ATNAC 2010 - Australasian Telecommunication Networks and Applications Conf., Nov 2010
This copy contains corrections which are not present in the camera-ready copy.

Application of Virtual Mobile Networking to Real-Time Patient Monitoring

Devan Rehunathan
University of St Andrews
School of Computer Science
Email: dr@cs.st-andrews.ac.uk

Saleem Bhatti
University of St Andrews
School of Computer Science
Email: saleem@cs.st-andrews.ac.uk

***Abstract*—We aim to merge the benefits of network mobility and virtualisation to provide a simple, mobile and secure method for providing mobile network (as opposed to mobile host) platforms. We demonstrate our approach by showing the use of a mobile network of sensors and wide area connectivity for maintaining and managing a Wireless Body Area Networks (WBAN) for healthcare. WBANs are a mature field of research, where the challenges and applications have been explored for quite some time. One of the most promising applications for WBANs is healthcare. Wireless sensors are used to monitor patient health statistics and activity. With the ubiquity of wireless mobile personal devices (such as smart phones), their increased CPU and power capability, the feasibility of using them to build mobile network platforms is increasingly possible. In this paper we describe our novel approach, which is to utilise, through virtualisation, an individual's smartphone not only as a mobile router that manages his personal mobile network, but also as a platform to host his WBAN.**

I. Introduction

Recent technological advancements in integrated circuits, wireless communications and physiological sensing allow miniature, light-weight, ultra-low battery power, intelligent monitoring devices. Advances in wireless technologies and communication standards such as ZigBee, 802.15.14 and Bluetooth are making it possible to form sensor networks from separate sensors with greater ease. Utilisation of such technology in mobile Wireless Body Area Networks (WBANs) for medical purposes allow for inexpensive, unobtrusive and unsupervised monitoring for patients during their daily activities for prolonged periods of time. These wearable sensors have applications in teh treatment of stroke, physical, myocardial infarction rehabilitation as well as traumatic brain injury rehabilitation [13].

WBANs are a mature field of research, where the challenges and applications have been explored for quite some time [19] [8]. One of the most promising applications is that of healthcare, where wireless sensors are used to monitor patient health statistics and activity [11] [19].

If a WBAN is considered as a single network, it is possible to engineer this network into a *mobile network*. In this way, patients can move uninhibited between separate access gateways and any existing network layer connections can be handed over from one network to the next for seamless connectivity. The advantages if this are that the patient is more comfortable, no longer encumbered by the multiple wires of these sensors, and physicians will be able to monitor the vital signs of these patients remotely despite them being mobile. Applications of this approach can not only be found in medical care but also in performance sports and gaming. Development of a standard solution in network mobility is being carried out by the IETF MEXT working group [1].

The benefits of mobile WBANs, especially in medical care can be described as ease of provision, maintenance and management. For provision, physicians will be able to provide the sensors as well as the virtual management module which can be uploaded privately and securely to the patient's smart phone. The smart phone will then act as a mobile gateway for the remote and local management of the WBAN. The virtual platform offers additional benefits such as increased security and a minimal mobility footprint [21].

For maintenance, a virtual management module – implemented as a virtual machine image (VMI) – can be backed up easily onto a desktop computer or a remote server. These VMIs can also be migrated onto spare/redundant mobile phones for added robustness, if the a mobile phone fails, for example. VMIs can also be easily modified for unique patient configurations. Having a separate management module also means wireless sensors can be added, removed or configured in real-time.

For management, the mobile WBAN is connected to the wide area network, allowing for real-time management of the WBAN remotely or locally as long as network connectivity is available. A virtual management module for the WBAN also improves the security of any application provided as a third-party VMI as it is effectively sand-boxed at run-time, and users can easily restrict access with native file and sensor permissions.

II. Background

A. WBAN Requirements

Mobile WBANs have a unique set of requirements that should be considered to ensure their successful deployment. Possible requirements such as biosensor design and sensor system design [14] are especially important for our medical scenario.

[1] www.datatracker.ietf.org/wg/mext/charter/

For biosensor design, our scenario needs sterile sensors as the intended target could be someone who is ill or someone who is recovering from illness. Similarly, the sensors should also be as comfortable as possible as they are likely to be attached to the subject for a considerable period of time, both day and night. It would also be beneficial if the sensors were discreet and/or visually unobtrusive, for patient comfort and ease.

