Proceedings of the 24th IASTED International Multi-Conference
BIOMEDICAL ENGINEERING
February 15-17, 2006, Innsbruck, Austria

MOTECARE: AN ADAPTIVE SMART BAN HEALTH MONITORING SYSTEM

Einstein Lubrin, Elaine Lawrence, Karla Felix Navarro
University of Technology, Sydney
Broadway 2007 NSW Australia
{einstein, elaine, karla}@it.uts.edu.au

ABSTRACT
An increasing aged population worldwide promises to test medical capabilities everywhere. A growing number of groups – research and commercial – have taken on this challenge of alleviating a potentially unmanageable situation. They are looking at existing and novel ways to care for the aged; however, it will still be some time before the medical profession will accept such new solutions. In this paper, the authors propose a smart, adaptive remote health monitoring system as a solution. They describe the building of a third implementation of a remote health monitoring prototype using Motes, a PDA, a personal server and a network management application to show that commodity-based hardware can be a viable solution. In addition, they describe their intent to use the prototype in developing a system that can adapt to the patient's condition and situation. Such an adaptive, smart body area network (BAN) system requires a number of issues to be addressed before commencing development; these will also be discussed.

KEY WORDS
Motes, health monitoring, Body Area Network

1. Introduction

The potential high cost and human resource implications of aging populations on health care management are increasingly attracting the attention of governments. Predictions about the worldwide population of over-65 persons set the number at 761 million by 2025, more than double the 1990 figures [1]. Healthcare providers will need to devise cheaper and more novel ways to manage and care for sufferers of age-related chronic diseases such as heart disease, Osteoporosis and Alzheimer's Disease.

Already, a number of commercial and research groups are involved in developing tools or systems that can be used to care for aged people. Intel, for example, is developing its own brand wireless sensor network devices (WSN) called an iMote (Intel Mote) and a platform from which data processing can occur, called the Stargate. [2]. Harvard University is developing CodeBlue, a wireless sensor network technology useful for a range of medical applications, including pre-hospital and in-hospital emergency care, disaster response, and stroke patient rehabilitation [3]. The European E-Care system will monitor patients with chronic, or long-term, illnesses such as diabetes or cardiovascular disease, and patients discharged after an operation or serious medical crisis, such as stroke victims [4]. In Australia, the company Pro Medicus, a secure smart messaging system for transmitting pathology results to over 19000 doctors, is proving a success story in IT Healthcare [5].

The authors contend that the use of commodity-based sensors and open standards for their applications will lead to a cost-effective solution to the growing problem. The MoteCare team have built three remote monitoring wireless sensor network prototypes which use Motes, a PDA or laptop and a network management application to show that commodity-based hardware and open source software may provide one answer to the problem of remotely monitoring people in their own homes, in aged care facilities and even whilst moving around a city. The innovative use of network management tools to solve personal healthcare management issues underpins an essential feature of our research.

In the following section, the authors describe their current project that resulted in a number of finished prototypes. Descriptions of their previous prototypes follow, after which the description of their latest implementation is presented. In Section 5, the authors explain how they intend to use this prototype for developing smart health monitoring applications and the benefits of such a system.

2. Project Description

The project is to develop an autonomous system for monitoring aged or chronically ill persons in their homes, in aged care facilities or as they move about in their outside environments. The architecture is based on literature reviews of current and developing health monitoring systems, experiments with Motes and prototype developments and a novel approach of applying network management tools to health monitoring.

With most healthcare monitoring systems available or in development, the equipment used is generally proprietary and thus, would be costly to the patient and/or state. CardioNet, for example, employs PDAs to collect data from their own ECG monitor and send it over a cellular network to a service center [1]. Medtronic uses a dedicated monitor connected to the Internet to send pacemaker information to a medical professional [1].

