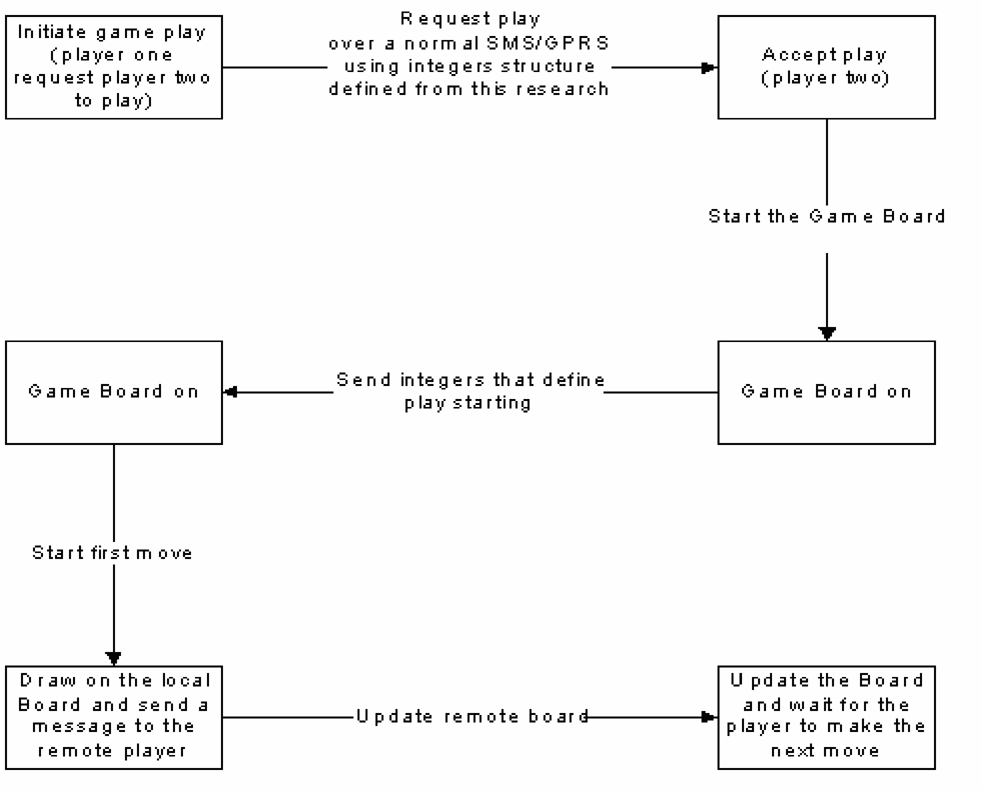

Figure 4.1.1, Game play initiation and communication mechanism for the game

This technique could be applied onto more complex games by tagging each move of the game with an integer value and thus process the animated images according to this value. This is the normal mechanism that takes place during game play but is at a lower level on the current common platforms. Thus the application of this architecture can be more defined and made more robust to accommodate more complex movement.

4.2. The platform architecture

The platform will consist of four important objects (see Figure 4.1.). These objects are the Session which define the game play and keep a constant communication channel that would keep the consistency of the game play, the MsgSender and MsgReceiver that would be responsible for the data communication and the Board that would be responsible for updating the game state by painting and repainting itself.