For the sensor system, the network should be made from off-the-shelf components to keep costs low. We believe that an open market and standard architecture will encourage early adoption for hospitals. It will also mean that physicians and patients will be able to select additional sensors to incorporate into the mobile WBAN.An open market would also encourage a large variety of sensors to be available. Such a market does not exist today, and we take the position that showing the feasibility of a low-cost, easy-to-use mobile network platform enabled using a consumer device, such as a smartphone, would encourage such a market.

B. Previous Work

WBANs have an important role to play in the future of mobile medical care: they have been considered for a number of years for potential healthcare applications. Below is a brief summary of key points of some projects that are particularly relevant.

One of the first application areas targeted has been cardiovascular monitoring (via ECG sensors). Projects such as CardioNet [2] have developed their Mobile Cardiac Outpatient Telemetry (MCOT) system that enable real-time heart beat monitoring, focusing on monitoring and treatment of arrhythmias. CardioNet also allows for integrated analysis and response, as it has an integrated diagnostic and patient management tool.

Another potential application is in the realm of first aid responders in a disaster recovery situation. AID-N (Advanced Health and Disaster Aid Network) [12] is one such example. Its main objective is to allow healthcare providers at disaster scenes and medical professionals within remote hospitals the ability to consult specialists who are geographically distant, on patient issues. Patients wear motes (MICAZ), which record their vital signs and transmit the data to a central database. Motes within an area form a wireless ad hoc network along with an on-site portable tablet PC which acts as a hub. AMON [16] is a similar system for patients with cardiac and respiratory problems. It works in the same way, through wrist-worn sensors, with continuous data collection and evaluation of multiple vital signs.

The usage of sensors in a WBAN usually calls for a transit/relay device that acts as a data sink and also as a router to the outside world. Usually, a portable tablet PC or, more recently, smartphones are used for the latter. HealthGear [18] and Mobihealth [5] are two such examples. HealthGear uses Bluetooth to form an ad hoc wireless network between its sensors and a cellphone. The cellphone stores, transmits and analyses the data and presents it to the phone user. Mobihealth was developed for the continuous monitoring of patients outside of the hospital environment. It proposes to integrate sensors and actuators to form a wireless BAN. Mobihealth goes further to leverage smartphone connectivity to push data online. One of the purposes of the project was to test the ability of 2.5G/3G infrastructures to support value-added healthcare services.

Another important aspect of WBANs is the placement of the sensors. One novel approach has been to embed the sensors with clothing. Some examples of this are a Lifeguard monitoring system [3], HealthGear [18], Wearable BAN [17] and LifeShirt [4]. Lifeshirt has the additional ability to share information with peripheral devices and incorporate patient input, which is then collected via a PDA and transmitted online.

There are also projects which go further and try to integrate WBANs on a larger scale. The CodeBlue [15] project looks at building an entire wireless infrastructure to support sensing via the WBANs. The BASUMA Project [1] is another such project that focuses on multimedia applications. The MyHeart Project [6] aims to integrate sensing into a person's daily routine to achieve user-defined lifestyle goals, such as better health and personal health goals.

C. Mobile Networking through the NEMO Protocol

NEMO enables network mobility by using an additional IP address, the *Care of Address (CoA)*, for the Mobile Router (MR). The CoA can be seen as a temporary address used by the MR as it moves. The CoA allows packets to be routed to the current location of the MR. The CoA acts as a topological locator. Meanwhile, the MR maintains another IP address that is available via DNS, its *Home Address (HoA)*, at its 'home network' (the IP sub-network to which the HoA belongs), and this is used for maintaining session state with Corespondent Nodes (CNs). The HoA acts as an end-system identifier, and is used for transport layer state. When the MR is not at its home network, the Home Agent of the MR (HA_{MR}), acts as a proxy for the MR, forwarding packets received at the home network (using the HoA) to the MR (using the CoA), via a bi-directional, IP-in-IP tunnel. Traffic from within the mobile network is sent to the MR. This traffic is encapsulated through this tunnel back to the HA where it is de-capsulated and forwarded. To CNs, the mobile network appears to be fixed.