The authors contend that one possible way to reduce the cost of such systems is to use open-source and commodity-based technologies. Open-source products eliminate the requirement to pay royalties, while commodity-based equipment reduces dependence on one specific vendor for maintenance; thus, resulting in cost savings. Motes – small wireless devices used for transmitting sensor data – are one such commodity-based technology, along with a PDA that was used in developing the system. Applying an open-source network management approach to the monitoring aspect also helps in reducing the cost. The authors assert that characteristic similarities between the network management and lthcare monitoring domains allow for adaptation of work management applications into this area.

Figure 1 below shows a high level view of the initial projected system. The system comprises a BAN (Motes Sensor Network - MSN) (1) connected to the remote MRTG server (4) across the WAN via a data collector (2), the base station, while the doctor is located in another area.

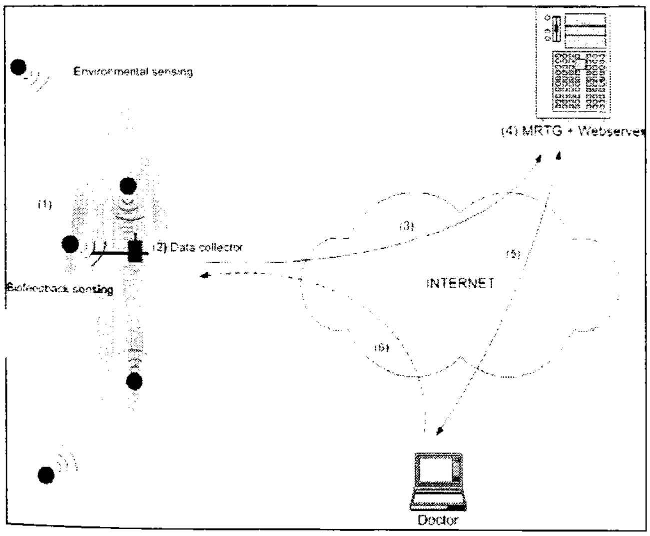

Fig. 1. Projected high level architecture of MoteCare [6]

3. Prototype History

There were two implementations of the health monitoring system prior to the most recent prototype; a summarized comparison of each can be seen in Table 1 below. The authors' main goal for the first prototype was to provide for remote monitoring of the patient. This was achieved by applying a network management approach to realize the remote aspect. An open source network management tool – Multi Router Traffic Grapher (MRTG) – was chosen to visually present the sensor data onto a webpage [7]. Normally, MRTG is used to monitor network traffic and SNMP data and post them on a webpage. It uses scripts based on Perl that identify the source of the data and formatting rules for graphical presentation. Data collected by MRTG is compressed and stored for, at most, two years before being discarded.

Table 1. Summary comparison of first two prototypes

First Prototype	Second Prototype
MRTG web server complete	MRTG web server complete (unchanged)
BAN connected to laptop	BAN connected to PDA
	Mobile Monitoring complete

MRTG provides a number of advantages. It is open source, so it can be readily adapted for other applications besides network management; it presents data visually and automatically using web based technologies; it provides short to medium term trends that can be useful with certain medical conditions; and it compresses data for efficient storage. Other open source network management tools, such as Cacti, use the same underlying engine to collect data as MRTG [7]. Further investigation into these other tools will be discussed in Section 5 of this paper.

The wireless sensor network (WSN) that was employed for the collection of the sensor data is based on an emerging technology called Motes. Initially pioneered by researchers from the University of California at Berkeley, there are now various prototypes of Motes available from different developers/vendors. Motes are essentially microcomputers that work in conjunction with different sensor boards to process and transmit sensor data. These Motes form ad-hoc mesh networks that allow monitoring over an entire household. Due to the small form factor of these devices, their potential can only be realized as networks of Motes. Data from this network passes through a gateway, comprising a Mote connected to a processing board to a remote server. A more detailed description is presented later in Section 4.

With this first prototype, the Motes base station was directly connected to the MRTG server on a laptop. In the proposed architecture, the server (MRTG) would be placed in a remote location – a hospital, for example – while the MSN would be located elsewhere – at the patient's home. Included in the architecture was the ability for mobile monitoring; the patient or doctor present would be able to monitor the person's condition directly and instantly with a portable device (i.e. – mobile phone, personal digital assistant (PDA), etc). Implementing this architecture was the subject of the second prototype.

