Technical Advisor: Dr. Stuart Tewksbury

Group Members:
Lincy Scaria
Renée Brewster
Uzoma Onunkwo
Pawel Owczarczyk
Sharmishtaa Seshamani

CONTENTS

I ABSTRACT

This project is aimed at designing and building a real-time, wireless interface between personal digital assistants (PDA). The communication between these devices will be demonstrated by making a multi-player game for PDAs that will be made interactive by playing it over mobile wireless protocol.

The problems to be solved include establishing communication between two or more PDAs, finding a way to cope with interference, and allowing more than two persons on the network to be playing the same game. This involves manipulating game outputs into a format that can be transmitted via radio frequency signals and then interpreted and changed into its original format by the receiver. It also involves devising a way to seamlessly change frequencies in the case of interference. Solutions to these and other problems, including the making of a robust game to allow synchronization of all user responses, will be proffered and worked on.

With the use of PDAs in this project, the method of inputting information into the game is also a problem that needs to be solved since these devices do not have keyboards, but use a touch screen instead. In addition, a game convention that defines the responses needed to allow and refuse connection to a game session, to log off from an ongoing game session, and ensure quick updates of each game status, must be created.

The wireless aspects of this project is made more complex because our aim is to make the network a distributed wireless network in which each PDA acts similarly to a base station.

II PROJECT PROPOSAL PLAN

INTRODUCTION

"Wireless" is one of the buzzwords in the telecommunications industry today. The main emphasis in wireless communication relates to "wireless networks", "wireless Internet", and "wireless messaging and e-mail".

One wireless device that is of growing importance in the market is the Personal Digital Assistant (PDA). PDAs were first developed as a method of keeping track of daily schedules, contacts, and memos. Over time, they have been developed to include more advanced functionalities. Today, the latest PDA's include wireless internet access and wireless instant messaging. However, direct wireless communication among PDA's has not yet been achieved.

The present wireless technologies are: Wireless Application Protocol (WAP); Global System for Mobile(GSM); Wireless Instant Messaging; and Wireless LANs.

Based on our research of these technologies and their capabilities, we propose a project that will seek to improve upon the present technology by allowing multiple PDAs to communicate with each other using RF signals. This communication will be done in real time and will be demonstrated by playing a multi-player game over this wireless network.

The benefits of this project include real time communication between devices and the ability to have multiple recipients of transmissions. The uses of this project will span across industries, as this product will be beneficial in the corporate, educational, manufacturing and numerous other arenas. The main weakness of this project is the short transmission distance expected if transmissions are done without the help of a base station.

DESIGN REQUIREMENTS

In order to maintain a robust wireless distributed network environment, there are several requirements that need to be met:

- Establish communication
- Find a way to handle interference without disrupting communication
- Allow more than two persons on the network to use the same application concurrently

Establish communication

- Create the necessary protocols to achieve communication.
- Find a way to contact individuals on the network in order to start a game / session.

Dealing with interference

- Devise a way to change frequencies seamlessly in the event of interference.

Allowing more than 2 people on the network using the same application

- Set conventions to follow for initializing a game and joining a game.
- Check for existing frequencies when a device is ready to make a connection.

DESIGN APPROACH

An issue in accomplishing the functions of Virtual Vicinity devices is the problem of a distributed wireless networking environment. This spans over connectivity, data transmitting speed, bandwidth of transmission, signal strength and interference. In the Virtual Vicinity (VV) project, we aim to solve some of these problems and, at the same time, maintain a robust multi-user environment.

Establishing a connection between VV devices requires the use of protocols. The VV devices will allow for mobile communication, and thus requires that ample amounts of time be given for transmission and reception of data packets. Depending on the application, it may or may not be necessary for the receiver end to acknowledge the reception of data. Protocols that enable this include TCP/IP, RIP/SAP, and IPX/SPX. VV devices, at the present, will be implemented for playing a real-time, multi-user game. Multi-user games require persistent responses from every user. This means that acknowledgement signals (individual "responses") will be needed to sustain synchronized game playing, and as such, a data reception acknowledging protocol will be used. Protocols also play a role in the recovery of data packets. UDP and IPX do not provide a data recovery scheme. So in order to use them, we will have to develop and implement a recovery scheme. On the other hand, TCP and SPX are connection-oriented protocols that provide data recovery schemes. In this project, TCP/IP is favored because it provides packet-based networks, which are less costly and easier to implement as opposed to switched networks. In addition, it allows for decentralized control and dynamic allocation of unique numbers to possible VV users for communication. TCP/IP is also independent of transmitting medium, which allows for possible expansion of the VV device communication to other compatible instruments/machines.

