PAK Publishing Group
Growing Knowledge for Future

Proceedings Book of ICETSR, 2014, Malaysia
Handbook on the Emerging Trends in Scientific Research
ISBN: 978-969-9347-16-0

SIHAT: Simplifying Interfaces in Health-nets for Achieving Telemetry

Qazi Mamoon Ashraf
Department of Electrical and Computer Engineering, International Islamic University Malaysia, Jalan Gombak, Malaysia
Mohamed Hadi Habaebi
Department of Electrical and Computer Engineering, International Islamic University Malaysia, Jalan Gombak, Malaysia
Jalel Chebil
Department of Technology and Engineering in Transport, Higher Institute of Transport and Logistic of Sousse, Tunisia

Abstract

Telemetry in healthcare refers to an autonomic process by which remote measurements of patients are taken and transmitted from non-Internet Protocol based equipment for monitoring purposes. The transmission is through specialized channels of communication using non-standard, proprietary processes. Internet of Things, an expansion of wireless sensor network technologies, aims to standardize the communication methods between such devices. The application of wireless sensor networks in healthcare is referred to as Health-Nets where networked sensors are deployed on a person's body to monitor vital functions. This paper discusses early work for a messaging communication system in an attempt to make Health-Nets compatible with the IoT by simplifying the interfaces. Concerns of data management, addressing, IoT telemetry and the corresponding security requirements are addressed as well. Preliminary observations of performance on a testbed are presented.

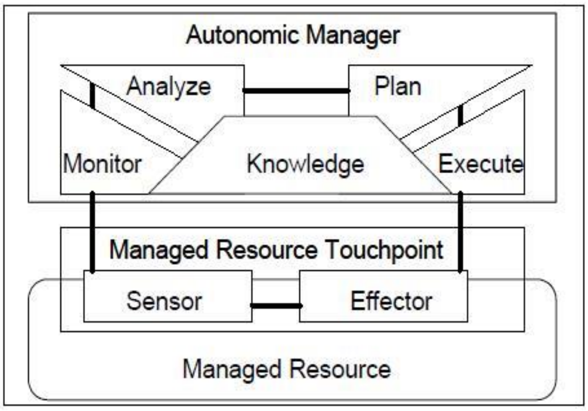
Keywords: Internet of things, Health-nets, Telemetry, Wireless sensor networks, Contiki, Zigbee, Health information management, Internet.

1. Introduction

Wireless sensor networks (WSNs) are widely deployed for the purpose of remote monitoring and comprise of heterogeneous components as well as non-standard interfaces (Rajasegarar, Leckie, & Palaniswami, 2008). A special application of WSNs is in the field of Health-Nets (HNs) or Body-Nets (BNs) where the subject being monitored is a person typically in a hospital environment. The sensor nodes are installed directly on top of the person's body, either as standalone wearable devices or as part of the clothing. These sensor nodes, characteristically employing wireless connectivity, are used to monitor the vital functions of a patient such as body surface temperature, location data, and movement data through accelerometers. More complex sensor nodes are also used to monitor functions such as electrocardiogram (ECG), blood sugar level etc. In the market, some devices are being sold which are capable of monitoring cardiac and hemodynamic functions (Chu-Pak Lau, 2008). The output from these sensor nodes is collected at a central station, typically a mobile cell-phone or any electronic device, where it is gathered, analyzed, forwarded or displayed appropriately. The gathered data is used for patient assessments, education, and family communication. It may even be medically analyzed to predict any future event with a certain probability.