3.1 Second Prototype

The second prototype was a proof-of-concept of the proposed architecture; the MSN was kept intact, while the MRTG server was redefined and the mobile monitoring (termed Mobile Monitor, refer to Fig.2) capability was added. In order to separate the server from the MSN, the data processing responsibility was removed from the server and placed into the Mobile Monitor. Thus, MRTG now only acted to present and update the sensor information coming from the MSN.

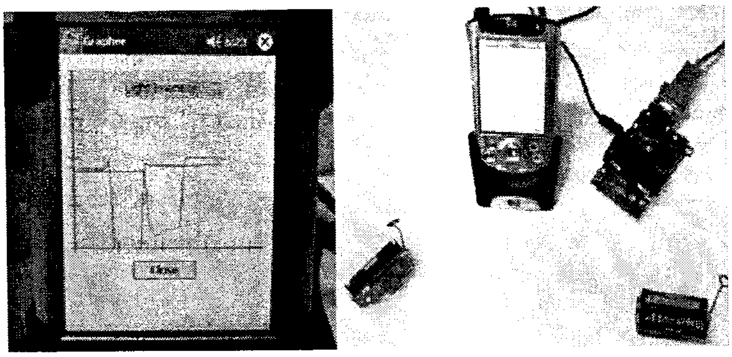

Fig. 2. Laboratory set up of second prototype

The Mobile Monitor performs mostly the same functions as MRTG with the addition of data processing. It presents data visually to the patient/doctor albeit in real time compared with MRTG (every five minutes). With the Motes base station now directly connected to the Mobile Monitor – in this case, a PDA was used – sensor information is extracted and processed from packets coming through the MSN within the device itself. Now the patient is able to carry the device throughout the household. Similarly, the data processed by the monitor is simultaneously sent, via wireless (Wi-Fi), through the Internet to MRTG.

Preliminary tests on the performance and usability of the system have shown that several improvements need to be made. One issue that requires improvement pertains to the real time perception of the system, mainly in the updating of the graphs. Over time, the user would experience a slowdown, where the delay between a change in the conditions and that change appearing on the graph, increases. Real time processing of the data within the PDA itself requires a significant amount of resources and so may be a major factor in performance. This is further addressed in Section 4.

In terms of usability, the authors believe that since the device is primarily intended for either medical personnel or patient, or both, the application should be simple to use. This is supported by the results of two surveys that the researchers carried out to ascertain whether a wireless health monitoring system would be accepted; ease of use featured strongly as a requirement among the respondents. In fact [9], who has done a study on a majority of the acceptance models available, has shown that 'Ease of Use' is one of the most important determinants of new technology acceptance. Aged, infirm and chronically ill people could not be expected to cope with difficult to use devices and applications. As the authors' experience with motes has shown that they are difficult devices to master at this stage, we have concentrated on providing a user friendly interface via a PDA. The next section will describe the latest addition made in the development of the health monitoring system, providing a foundation for expanded capabilities.

4. New Prototype Description

Building on the previous two versions, this latest implementation aimed to improve the user experience by augmenting the system with increased functionality. It also aimed to resolve a number of issues such as fixing the slowdown in graphical refresh and providing a better graphical interface for an improved user experience. Figure 3 below shows an updated high-level view of the system.