As mentioned earlier, VV devices are intended for multi-user environments. This requires that appropriate convention be taken in establishing connection. One such convention that we proffered was to enable frequency hopping. This basically ensures that each connecting VV device checks for already existing (busy) frequency lines, and adjusts its frequency of transmission to prevent interference.

Interference can be from similar VV devices or from different RF generating devices. We will reduce and, if possible, eliminate interference between VV devices using an appropriate frequency selection scheme. Interference with other non-VV devices however, is bound to occur. This has been a problem even with cellular, PCS (Personal Communication Services), and Packet Radio communication. However, by

appropriately filtering received signals, and devising an encoding and encryption scheme of the data packets, outside interference can be reduced.

The transmission bandwidth being targeted is the unlicensed 2.4GHz – 2.4835GHz because it is in existence all over the world. Deciding on this will depend on the cost of obtaining the transceiver unit. It is still likely that we may implement the unlicensed 902MHz – 928MHz bandwidth (mostly existing in the USA) or other bandwidth depending on the availability and characteristic of the transceiver unit. Some of the transceivers under consideration include Iwireless' RF49 transceivers and Lucent's RG-1000 transceiver unit. Since VV devices will basically be wireless RF LAN products, they will be made compatible with the IEEE 802.11 committee. It is our aim to harness all the resources needed – transceivers, wireless software, PDAs, and protocols – to ensure 2Mbps – 11Mbps communication, thereby leading way to the advent of Virtual Vicinity where a real-time application can exist over a distributed wireless network environment.

APPLICATION

The application that we propose to build to demonstrate the use of a distributed wireless network environment is a game. The game will be interactive between two or more players, and have a graphical user interface.

2-D or 3-D game?

2-D and 3-D games each possess different elements. They both have different ways of displaying data to the player. Both are viewed on a 2-D screen. However, the difference begins with data storage. To display 2-D images, the mainstream method is to store all of the data in pre-drawn image files, whereas to display 3-D images, information about the object, such as its vertices or polygons that make up the object is stored. 2-D games are relatively fast to draw and process. However, animations are more flexible in 3-D games.

After careful consideration, it has been decided that a 3-D game would not be practical to implement because of limited memory size and limited display size on the devices available. Therefore, the team decided that the application should be done as a 2-D game.

Game Design

Overview:

The specific type of game has not yet been decided. However, some of the proposed games are Battleship, Chess, PacMan, Maxit, Snake, Connect-4 and Mastermind. Some of the important factors that need to be taken into consideration to build the game application are:

- Memory Size of the application
- Execution time required
- Programming language to be used
- Complexity of supporting software
- Graphical intricacy of the game

Memory Size:

The memory size of the application cannot be larger than the available memory in the device. At the same time, the game program has to be large enough to produce a complex enough application that will maintain user interest. In the Cassiopeia, the memory size available is 16MB, which is superior when compared to other handheld PDAs. This makes Cassiopeia a favored PDA in our design of Virtual Vicinity.

Execution Time:

The execution time of the game application is another factor to be taken into consideration. Since the game is interactive, the execution time will be made short enough to maintain a quick response time and retain user interest.

Programming Language:

The programming language to be used is a very important factor. The choice of a programming language is to be influenced by platform dependency, size of compilation software and compatibility of compilation software with the device's operating system. Available programming languages are C++, Java, Delphi and Visual Basic.

Supporting Software:

The complexity of supporting software also has to be considered. If graphical information is to be transferred over the wireless network, image compression software would be needed. The image compression software would reduce the size of the data to be sent across and therefore increase the efficiency of wireless data transmission.

Graphical Complexity:

The graphical complexity of the game is also an important factor to be considered. The Cassiopeia has a color display that is 240 X 320. Since the display is not very large in size, the game application cannot have a graphical user interface that is too large to fit on the screen.

Details:

The design of the game is divided into three main categories:

- I/O Structure
- Game Structure
- Program Structure

I/O Structure:

This involves the communication between the player and the game. The device that will be used has a touch-screen and therefore input from the user is going to be in the form of a "touch" on the screen. The output to the player will be in a graphical format.

In the case of Battleship, Connect-4 or MasterMind, the output will be in the form of a grid. In the case of chess, the output will be in the form of a checked board. In the case of PacMan and Snake, the output will be in the form of a maze.

Game Structure:

This part consists of the basic structure of the game and how the game is to be implemented. The game will be designed for two players and information will be passed in turns from one player to the other.

Information formats:

The type of information that has to be passed on from one player to the other will be dependent on the type of the game. In the case of many of the games under consideration, information will be displayed graphically to the user but the information that is to be transferred over the wireless network will be text. Therefore, we have to design an algorithm to implement this conversion.