Recently, interfacing in low power devices has been successfully achieved by using standard IPv6 over Low power Wireless Personal Area Networks (6LoWPAN) based solutions (Shelby & Bormann, 2011). Other proposed solutions such as (Gonzalez-Valenzuela, Chen, & Leung, 2010) (Masri & Mammeri, 2007) (We, Bein, & Phoha, 2010) achieve good results and can be suitably modified for use in HNs; particularly those involving middleware deployments to simplify the interfaces. Another solution for interfacing is the layered deployment of middleware to enable Internet Protocol (IP) connectivity through Zigbee as proposed in (Narmada & Rao, 2012). An approach considers mobile middleware agents to achieve the interfacing (Fok, Roman, & Lu, 2009), and one more proposes a solution which also caters to security issues (Usman, Muthukkumarasamy, Wu, & Khanum, 2012). An interesting approach is to introduce connectivity in medical sensor devices by coupling with Zigbee based technology; interconnected with a bridge networking method (Jung, Chang, & Gerla, 2007). Zigbee is based on IEEE 802.15.4 technology for establishing low rate personal area networks which defines the physical and MAC layers with star and peer-to-peer topologies support. It defines the network layer specification extending 802.15.4 functionality to tree and mesh based topologies (Digi International Inc.). On the other hand, Bluetooth based architectures have also been used because of low-power and low-cost requirements, to standardize interconnections between different devices. However, Bluetooth may not be suitable because of its non-reliability in communication, low range and the complex method of enabling connectivity. To address problems of delay, data delivery and power consumption in Bluetooth HNs, a new protocol with features of simple pairing and inquiry response messaging has been proposed in (Lee, Jung, Chang, Cho, & Gerla, 2008). These approaches, although enlarging the scope of abilities to interface, deliberately adopt a loose approach to maintain generality towards multiple target environments. In general, there is not much innovation and effort needed to deploy a HN, except for careful provisions regarding the data format (Hesse, 2012) and the critical nature of timely readings.

Fig-1. IBM Autonomic Control Loop (Kephart & Chess, 2003)



The remainder of this article aims to discuss reasons to connect a HN to the Internet of Things (IoT). Issues such as data management, addressing, and the security requirements that surface are discussed. Related work motivating for telemetry in IoT is presented in the second section. This is followed by the system description of the suitable setup. The fourth section presents the experimental setup and fifth section highlights some initial performance measurements.

2. Telemetry and IoT

IoT is defined as a vision to connect the planet and all the things on it by standardizing the methods of communication. It is a platform where traditional Internet is being extended to include heterogeneous objects other than the traditional computers (Iera, Floerkemeier, Mitsugi, & Morabito, 2010) (Zheng, Simplot-Ryl, Bisdikian, & Mouftah, 2011). The core idea leading to this vision suggests that devices, capable of presenting some sort of unique identification, with communication and sensing abilities, should be IP enabled. IoT is a thus a combination of embedded smart objects, sensors and web-based services, all connected over the Internet (Shelby & Bormann, 2011).

One prerequisite for IoT is that the components should achieve organizational structure without human intervention. In this regard; the aims of IoT and telemetry are similar. Both require automatic processes to manage their components; IoT because of its large size of scalability issues, and telemetry because of remotely located components. In both these cases, manual maintenance is not an option. Both cases require that "*setup*" must occur automatically, as well as dynamic modifications to constituent configuration should occur to best handle changing situations. Telemetry aims to achieve organization of these IoT smart objects by autonomic and self-star principles. The aim of these principles is to reduce the need for manual intervention and management at every level as much as possible.

A telemetric system must arrange and reconfigure itself under varying and unpredictable conditions. Dynamic system configuration must occur automatically and without any human intervention. A scenario arises when smart objects such as those belonging to IoT are allowed to join networks and both offer and consume services. For low power devices, such smart objects are usually equipped with a set of configuration information already installed. On the event of moving into a different network, it is required to configure them again with new network parameters catering to the new requirements, and keeping the interfacing common for both the cases. For a HN, it will be efficient if the medical sensor nodes have some kind of mechanism to deal with this automatically. A dynamic mechanism to solve the problem of scalability as well as to optimize any particular service is needed. Thus in this context, telemetry deals with new medical sensors joining, exiting and the complete system reconfiguration and management as a result of such a topology change.