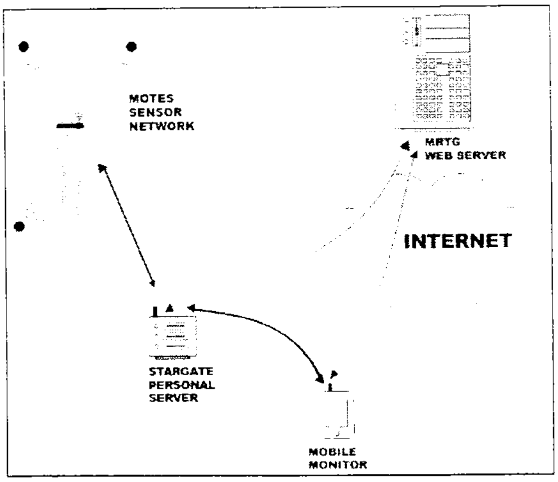

Fig. 3. New high level architecture of MoteCare

4.1 Motes Sensor Network

In accordance with the previously proposed architecture, the Mica2 family of Motes was used in this prototype system. These Motes, commonly known as "Smart Dust," are tiny microcomputers that employ wireless media to communicate with other Motes. Made with off-the-shelf parts, they are relatively low cost compared with other wireless devices with the same functionality and allow for what is known as Commodity-Based Wireless Networking. Ranging in size from a few centimetres to a matter of millimetres, they can be placed in the most space-constrained area. Each Mote has the capability of providing various sensor measurements, ranging from measuring the surrounding magnetic field and sound level, to measuring temperature and acceleration. With a variety of sensor types, applications for these Motes can be realized such as in military applications (enemy detection) or in health applications (monitoring of vital statistics) [10].

Every Mote has the ability to self-form an ad-hoc wireless network with other Motes [11], [12], giving them the ability to cover a large area. Because these Motes are wireless and can self-form into a network they may be placed in remote regions which people find difficult to access. All these benefits consequently come with a price. The Motes' small size limits the amount of power available to each device however the effect of this constraint is reduced since most of the component parts of a Mote consume little power [10].

In this implementation, all the sensors except for the magnetometer were used. The motes were programmed with the modified Surge program from [13] to monitor changes in brightness, temperature (environmental and external body), and horizontal and vertical movement.

The lowest level of the architecture consists of the sensor nodes which provide physical access to vital signs and environmental parameters. These devices are placed in areas of interest and collect data primarily about its immediate surroundings. The motes could be used to monitor body temperature, heart rate and pulse rate, as well as many other parameters once particular sensors such as pulse oximeter sensors or electromyogram sensors are attached. Different versions of the basic design are being developed, but the authors will focus on the MICA2 Mote and MTS310CA sensor board (see Fig), whose most interesting technical aspects are [10][14]:

- Atmel ATMega 128 7.7 MHz processor
- 512 KB on-board flash.
- Tunable FM radio
- Frequency: 315, 433 or 915 MHz.
- 38 Kbps encoded
- 2 serial ports
- Over-the-air programmable
- Unique ID
- 6-channel 10 bit ADC converter
- Light, temperature, acoustic, 2-axis accelerometer, 2-axis magnetometer and sounder

4.2 Stargate Server

The architecture of the system was augmented with a local server – the Stargate server – to remove the processing burden from the Mobile Monitor (refer to the updated high level view on Figure 3). Previously, the Mobile Monitor performed all tasks of receiving the sensor network packets, parsing them to obtain the required sensor data, presenting the data on graphs and repackaging them for transmission to the MRTG server. Test runs on the prototype showed that over time, the user would experience a significant slowdown in the refresh rate of the graphs. Analysis of the underlying code showed that the data buffer was the likely cause. However, reducing the buffer limit meant that much of the more recent data would be lost and could pose a risk to the patient. Adding more functionality to the application – providing configuration options, multiple graphs for different rooms, etc – would only serve to exacerbate the user experience.

The Stargate is still located locally in the patient's home, but will now act as the gateway to the MSN. Portable devices such as the PDA can connect with the Stargate and use its capabilities. As the gateway to the MSN, the Stargate is programmed to accept commands from clients, such as requesting to transmit data, requesting sensor information, focusing on a Mote, changing the sampling rate, turning on/off a Mote and turning on/off different sensors.