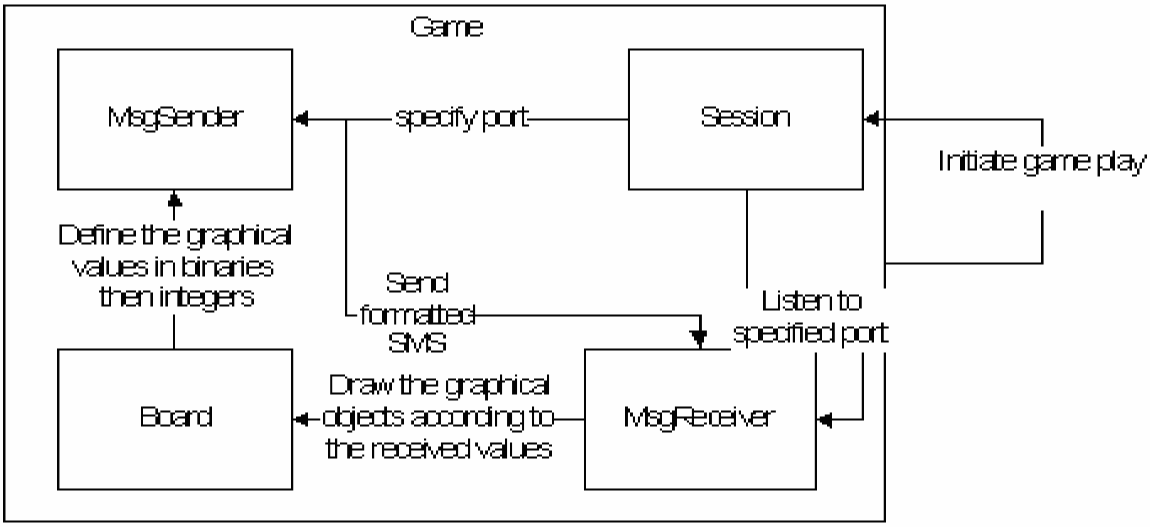

Figure 4.2.1, the game platform architectural design

The game play is commanded by the graphics display over the Board which is defined by the paint method. Bellow a piece of code that demonstrates the Board painting mechanism.
The game will start by initiating game play and create a Session that would determine the communications port. This will provide the function of identifying the information required for the game and ignore any SMS that has no connection to the game if received during play. The Session creates 2 players and defines the player to start (current player) during creation and a common port. The Session is the main play control which allows communication on the network.
This communication is achieved by MsgSender and MsgReceiver. The run method (on the sending side) determines communication mechanism (in this case MessegeConnection). The MessegeConnection object receives the token and sends it to the remote device listener (MsgReceiver) object.
Similarly the run method in the receiving device determines the connection like the sending object defined above. The only difference is that, this run function listens to the port number determined by the Session object and pick up any incoming message and translate so it can tokenise it. Due to this listening structure (able to listen to any type of message) this architecture is easy to use and adapt the mechanism can be implemented to other communication techniques like the GPRS and EDGE.

5. The experiment

The major issue in Network Gaming is the accuracy and speed of delivery. These two issues define the actual playability of the game. If one message does not arrive the game will not continue causing a crash. If the delivery is too slow it might cause the players (sending and receiving players) to assume they have been disconnected and so quit the game before it has ended.

```
public void run()
{
   try{
        conn = (MessageConnection)
Connector.open(url);
        TextMessage msg = (TextMessage)
conn.newMessage(conn.TEXT_MESSAGE);
        msg.setPayloadText(textToSend);
        conn.send(msg);
      }
      catch(Exception e){

        e.printStackTrace();
      }
      finally {
         if( conn != null ){
              try {
                   conn.close();
              } catch( Exception e ) {
              }
         }
      }
}
```

```
public void run()
{
   try{
         conn = (MessageConnection)
Connector.open(url);
         while(true)
          {
            TextMessage message = (TextMessage)
conn.receive();
            System.out.println("Handling message");
            handler.handle(message.getPayloadText(),
essage.getAddress());
            System.out.println("Message Handled");
          }
       }catch( Exception e ){
            e.printStackTrace()
       }
      finally {
          if( conn != null ){
            try { conn.close(); } catch( Exception e ){
               }
             }
       }
}//end of run...
```

Using a Sun Java Wireless Toolkit 2.4 for CLDC an experiment has been conducted to test the performance of the system (delivery consistency and speed). This was done using the presented architecture and the code defined on the previous sections. Seven moves games were played and the time taken for state of the game to change was randomly recorded. The table bellow shows the time.
This experiment was done by compiling the code into software that was run over 2 emulators (provided by Sun Microsystems which can simulate a network environment over generic software based mobile devices). These emulators could provide accurate testing for the delivery mechanism but do not have the function to simulate distance between devices and the nodes a massage will pass to arrive to the receiver from the sender. Hence, the result for delivery was 100% while the speed was hard to determine as the distance and the number of nodes on the network was constant.
For above reason the software was ported into two real devices for an uncontrolled experiment. The reason for the uncontrolled environment experiment is that the networks used have their own routing beyond this experiment’s control. This experiment used real O2 to O2 and O2 to T-Mobile telecommunication cells interchanges. The massages were recorded at random intervals.

Device 1 (on network 1)	Device 2 (on network 2)	Time taken
J2ME emulator (on PC)	J2ME emulator (on PC)	0.1
J2ME emulator (on PC)	J2ME emulator (on PC)	0.1
J2ME emulator (on PC)	J2ME emulator (on PC)	0.2
J2ME emulator (on PC)	J2ME emulator (on PC)	0.1
J2ME emulator (on PC)	J2ME emulator (on PC)	0.1
J2ME emulator (on PC)	J2ME emulator (on PC)	0.2
J2ME emulator (on PC)	J2ME emulator (on PC)	0.1

Table 5.1. Testing the delivery over Emulators running on PC (Windows XP Pofessional)

An interval is defined by one move from the sending to the receiving player (from a player drawing the X or O on the local Board and the change being implemented on the remote player Board). Table 5.2. shows the results of the communication.