In general, a telemetric network refers to an auto-configuring, autonomic network. IBM (Kephart & Chess, 2003) has introduced a framework of autonomic computing which is aimed to make the management of systems easier. The framework consists of two entities: a 'Managed Resource' and an 'Autonomic Manager'. **As seen from Fig. 1,** the managed resource is the less complex entity and comprises of sensors and effectors, which the architecture recommends to be linked with each other. They together make the 'manageability interface' which is available to the autonomic manager. The autonomic computing concept revolves around what is called as a control loop which is implemented by the autonomic manager component. However, the control loop is more like a structural arrangement than a control flow.

3. System Description

HNs are characterized by small area of deployments which will typically cover less than 1.5 square meters of space. We do not require the sensors to have long ranges of transmission; just enough to reach to one or a few neighbor nodes, at maximum the data sink. There may also be a provision to dynamically adjust the transmission power based on RSSI values such as done in (Habaebi & Elashaal, 2011) for Zigbee based HNs; suitably modified for a single hop based setup. The sensors will be suitably configured to take readings after a given amount of time. This configuration may be done manually or by some suitable method of auto-configuration. One important distinction between a HN and other networks in IoT, is that data in a HN will be mostly unidirectional i.e. data will mostly originate and sent from the HN. There is very little scope for actuators being defined in a HN which will act appropriately on a signal from person in-charge such as a doctor or by any automatic procedure. Only in some cases, actuators may be enabled e.g. to give automatic insulin shots upon blood sugar exceeding some particular threshold.

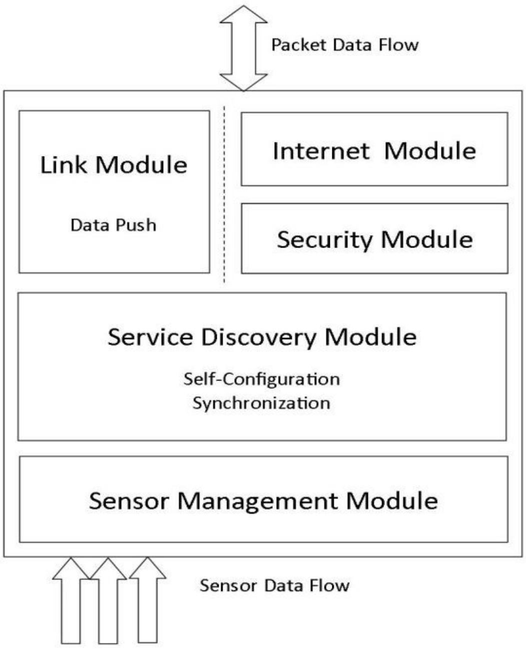
Many traditional methods employed for personal HNs focus on mobile cell-phones as central devices to gather, manage, store and process the sensor data (Barnickel, Karahan, & Meyer, 2010). In other cases, the mobile cell-phone may be used as an intermediate router to decide what to be forwarded to another server and does not take part in any data analysis. A weakness manifest in these approaches is that these mobile cell-phones, unlike the highly durable sensor mote devices, run out of energy very fast due to the multitude of capabilities and uses. Also, there is the assumption that the mobile phone will always be nearby the HN. This approach is very risky as it is a common phenomenon for the cellphones to run out of battery and misplaced. The advantage of using mobile phones is high for home monitoring by families. A cell-phone should not be used as a data sink but instead just as a user interface to view the data and the reports. A sensor mote device should be dedicated entirely for this purpose of interfacing and thus act as a gateway. The capabilities of this node are not just restricted to being a data sink, but also as an interface to translate messages from the sensor nodes into a suitable format to be forwarded to the appropriate web service. To save on the number of transmissions, multiple sensed data is sent in one packet. This greatly affects the performance of the HN. The fact that the transmission radius is also very small helps in preserving energy as well. This emphasizes the feasibility and flexibility of the messages that have been constructed.