The NEMO approach allows the MR and its Visiting Mobile Nodes (VMNs) to maintain pseudo-end-to-end connectivity despite changing network attachment points. A VMN achieves this by keeping its own Home Agent (HA_{VMN}) updated with its new CoA, using Mobile IPv6, as it moves. One benefit of this approach is that it does not change the way the IP address is used today. There are also no additional changes required to the IP architecture. The location of the mobile network is inconsequential so long as the MR and its HA_{MR} can set-up and maintain the bi-directional tunnel between them.

When a MR running NEMO migrates to a foreign network, it replies to any routeing Advertisements it receives from the local *Access Gateway (AG)*, to receive a new CoA on the visited link. The MR then sends a Binding Update (BU) message to its HA_{MR}, informing it of its change of CoA (See Figure 1 step (1)). The HA_{MR} updates its HoA-to-CoA cache for that MR and replies with a Binding Acknowledgement (BA). This act sets up and maintains the bi-directional tunnel between them.

Packets for the MR are received by the HA_{MR}, which uses IP-in-IP encapsulation to forward the packets to the MR at its latest CoA. All egress packets from the mobile network, sent from each VMN to its CN, must follow the same return path through the MR-HA_{MR} tunnel first before proceeding to its own respective HA_{VMN}(s) (See Figure 1 step (6)).

A mobile node has its own *Home Address (HoA_{VMN})*, which is always returned when a DNS lookup is performed for that mobile node. When this node becomes a VMN and joins a NEMO mobile network, it first receives its new CoA (Figure 1 step (2)). It then updates its HA_{VMN} with its CoA by sending a *Binding Update (BU)* message (See Figure 1 step (3)). The HA_{VMN} responds with a *Binding Acknowledgement (BA)*. If the VMN is communicating with any MIPv6-aware CNs, it will execute a return routability test(RRT) (Figure 1 step (5a)) and subsequently update its CNs with its new CoA, via a BU/BA exchange (Figure 1 step (5b)).

Upon receiving its CoA, a VMN running MIPv6 maintains its own bi-directional tunnel between itself and its own HA_{VMN}. Operationally, the VMN-to-HA_{VMN} tunnel exists within the MR-to-HA_{MR} tunnel. Mobility of the MR and VMN is hidden as all traffic eventually is sent to/from their respective HA_{VMN}s.

If the MR changes location, it will again negotiate and receive its new CoA and update its HA_{MR} with its new location (Figure 1 step (4)). The HA_{MR} then updates its Binding Cache and the bi-directional tunnel is maintained as it forwards MR packets to the new location.

As for the VMN within the mobile network, it will be unaware of its own mobility as the MR ensures that the address on its ingress interface remains unchanged. The mobility of the MR only affects its egress interface. As a result, the VMN will not execute any handovers with its HA_{VMN} or CNs.

D. Virtualisation with User Mode Linux

Modern computers are sufficiently powerful to use virtualisation to present the illusion of many smaller virtual machines (VMs), each running a separate operating system instance [7]. With the rapid advancement in the mobile phone industry, smart-phones are quickly reaching the point where they too can run VMIs[2]. Companies such as VMWare are already planning to release versions of their virtualisation platform for mobile phones[3]. We also see a similar trend towards the creation of person mobile wireless networks. Android is planning this for Froyo 2.2[4]. Audi recently built a car that

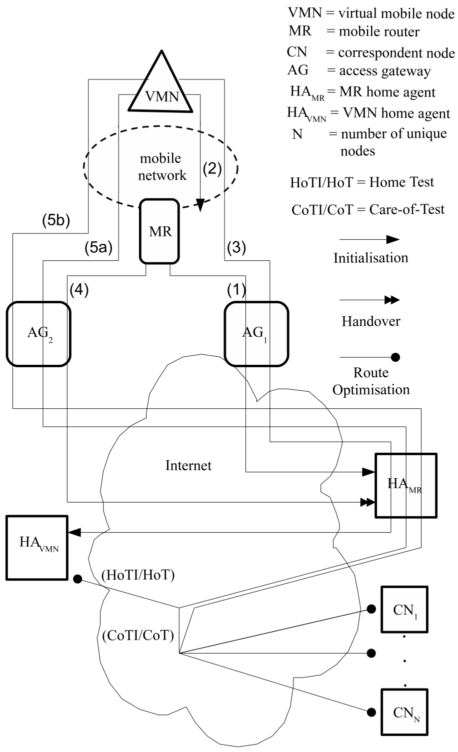

Fig. 1. The phases of initialisation and handover for a VMN (running Mobile IPv6) and MR (running NEMO). Step (1) shows the MR updating its HA_{MR} via AG_1. Step (2) shows a VMN arriving at the mobile network and registering an IP address gained from the MR. Step (3) shows the VMN updating its own HA_{VMN}. Step (4) shows the MR moving and conducting a handover by informing its HA_{VMN} of its new CoA. Step (5a) shows the VMN executing a RRT with its CNs. Step (5b) shows the VMN updating its CNs with a new CoA.

has mobile Wifi built-in [5], via their new *MMI Touch and Infotainment System*.