Device 1 (on network 1)	Device 2 (on network 2)	Time taken
Nokia N70 (O2 in London, UK)	Nokia N70 (O2 in London, UK)	0.4
Nokia 6230 (O2 in London, UK)	Nokia N70 (O2 in London, UK)	0.8
Nokia 6230 (O2 in London, UK)	Nokia N70 (O2 in London, UK)	0.4
Nokia N70 (O2 in London, UK)	Nokia N70 (O2 in London, UK)	0.2
Nokia N70 (O2 in London, UK)	Nokia 6230 (O2 in London, UK)	0.3
Nokia 6230 (O2 in London, UK)	Nokia 6230 (O2 in London, UK)	0.5
Nokia 6230 (O2 in London, UK)	Nokia 6230 (O2 in London, UK)	0.6
Nokia N70(T-Mobile in London, UK)	Nokia N70 (O2 in London, UK)	0.6
Nokia 6230 (O2 London, UK)	Nokia N70 (T-Mobile in London, UK)	0.5
Nokia6230(T-Mobile in London, UK)	Nokia 6230 (O2 in London, UK)	0.4
Nokia N70(T-Mobile in London, UK)	Nokia 6230 (O2 in London, UK)	0.7
Nokia N70(T-Mobile in London, UK)	Nokia N70 (O2 in London, UK)	0.2
Nokia 6230 (O2 in London, UK)	Nokia N70 (T-Mobile in London, UK)	0.4
Nokia 6230 (O2 in London, UK)	Nokia6230(T-Mobile in London, UK)	0.9

Table 5.2, Testing the delivery over Nokia 6230/N70 over O2 and T-Mobile Networks

The main purpose for the experiment was to determine the maximum and minimum time taken for delivery. It was found to be too small to be noticed for semi-dynamic games. The tests showed that on average less than 0.2 seconds on the emulator and less than 0.5 seconds on the real devices (Nokia model 6230 and N70) proving the consistence and ability to exploit the technique for more complex games.

6. Conclusion

This experiment has been further tested on more complex devices (Nokia N91) to examine the consistence of the architecture. This architecture can be applied to WAP/GPRS/ADGE and all other similar network architecture environments as all these architecture environments transfer data as a stream of bytes (reference) which are eventually translated into their required form on the receiving device. It can be concluded that this architecture is reliable.

References

Quint, T., Shubik, M. (2003) On Local and Network Games, Cowles Foundation Discussion Paper No. 1414

Pelkonen, T. (2004) Mobile Games, Anticipating Content Technology Need an E-Content Reports by ACTeN

Uhlir, K. (2005) Industry's lack of understanding of the complexities of location technology is holding back LBG, Mobile Games 3: 8 by Informa Telecoms

Sanneblad, J., Holmquist, L. (2002) Using Ad Hoc Network Games to support Face-to-Face Interaction in Public Places, UIST DEMO-CAFETreck

Varshney, U., Vetter, R. (2004) Mobile Commerce: Framework, Applications and Networking Support. Mobile Networks and Applications, 7: 185-198. Springer, Netherlands.

Harnad, S. (2006) The Annotation Game: On Turing (1950) on Computing, Machinery, and Intelligence, in Epstein, Robert and Peters, Grace, Eds. The Turing Test Sourcebook: Philosophical and Methodological Issues in the Quest for the Thinking Computer. Kluwer.

Van der Pool, L. (2006) Deshpande Center revises game plan for funding. Boston: Boston Business Journal, 10-10-2006.

Sydney, G. (1999) Advanced Battery Technology. Power 99 - 7th annual international conference on power requirements for mobile computing and wireless communications Santa Clara, California October 3-6, 1999.

UPI (2006) Strong laptop growth seen in China. United Press International at http://www.physorg.com/news5645.html (accessed on 12-06-2006)

Le Grand, G. Ben-Othman, J. Horlait, E. (2000) Providing quality of service in mobile environments with MIR(Mobile IP Reservation Protocol) Networks, 2000. (ICON 2000). Proceedings.

Al-Zakwani, A. (2006) Mobile Devices - Evolution and Revolution: A Cause for Security Concern. ICOMP'06 in World Congress in Computer Science, Computer Engineering, and Applied Computing