An additional distinction between HNs and other conventional networks is that HNs are highly mobile networks such as those defined in the network mobility protocol (NEMO) (Devarapalli, Wakikawa, Petrescu, & Thubert, 2005) where the whole setup of nodes can attach to one access point and then to the next. For these single-hop requirements, a solution similar to RIoT can be beneficial for the node registration (Ashraf, Habaebi, Sinniah, & Chebil, 2014). Yet, an important factor to take note is the criticalness involved and the attempts towards risk management. Obviously, if the central device fails, then the whole sensor network for the patients goes down with it. To cater for that requirement it is possible to deploy multiple data sinks, allowing for failure of one without any major effect on the HN. This is similar to the failover concept applied by cellular communication systems.

A. Node Interfaces

Fig. 2 shows the proposed modular component diagram in a HN node which consists of three required modules and two optional modules. The figure also shows the two interfaces with the environment, one to cater for incoming sensor data flow, and the other one to interface for the packet data. The sensor management module deals with frequency of sensing from the medical sensors based on case by case basis. In some cases, a constant frequency of sensing, whether low or high, may be required such as hourly body surface temperature or daily blood sugar levels. In other cases, specific sensing activity may be triggered by other elements such as triggering power-consuming heart rate sensor after detecting high fluctuations in motion using an additional low power accelerometer based sensor. The service discovery module is employed to assist in configuration related activities to be done automatically. This is introduced to impart a level of telemetry into the framework. The service discovery module consists of functions such as configuration and synchronization. Traditional IoT architectures are service-resource based and such a module may be used to interface directly to any IoT service provider. To interface the data between the body sensors which might not support any IP-

Fig-2. Module Component Diagram in a HN Node



based communication, an essential Link Module is provided for low level data transfer. For nodes which may support IP based communication, the optional Internet Module is activated. The security module has been included to address privacy issues and may be used to implement trust based or encryption based systems.

B. Network Overview

A data sink is essential in the architecture and should be able to support IP as well as non-IP based communication with the HN nodes. The sink will have to be constantly in the listening mode, and thus the physical requirements of a sink are higher than that of a sensor node. Every data originated in the HN has to be sent to sink. For readers familiar with existing sink-based frameworks, it will be apparent that a sink is a central point of failure, and it may not be a good idea to forward all the traffic through the sink. Thus, we are using two sinks in this approach, which will switch functionality after given intervals. This approach reduces the probability of failure of the HN by two times as one sink will have the ability to replace in case the other fails. Also, energy efficiency per node is improved, as the duty cycle of each sink will now be half of the actual value. For simplicity in discussion, for the remainder of the article we consider both the sinks as one virtual sink.Assuming the case setup of a hospital, access points can be provided in a manner to result in full coverage of the hospital campus based on average transmission radius of the sink. Thus, it will be very unlikely that a patient with a HN installed will ever disconnect from the network. Obviously, the sink has to be attached to the patient's body so that no sensing activity is lost. The network would resemble Figure 3.

C. Addressing For Sensor Nodes

For addressing, IPv6 may be preferred for identifying IP-based nodes but it solely depends on the physical hardware capabilities of the HN. For nodes supporting IP-based communication, the Internet Module will be activated and generate the node's address suitably. It may be done by including the node MAC addresses itself into the IPv6 such as in stateless address auto-configuration (Narten, Draves, & Krishnan, 2007). For non-IP based nodes, RFID/QR codes may also be used to '*identify*' things instead of IPv6, and the address functionality will lie in the Link Module. IPv6 will definitely support any additional schemes such as implementing temporary IP addresses to protect privacy, ability to change this identifier, as well as improved efficiency. Efficiency will be apparent as less overhead will be involved in continuous address translations that may happen in the case of other identification/addressing technology being used.