Currently, smartphones are released with proprietary operating systems already installed. While it is possible in some cases for users to install alternative operating systems, this may void the warranty of the phone or may even be considered illegal. By leveraging virtualisation technologies, users are able to safely incorporate features of other operating systems into their default installed ones. With the ability to run multiple virtual machines at the same time or switch between them, the user also gains access to greater functionality and choice.

The use of virtualisation, where the virtual machine is run as a single process, allows the user to manage the resources (e.g. CPU time and hardware interfaces). This means that despite having limited hardware resources, these resources can be fully utilised by switching between virtual machines. Additional benefits such as portability, flexibility and security are also inherent characteristics of employing virtualisation techniques.

User Mode Linux (UML) is a port of the Linux kernel to Linux. It implements a Linux virtual machine running on a Linux host. Its hardware is virtual, being constructed from

[2] www.qualcomm.co.uk/products_services/chipsets/snapdragon.html
[3] www.vmware.com/company/news/releases/mvp.html
[4] www.techcrunch.com/2010/05/20/froyo-android/

[5] www.audiusa.com/us/brand/en/exp/progress/Upcoming_Models/new_a8.html

resources provided by the host. UML can run essentially any application that can be run on the host [9]. To the host kernel, the UML instance is a normal process. To the UML processes, the UML instance is a kernel. Processes interact with the kernel by making system calls, which are like procedure calls except that they request the kernel to do something on their behalf [10].

As of Linux 2.6.0, UML has been integrated into the main kernel source tree. While UML was designed for the x86 processors, it has since been ported to other architectures including IA-64 and PowerPC. For our experiment, we use UML for hosting virtual machines in our testbed. It is also used for kernel development due to its sand-boxing properties.

One example in which UML is being used for large scale network experimentation is ORBIT [6]. *"ORBIT is a two-tier laboratory emulator/field trial network testbed designed to achieve reproducibility of experimentation, while also supporting evaluation of protocols and applications in real-world settings."* UML images are used to encapsulate operating system-specific requirements and configurations. This is to allow for multiple users of the test-bed to use the common hardware easily.

There are some experimental limitations that a researcher must be aware of when using UML images for networking experimentation, especially for scenarios that look at wireless layer effects such as ours. If we look at common metrics used in the characterisation of wireless networks, such as throughput, delay and jitter, we find some work has been done in ORBIT to quantify the impact of using UML. From [20], the following points have been demonstrated and need to be considered for our experiments:

Long Duration

Virtualisation has minimal effects on UDP experiments when the experiments are carried out over a long period of time. Experiments that measure instantaneous throughput are often inaccurate as a result of the virtualised platform, due to a considerable increase in variance of throughput close to saturation (30Mbps).

Packet Size

Virtualisation creates a limitation on experiments that require small packet sizes and high bit rates (less than 30000 packets per virtualisation platform). The recommended packet size used is 1470 bytes.

Other important factors that might potentially skew our results are the impact of running multiple UML instances on a single machine and how they might affect each other. It has been shown in [20], that 2 UML instances running on the same machine have some impact as the packets are buffered for a random amount of time, for the UML to be context-switched, before they are sent over the wireless interface. For our experiment, we have a larger number of UML instances but this is tempered by the fact that we do not have any wireless interfaces and use virtual interfaces. We are also planning on running calibration tests on our virtual testbed to quantify the background effects before we run any experiments.

The delay and jitter results from a virtualised testbed such as ORBIT might also be higher due to the additional UML layer. As stated previously, we ran calibration tests to quantify the average delay and jitter beforehand and adjusted our conclusions accordingly based on the corrected results.