The Stargate is the result of researchers from Intel developing a Personal Server for 'ubiquitous access to personal information' [2]. Key features of the Stargate server are:

- 400 MHz PXA55 XScale Processor
- 64 MB SDRAM, 32 MB Flash
- Ethernet connection
- Compact Flash slot
- Built-in bluetooth
- Wi-Fi ready
- Linux kernel

4.3 Mobile Monitor

The Mobile Monitor consists of a PDA that is wirelessly connected to the Stargate server. The PDA is an HP iPAQ rx3417 with Windows Mobile 2003 Second Edition and the following technical specifications:

- Samsung S3C 2440 processor
- 92 MB Memory
- Integrated Wireless LAN 802.11b

- Integrated Bluetooth
- Serial cable connection

Its new implementation results in a true mobile experience as the MSN gateway was made separate from the PDA. The monitor serves a number of major purposes within the system:

- Provide real-time remote monitoring to the medical professional
- Provide MSN configuration capabilities
- Provide an intuitive warning system.

A graphical front-end was created to display the readings from the sensors in real-time. It can monitor a maximum of 65,535 motes at one time, albeit with performance tradeoffs [10]. The application is written in C# using the .NET Compact Framework. Initially, Java was chosen as the language and platform for the application. However, for simplicity, C# was selected as the compact framework can easily be installed to any Pocket PC or Windows Mobile device. The compact framework is also a subset of the full .NET framework, making the transition from a desktop to portable platform smoother [10] [15]. Using Sun's J2ME platform requires choosing between different configurations and profiles depending on the PDA used. In addition, depending on the configuration (Connected Device Configuration {CDC}) or (Connected Limited Device Configuration {CLDC}) compatibility with the desktop platform API may not be easily achieved [10][16].

```
// write the readings for the motes if all stated
// motes have data
oscope.dataAvailable(numMotes, DumpUpdate.valueMotes,
    DumpUpdate.idMotes);

// format out the data into a file
DumpUpdate.formatPrint(formatFile, 0, 0, true);

// reset flags
DumpUpdate.printOut = false;
DumpUpdate.count = 0;
DumpUpdate.valueMotes = new ArrayList();
DumpUpdate.idMotes = new ArrayList();

// send to web server
if(client.isConnected())
{
    client.TCPsendFile(formatFile);
}
```

Fig. 4. C# sample code [10]

Additional elements were added to the existing graphical user interface (GUI) to accommodate the new capabilities of the system. For example, the user – either the patient or a doctor present – can now open a new window in the application to configure at least 1 Mote in the sensor network. They can be configured by first locating them with the 'Beep' button, then by clicking on a specific parameter to change. Other capabilities of the Mobile Monitor include the ability to show error/debug messages during runtime, change connection settings and display help pages. Another major capability for the Mobile Monitor is the ability to monitor multiple rooms – the bedroom, living room, etc. Here, the user can switch views between them whenever shifting locations.

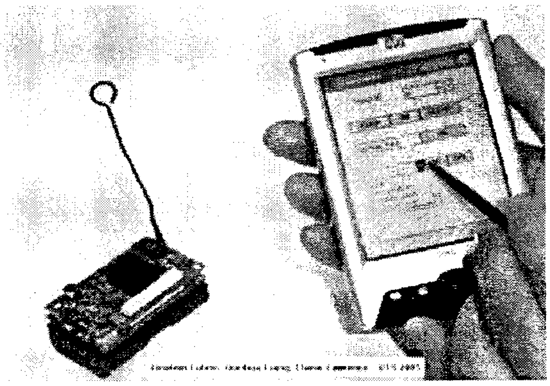

Fig. 5. Interface view of network control

Changes to the layout of the interface were also made as consideration for usability was addressed (refer to Fig. 2 and 6 for a comparison between the old and updated designs). For example, users can switch between a graphical representation of the data or to a textual one. This is mainly for personal preference – users may prefer to look at numerical numbers rather than graphical lines. The choice of whether to view debug/error messages is up to the user (default is 'off'); this is an example where the user can decide on his/her level of involvement/comfort with the system.