D. Security Requirements

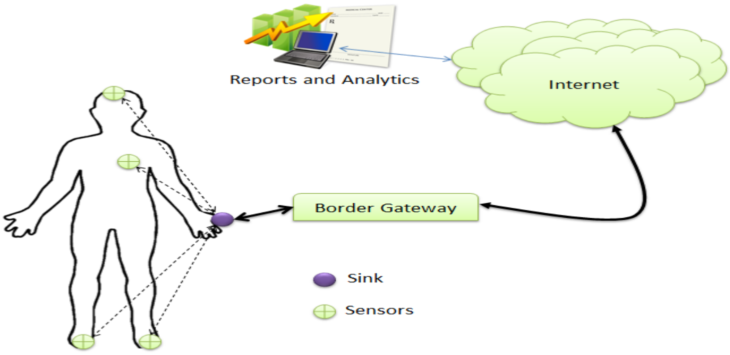
Security not only encompasses privacy, but also availability and integrity. For recovery, the HN and in fact another IoT stub-network would need some routing capability, as a backup method to route the data. So instead of just behaving as an add-on to the existing infrastructure of the Internet, we should try to make it as self-sufficient as is practical. Thus, even if the networks are not addressed by IPv6 per se, but some ability to understand IPv6 (perhaps extended by some auto-configuring service) will definitely be an advantage. Obviously, relying too much on IPv6 will also be fatal; thus motivating the need for inherently link layer based communication abilities between the sensors and the sink.

Following this framework, data may be read without the patients explicit permission during an emergency scenario such as when authorized by a medical expert. For reasons of making the interface simple, security mechanisms such as authentication have been deliberately avoided. But if the need arises, not much work is needed to secure this framework. All data which is to be stored in the cloud, can be transmitted though by secured channels and only allowed for read capabilities. Data stored on the cloud should be only read by authorized persons, e.g. the family members of the patient, and specific staff of the hospital.

E. IoT Resource Detection

Services and resources can be discovered using Web Services Description Language (WSDL), or by other implementations such as Simple Object Access Protocol (SOAP). A node can ask the AP about the available services, and an AP can hand down appropriate services based on the capabilities of the attached sensor node. To find the appropriate services, detailed information may be requested

Fig-3. A single-hop star configuration with one Sink. The sink connects to a border gateway interface to the Internet.



form the sensor node and information such as sensing abilities, protocol versions, remaining energy or information about memory requirements can be used for decision making. Generally, enabling services require heavy manual configuration and include a large number of parameters. A third party IoT service provider may interact with the AP, and the framework can be suitably modified to accept messaging data into the sensor node. The services can then be suitably accepted or rejected by the node.In the context of a HN, services may refer to sensor based functionalities where a single mote device may be capable of sensing multitude of parameters. Services would then refer to a particular set of parameter being sensed, as well as the duration between consecutive sensing activities. In conjunction with IoT service providers, HNs can provide a powerful example of IoT services being beneficial to the humanity.

4. Experimental Setup

Zolertia Z1 devices were used as sensor motes because of their simple networking capabilities including support for IEEE 802.15.4 standard, with limited processing power. They contain few on-board sensors viz. TMP102 temperature sensor and ADXL345 accelerometer sensor which were used to check for body temperature as well as motion related information. To start off with, we put together a basic testbed to investigate the working of the architectural prototype. Two Z1 sensor nodes were used to generate sensor data based on temperature, and then build a suitable message packet with the new measurements. The nodes would then send it over the wireless interface to the sink, which temporarily stored the data in a buffer and then suitably modified the data packet before transmitting to another Z1 node representing an Access Point for simplicity in implementation. Had we simulated the framework, as opposed to testing it on real nodes, the results would have been similar, with some slight modifications to the overhead that is generated on a testbed.

We implemented the message building method directly in the Z1 nodes, which were used to

Table-I. Test Bed Hardware Specification

Components	Description
Model	Zolertia Z1 WSN mote, USB powered
MCU	MSP430F2617 16-bit RISC CPU@16MHz
RF transceiver	CC2420 2.4 GHz @250kbps Spreading gain: 9dB 128(RX) + 128(TX) byte data buffering (Texas Instruments, 2013)
Memory	92 KB + 256B Flash Memory, 8KB RAM
Routing Protocol	RPL- Single Hope
Identification	IPv6 based
Embedded OS	Contiki