III. Approach

Here we provide a user scenario to explain in greater detail how our approach would work. Imagine that Bob is a construction worker who has recently been injured on the job. After being admitted to hospital, his physicians attend to him, after which they attach sensors to his person in order to monitor his vital signs while he is recovering within the hospital (See Figure 2A). After a week or so, he is deemed healthy enough to return home, but the physicians would still like to continue monitoring his progress. Before he leaves the hospital ward, Bob is issued a set of wireless sensors that form a WBAN. Bob is also given a link to the hospital website, which he can access via his mobile phone to download the VMI of the virtual management module for the WBANs.

At home Bob attaches the sensors and activates the VMI on his mobile phone, which proceeds to detect the WBAN on Bob. The management module then downloads configuration settings for the WBAN and starts collecting/sending sensor readings. With this sync complete, both Bob (See Figure 2B) and his physician (See Figure 2D) are able to monitor the sensors. Because Bob's mobile device also acts as a mobile router, whenever Bob moves outside (See Figure 2C), the existing network connections that constitute the feedback to his physician is not destroyed after handover.

Eventually, once the physician has deemed that Bob is healthy enough to continue without monitoring, an appointment can be made for Bob to return return the sensors to the hospital.

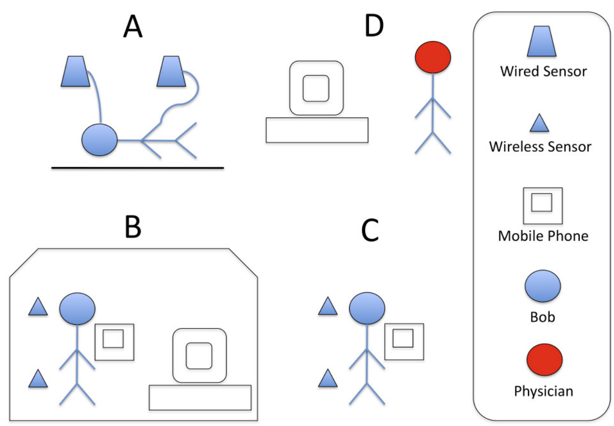

Fig. 2. Example of mobile WBAN application in medical care

[6]www.orbit-lab.org/

IV. Preliminary Results

For managing and manipulating the network level connectivity of our UML images, we used a tool called Cloonix-Net[7]. Cloonix-Net is openly available software that creates virtual networks using UML virtual machines. It consists of two major components. The first component is the Virtual Switch, which creates the virtual LANs required to connect the individual UML machines in a user-defined network topology. This switch receives XML messages to configure all the LAN communications, then switches all IP packets according to the network configuration. The network topology is user defined and stored as a text file. The second component are the Virtual Machines, which are implemented by pseudo-filesystems, made available as UML or KVM machines for the Debian and Fedora operating systems. These Virtual Machines have Ethernet interfaces plugged to sockets that are in turn connected to the Virtual Switch.

Cloonix-Net restricts the user to a single kernel for Debian and Fedora operating systems. Multiple file-systems are allowed but these are used only as references. Any writing to file is saved in a separate file to implement a Copy-On-Write (COW) system. This limitation of one kernel makes it difficult to create a testbed of different network entities. Cloonix-Net allows for multiple kernels and multiple file-systems. The user can create a unique testbed topology consisting of very different machines to suit his needs.

So far, we have built and tested three virtual testbeds. The first testbed consists of a Mobile IP network. The second consists of a NEMO network. And finally, the third consists of a nested mobile network, which is a NEMO network that includes Mobile IP nodes and static nodes.

For the host machine we used a desktop node with the following specifications:

- Intel Dual Core, Pentium 4 2.80GHz
- 1 GB memory
- Running Ubuntu 9.04 Desktop
- Ubuntu 2.6.32.22 i686 GNU/Linux kernel

Each UML virtual machine had the following specifications:

- Linux 2.6.32.8 i686 GNU/Linux
- total diskspace 4.0GB
- 512 MB memory
- NEPL umip 0.4
- kernel size of 27 MB
- Running Debian GNU/Linux 5.0

We have tested handovers in all three of our test virtual networks and found the NEMO and Mobile IP daemons to be functioning correctly. We ran the protocols with security (via IPSEC) turned off to reduce complexity. The reader should also note that the current size of a single virtual image file-system is approximately 4GB. The UML kernel is about 50 MB. Given the current storage space on smartphones today, there is more than enough room to store and execute at least one UML virtual image.