Rules:

The rules of the game have to be decided. The rules will depend on the type of input that the player will use, and on the type of game that is being programmed.

Maximum Response Time:

The maximum response time for each player should also be decided. For example, during a game, if a player does not respond within a short period of time, it would be a waste of time for the other player to keep waiting, and it would also be a waste of connection time. Therefore, a timeout should be set.

Error Detection:

The problem of overloading the network with unwanted information is important to consider. For instance, if a player sends a response to the other player and then accidentally touches the screen again, generating information for transfer, this would be

an error. Obviously, we do not want this second piece of information to be transmitted. There should therefore be a mechanism within the game itself that detects this.

Menus:
Menus should be divided into subsections for the sake of comprehension. Subsection topics could be the following: New Game, Score, Quit, Restart Game, Help.

The New Game option would be used to initialize a game, the Score option would keep track of the players' scores, the Quit option would quit the game, the Restart option would restart the game with the original settings and the Help option would display help files. Creation of help files will also be additional software development. Help files will be needed in order to make the application more user-friendly.

User Identities:
The game would also have to be able to keep track of identities of users. This part of the programming would have to be incorporated with the client-server programming that is implemented for wireless communication.

Program Structure:
This part consists of constructing code for implementing the I/O structure and the Game Structure. As mentioned above, the programming language to be used has yet to be chosen. The program structure will be designed according to the programming language selected.

FINANCIAL BUDGET

The proposed budget for this project is presented below. These costs reflect our estimated materials, equipment, and documentation expenses. Labor and other costs have been omitted since they are irrelevant to the purpose of this proposal.

- The costs for materials and parts are calculated for four PDAs and two Notebook PCs.

- Documentation costs are calculated to take into consideration the costs associated with producing the necessary reports and presentations.

Components	Unit Cost	Total
Materials and Parts		
Personal Digital Assistants (PDAs)	$ 600	$ 2,400
Notebook PCs	$ 2,000	$ 4,000
Parts for base stations / transceivers		$ 1,000
Test Equipment		$ 100
Documentation Costs		$ 100
		$7,600

Table 1: Proposed budget for Virtual Vicinity Project

PROJECT SCHEDULE

The project schedule for the fall semester is concentrated on researching all aspects of the project, producing reports, and making presentations about the project. To this end, every individual in the group will take part in every phase of the project in the fall semester. The only specific segregation of duties applies to the actual writing of reports, where each individual will be responsible for writing a particular part of the report.

The research to be done this semester will serve to help the team better understand everything that will be involved in making the project a successful one. It will also serve as a planning tool, so that the activities next semester will be primarily execution of these plans. This research will help the group to decide on the game that will be used, the programming language that it will be written in, and the types of algorithms needed to program the game. The research will also allow the group to decide on a tentative design for the transceiver.

As mentioned above, the project schedule for the spring semester concentrates on the execution of the plans developed in the fall. At the beginning of the semester, the team will be divided into two subgroups that will work concurrently on separate parts of the

project. One subgroup will be focused on programming the game, testing it, and getting it to work with two PDAs connected with a cable. The second subgroup will be focused on building the transceivers and/or base stations needed for the wireless connection. When both of these areas are completed, they will be combined in order to have the PDAs communicate without the cable. At this point, we will attempt to address one or more of the problems faced by wireless communications. The specific problem to be addressed has not been decided on at this point in time, but it will be included in the final design report.

III CONCLUSION

This project is very relevant to today's technology and it exhibits several strengths and weaknesses. The major strengths of this project are that it has never been done before, most of the needed technology exists and is available, and it has great entertainment value. The main weakness is the high cost of the PDAs.

This project provides great opportunities because of its timing. It encompasses where technology is at this point in time and is therefore very relevant. Wireless capabilities are relatively new to PDAs and we are seeking to take it a step further with the view that in the not too distant future, wireless communication between PDAs will be the norm. The project also presents the project team a great learning opportunity.

However, there are some threats to this project that cannot be easily dismissed. All of the complexity involved in wireless communication cannot be solved in the three or four months that we have to work on the project. Also, the project team does not have the resources necessary to complete the project. To this end, the team will focus on one of the specific problems within wireless communication, and seek to solve that.

The most important aspect is that this project will make a contribution in the area of wireless communication between PDAs, and future teams can build on this project and

move toward achieving the ultimate goal of seamless communication between these devices.

IV REFERENCE

1. Nemzow, Martin - *Implementing Wireless Networks.* McGraw Hill, New York, 1995.
2. http://www.microsoft.com/technet/security/tcpip.asp (25th October, 2000)
3. Theodore S. Rappaport – *Wireless Communications: Principles & Practice*. Prentice Hall PTR, New Jersey, 1996
4. http://www.gamedev.com