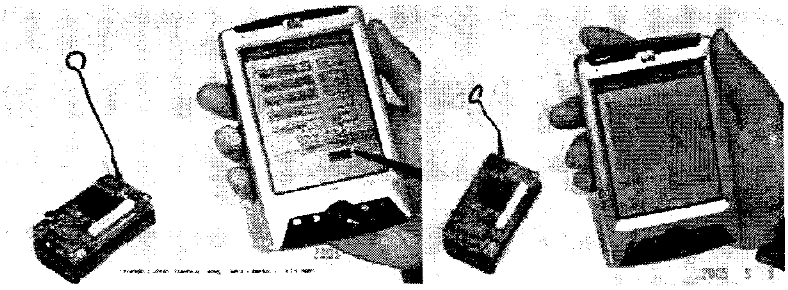

Fig. 6. Textual and graphical representation of sensor data

The monitoring application uses sockets to make a connection to the Stargate server on the network. Microsoft's .NET Compact Framework provides two ways of accessing services from a network computer/server: Sockets and XML Web Services. Sockets provide access to lower level protocols such as TCP; XML Web Services provides access using HTTP [15]. Considering that the mobile monitor will communicate with the server within a private network, the choice of using sockets is suitable. If the web service had)een used, large amounts of XML data would have had to be transferred [15]. Future implementations may employ the XML Web Services as part of a profiling scheme, which will be discussed later. Our socket solution meant that data is transmitted as a text file and is directed to the MRTG server for processing, representing a huge reduction in network traffic

4.4 MRTG Server

The application server consists of the MRTG server and the socket server. The MRTG server employs a Perl script to read the received text file and extract the required data. It uses Microsoft's IIS as the web server to support the web site. Using IIS and .NET framework allows the most compatibility with applications developed on the compact framework.

The device being monitored normally supports the SNMP network management protocol, but MICA2 motes do not support this protocol. MRTG is able to support non-SNMP enabled devices by creating a personalized script that would do the readings or polling of information instead of using the SNMP protocol. This makes MRTG a very flexible and powerful tool way beyond its use as a network management application alone. The source is a text file which is the output of the C# application in a patient's or doctor's PDA. When this file arrives at the MRTG server, the script will open it, extract required data, format into SNMP protocol, and pass it to MRTG.

In our experiments we captured light readings as proof of concept as seen in Figure 4. Every time MRTG captures another measurement, all files are updated so several patients can use the service at the same time. In this case we will need a database for storing all the health data related to each patient in a structured way. We propose the use of the latest release PostgreSQL [17], a free and open source database that is now able to run as a server natively on Windows, or another stable database.

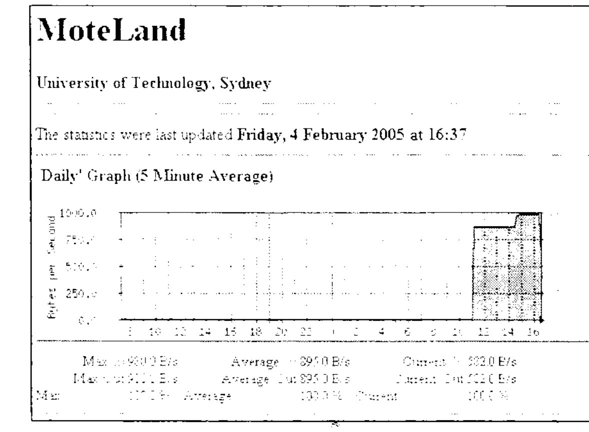

Fig. 7. Sample MRTG website displaying light sensor data graphically

The authors contend that MRTG provides the system with a way for publishing in a website all the health and environment parameters sensed by the wireless network sensor in form of graphics. The advantage of this prototype is the fact that such graphics are also easily displayed on a small mobile device thereby increasing the readability level of the data. Further research into other open-source applications is underway for possible alternatives to MRTG. For example, the combination of Cacti (network management) with HealthMonitor (computer management with alert system) would serve to provide the remote monitoring and warning system capabilities for MoteCare. Cacti uses the same underlying collection/compression engine as MRTG, rrdtool, while providing for more long-term data storage and analysis. HealthMonitor is a computer management tool that provides for logging and warning capabilities [18].