[7]clownix.net

V. Experiment

In order to test our novel approach, we have designed a proof-of-concept experiment. The main idea of this experiment was to show that the application of mobile networking to a realistic medical care scenario using virtualisation tools is feasible.

We leveraged previous expertise with virtualisation (i.e. UML) to create a WBAN using a smart phone running Android for mobile network connectivity. The purpose of this WBAN would be the real-time monitoring of vital signs of the mobile user. A mobile network architecture is used as it allows us to aggregate the various real-time sensors attached to the user that monitor his current state-of-health, as one mobile entity. This makes management of mobility and data less complex and more locally centralised. While we focus on e-health aspects, this idea is applicable to other scenarios, such as military, performance sports as well as recreational, by using communication technologies such as 3G and Wifi to construct a mobile WBAN that is centrally managed by a mobile router, for example.

The mobile router (MR) will exist as a virtual machine that resides within the users mobile phone. The MR acts as a access gateway to static sensors on the user's body, providing constant network connectivity, while on the move. The virtual machine can be moved across to other machines and re-use the sensors available locally. The virtual machine also serves as a management entity for control as well as a repository of data. By encapsulating these aspects within a VMI, we make management simpler.

We have targeted the following three objectives (See Figure 3): (i) to explore the possibility of running a UML machine within the Android OS, (ii) to test the feasibility of using the Android platform to share mobile phone resources, and (iii) to explore the challenges in building a mobile network using UML virtual machines on Android. In the following section, we give a brief description of experiment stages.

A. *Experiment 1a:*

The biggest concern was the usage of virtualisation on the Android platform. We have experience building virtual mobile networks using User Mode Linux (UML), but as yet no work has been done for UML on Android. As a result, our first experiment was to get UML working on an Android smartphone. We know that Android is based on the Linux 2.6 kernel and UML is an integrated part of the kernel.

B. *Experiment 1b:*

The next step was to get a virtual mobile router running the NEMO protocol on an Android smartphone. The virtual mobile router servered as both an access gateway as well as a point of information management for the various sensors. We used existing off-the-shelf wireless heart rate sensors (sport sensor) that used Bluetooth for connectivity to the smartphone.

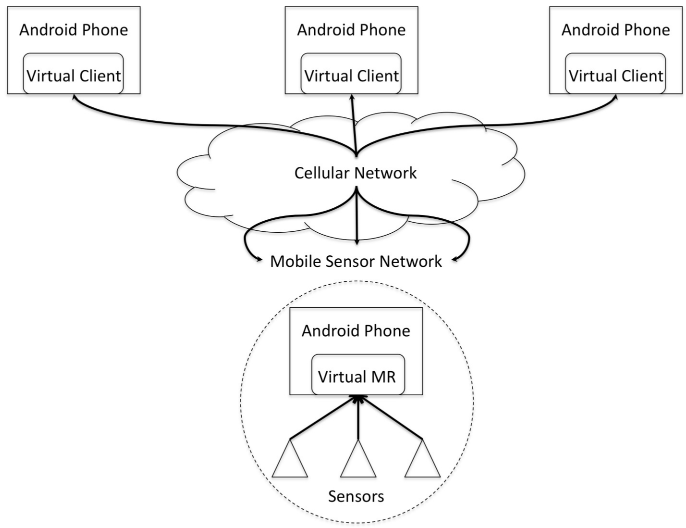

Fig. 3. Wireless Body Area Sensor Network

C. Experiment 1c:

Here we tested the connectivity of the sensor network with the virtual mobile router, while on the move. We checked that the NEMO protocol was able to handoff the connection streams that send the readings of the various sensors across the network. Calibration of the network settings, such as security measures, positioning of sensors and networking protocols (TCP/UDP) were conducted to ensure optimum handovers.

D. Experiment 1d:

The final step was to connect the outgoing sensor streams of information to external repositories across the network. Here we explored privacy concerns such as how the sensors could be read/managed from outside the network without allowing access to personal information on the user's smartphone.

VI. Future Work

The field of virtualisation for mobile devices is still very young. There are many possible applications to virtual mobile WBAN technology. One possible application could be the training of high performance athletes. Sensors used for monitoring health signs are equally valid for the purposes of training. Coaches can make use of this information to add value to existing training regimes as real-time feedback is available. For our particular experiments, we are particularly interested in the mobility aspect of WBANs. We are also especially interested in exploring ways in which to improve network mobility performance.