5. Introduction to the Smart Monitoring System

The development of the system has reached the point where the possibility of implementing an adaptive system can be realized. The authors define a smart remote healthcare monitoring system as a system that automatically changes sensor parameters and notification settings according to the patient's condition, profile and further patient/doctor input.

In this adaptive smart monitoring system, MoteCare, the authors have identified several areas of research and development that need to be undertaken. At the moment, this includes, but is not limited to:

- Authentication and authorization of users
- Profiling of patients and doctors
- Automatic modification of sensor parameters and notification according to profile, current patient and doctor input and current context
- Efficient transport of medical data over wireless
- Secure transport of the data
- Secure storage of patient records

Any application that deals with patient information must include a high level of privacy such that only authenticated users are allowed to gain access to personal data. Of those users that are authenticated, only those that are authorized to view the information should do so. For example, a family doctor should only be able to view health records from his patients and no one else's. This leads to the requirement for profiling both patients and doctors/medical personnel. Depending on their profile – if they are a patient, doctor and nurse – various levels of access are granted. A doctor, say, would be able to change parameters of the MSN while being able to view current readings; whereas the patient and nurse may only view the current readings and provide notes to add to the record. E-Care's proprietary system dynamically produces data depending on who accesses a patient's record. A doctor will see all the health information, a system or medical administrator will see data relevant to them, while patients, their friends or family, will see another set of data, all coming from the one file [4]. The authors propose to use a health record system stored on a remote server – MRTG or Cacti– which would provide the correct profile for the individual logging in to the system (either remotely or locally on the Mobile Monitor).

In conjunction with the profiling of patients and medical personnel and input from any individual involved in the patient's care, the new system should automatically modify and notify sensor parameters. As an example, if a patient who was being monitored was known to have heart problems and was recently having increased symptoms, the system would react by increasing the sampling rate of the sensors. The system is then more focused on the patient than under normal conditions so that in the case of an emergency, the system would send out an immediate notification to the hospital requesting immediate aid.

For privacy requirements, the system should also be secure from intruders whose aim is to compromise the system. This means that the system should be able to provide secure transport and storage of patient records and sensor data. The following table illustrates some of the current thinking on how to secure wireless sensor networks. The issue is especially relevant in the case of healthcare data. Experts from Berkeley believe that security for motes must be integrated into every component as components designed without security are the vulnerable points where attacks start [19]. The accuracy and security of vital medical information received from motes must be maintained and is an area for further exploration. If motes move into critical healthcare applications they may be subject to regulation as medical devices [20]. Of course, performance issues also need to be taken into account as the application should exhibit real-time characteristics.

Table 2. Security and Motes [19] [21]

Method	Comment
Key establishment	Traditional methods will not be able to scale up adequately with hundreds of nodes
Trust setup	Necessary to establish a secure and efficient key distribution for large scale sensor networks.
Secrecy & Authentication	Research has indicated that software only cryptography is practical with motes

Another issue that needs to be addressed is to ensure that the huge data stream of information regarding the patients does not overwhelm the medical personnel and does not clutter the information system with unnecessary detail. However sensors are capable of sending vast streams of personal, private, medical information wirelessly where it could be intercepted by unauthorized people or perhaps mishandled by medical/office personnel who could be unused to handling such large streams of information and who may be unable to work out which data should be kept and which discarded in a safe fashion. The huge data stream of information regarding the patients must not overwhelm the medical personnel and clutter the information system with unnecessary detail. Healthcare providers or others who have access to health information and do not act on it may incur substantial liability. [20].