References

[1] Basuma health project. http://www.basuma.de/.
[2] Cardionet http://www.cardionet.com/.
[3] Lifeguard. http://lifeguard.stanford.edu/.
[4] Lifeshirt. http://www.raesystems.com/products/lifeshirt.
[5] Mobihealth. http://www.mobihealth.org/.
[6] Myheart project. http://www.hitech-projects.com/euprojects/myheart/.
[7] P. Barham, B. Dragovic, K. Fraser, S. Hand, T. Harris, A. Ho, R. Neugebauer, I. Pratt, and A. Warfield. Xen and the art of virtualization. In *SOSP '03: Proceedings of the nineteenth ACM symposium on Operating systems principles*, pages 164–177, New York, NY, USA, 2003. ACM.
[8] P. Brand ao and J. Bacon. Body sensor networks: can we use them? In *M-PAC '09: Proceedings of the International Workshop on Middleware for Pervasive Mobile and Embedded Computing*, pages 1–6, New York, NY, USA, 2009. ACM.
[9] J. Dike. User-mode linux. In *ALS '01: Proceedings of the 5th annual Linux Showcase & Conference*, pages 2–2, Berkeley, CA, USA, 2001. USENIX Association.
[10] J. Dike. *User Mode Linux*. Prentice Hall, 2006.
[11] T. Falck, H. Baldus, J. Espina, and K. Klabunde. Plug 'n play simplicity for wireless medical body sensors. *Mob. Netw. Appl.*, 12(2-3):143–153, 2007.
[12] T. Gao, D. Greenspan, M. Welsh, R. R. Juang, and A. Alm. Vital signs monitoring and patient tracking over a wireless network. In *in In Proceedings of the 3rd International Conference on Information Communication Technologies in Health (ICICTH'05*, 2005.
[13] E. Jovanov, A. Milenkovic, C. Otto, and P. de Groen. A wireless body area network of intelligent motion sensors for computer assisted physical rehabilitation. *Journal of NeuroEngineering and Rehabilitation*, 2(1):6+, March 2005.
[14] P. Kulkarni and Y. Öztürk. Requirements and design spaces of mobile medical care. *SIGMOBILE Mob. Comput. Commun. Rev.*, 11(3):12–30, 2007.
[15] K. Lorincz, D. J. Malan, T. R. Fulford-Jones, A. Nawoj, A. Clavel, V. Shnayder, G. Mainland, M. Welsh, and S. Moulton. Sensor networks for emergency response: Challenges and opportunities. *IEEE Pervasive Computing*, 3:16–23, 2004.
[16] P. Lukowicz, U. Anliker, J. Ward, G. Troster, E. Hirt, and C. Neufelt. Amon: A wearable medical computer for high risk patients. *Wearable Computers, IEEE International Symposium*, 0:0133, 2002.
[17] A. Milenković, C. Otto, and E. Jovanov. Wireless sensor networks for personal health monitoring: Issues and an implementation. *Comput. Commun.*, 29(13-14):2521–2533, 2006.
[18] N. Oliver and F. Flores-Mangas. Healthgear: A real-time wearable system for monitoring and analyzing physiological signals. In *BSN '06: Proceedings of the International Workshop on Wearable and Implantable Body Sensor Networks*, pages 61–64, Washington, DC, USA, 2006. IEEE Computer Society.
[19] B. Quach, M. Balakrishnan, D. Benhaddou, and X. Yuan. Implementation of integrated wireless health monitoring network. In *WiMD '09: Proceedings of the 1st ACM international workshop on Medical-grade wireless networks*, pages 63–68, New York, NY, USA, 2009. ACM.
[20] S. Singhal, G. Hadjichristofi, I. Seskar, and D. Raychaudhri. Evaluation of uml based wireless network virtualization. In *Next Generation Internet Networks, 2008. NGI 2008*, pages 223 –230, 28-30 2008.
[21] S. Smaldone, B. Gilbert, N. Bila, L. Iftode, E. de Lara, and M. Satyanarayanan. Leveraging smart phones to reduce mobility footprints. In *MobiSys '09: Proceedings of the 7th international conference on Mobile systems, applications, and services*, pages 109–122, New York, NY, USA, 2009. ACM.