6. Conclusion

In this paper, the authors have discussed the need to develop a low cost, robust system to care for the aged and those in poor health. They have presented a third prototype of their health monitoring system, MoteCare, providing remote and locally available mobile monitoring and the provision of an easy to use graphical user interface. . The use of commodity-based technology –

Motes – and applying network management techniques using MRTG have shown that indeed, low-cost healthcare monitoring is possible. The next objective is the implementation of an adaptive, smart monitoring system. With an increasing attention being focused on healthcare, particularly developing new technologies for healthcare monitoring, the authors believe that the MoteCare system shows a way to develop a smart, adaptive Body Area Network that will help to improve the daily lives of elderly, infirm and chronically ill persons. Further research will concentrate on using medical sensors such as pulse oximeters, etc on MoteCare, and developing secure yet easy to use interfaces to handle the private medical data.

References

[1] P.E. Ross, *Managing Care Through the Air*, IEEE Spectrum, December 2004, pp 14-19.

[2] *Stargate: A PlatformX Project* [Online] Available: http://platformx.sourceforge.net, accessed 5 Sept, 2005

[3] *CodeBlue: Wireless Sensor Networks for Medical Care* [Online] Available: http://www.eecs.harvard.edu/~mdw/proj/codeblue/, accessed 5 Sept, 2005

[4] M. Devoti, *Remote Healthcare Monitoring Not so Distant*, Information Society Technologies, March 2005. [Online] Available: http://istresults.cordis.lu//index.cfm?section=news&tpl=article&ID=74817, accessed 6 Sept, 2005

[5] K. Dearne, *Project Flatline*, Australian IT, 6 Sept 2005, p1.

[6] J. Riudavets, K.F. Navarro, E. Lawrence, R. Steele, M. Messina, *Multi Router Traffic Grapher (MRTG) for Body Area Network (BAN) Surveillance*, 2004

[7] *MRTG: Multi Router Traffic Grapher* [Online] Available: http://www.mrtg.org

[8] *Cacti: The Complete rrdtool - Based Graphing Solution*, 2005 [Online] Available: http://www.cacti.net Accessed: 6 Sept 2005

[9] N. Mallat, *Theoretical Constructs of Mobile Payment Adoption*, IRIS27

[10] E. Lubrin, E. Lawrence, K. Felix Navarro, *Wireless Remote Healthcare Monitoring with Motes*, Proceedings of the International Conference on Mobile Business (ICMB), July 2005, pg 235-241

[11] W. Connor, L. Krishnamurthy and R. Want, *Making Everyday Life Easier Using Dense Sensor Networks*, Ubicomp, 2001, LNCS 2201, pp 49 – 55, Springer-Verlag berlin Heidleber, 2001.

[12] PBI Media, *Strong Prognosis for Wireless in Healthcare*, Communications Today, 24 February, 2003. [Online] Available: http://www.findarticles.com/cf_0/m0BMD/35_9/98013124/p1/article.jhtml

[13] Crossbow Technology Inc., 2005. [Online] Available: http://www.xbow.com Accessed: 4 Sept 2005

[14] Mica Wireless Measurement System, Davidson Measurements [Online] Available: http://www.davidson.com.au

[15] Microsoft Developer Network, 2005. [Online] Available: http://www.msdn.com

[16] Java Technology, 2005. [Online] Available: http://java.sun.com

[17] Group, P.D., PostgreSQL 7.3. Documentation. 2002.

[18] *HealthMonitor: Your Open Source Monitoring Solution*, 2005 [Online] Available: http://healthmonitor.zucchetti.com/ Accessed: 6 Sept 2005

[19] D. Culler and W. Hong, *Wireless Sensor Networks*, ed. Communications of the ACM, June 2004, Vol 47, no 6., 2004

[20] J. Riekki, J.P. Huhtine, P. Ala-Siuru, P.Alahuhta, J. Kaartinene and J. Roning *Genie of the Net: Context Aware Information Management*. 2000 [Online] Available: http://www.pbol.org/projects/genie/publications/infomanage.pdf

[21] E. Lawrence, K. Felix Navarro, J. Riudavets, M. Messina, *Macroscopic Sensor Networks: Application Issues in the Healthcare Industry*" Third International ISCA Conference on Computer Science, Software Engineering, Information Technology and eBusiness and Applications (CSITeA 2004), December 27-29 2004