Fig-4. Listening Duty Cycle for a Sensor Node as compared to the Sink

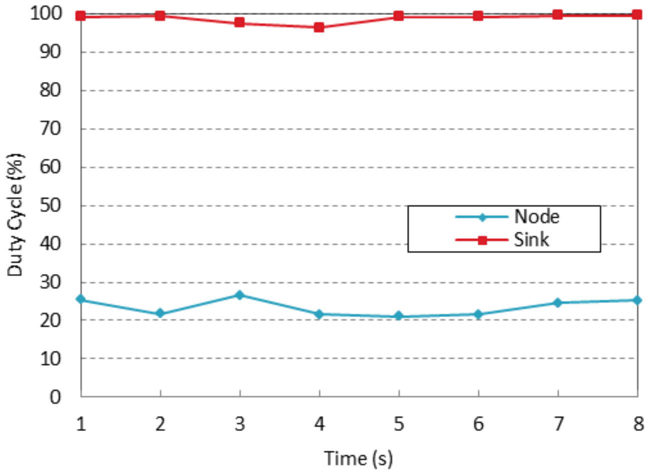


Fig-5. Average Power Consumption for a Sensor Node as compared to the Sink

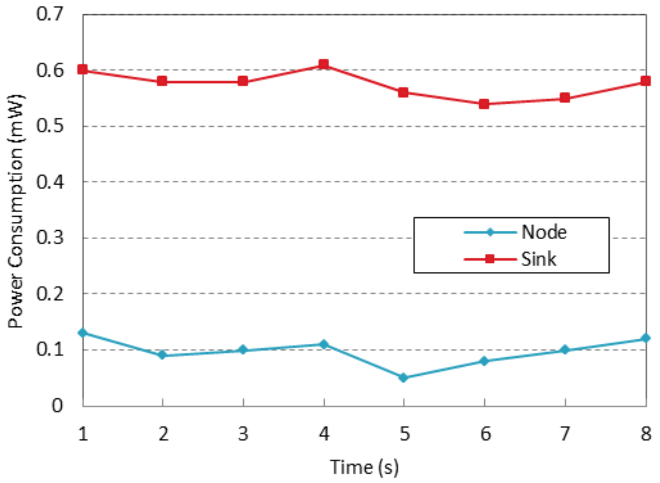
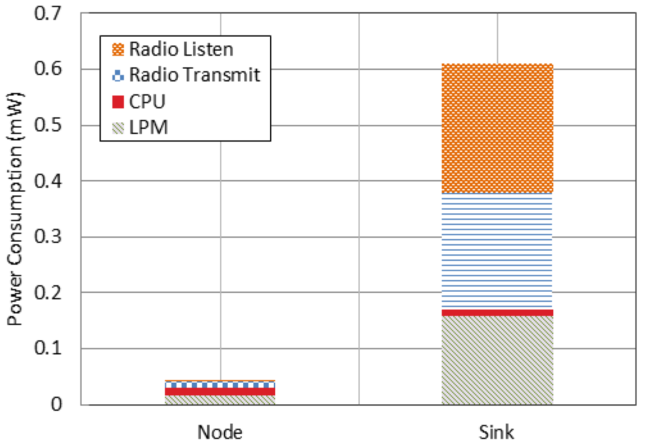


Fig-6. Breakdown of the Average Power Consumption for a Sensor Node and a Sink



represent the traditional medical sensors. For the sink, 100% listening radio duty cycle was enabled to minimize the packet loss, and a function was introduced to analyze the incoming message, strip the header and retrieve the data. Contiki was used as the Operating System (OS) and the programming was done in C. Table I describes the hardware specification for the HN test bed setup.

5. Performance Evaluation

We evaluate the performance of the sink as compared to the performance in the sensor node. Our performance measurements represent results to analyze the factor of energy consumption. For determining the same, a total of 36 packets were sent from a sensor node to the sink, where they were stored. In our setup, these packets contained temperature measurement data from a Z1 node.

The CPU usage to take the temperature measurement, construct the message, and then transmit the message is significantly low as compared to the CPU usage in the sink. Furthermore, transmission utilizes more energy than simple calculations in the CPU. It is observed that the energy consumption of the sink remains much higher than a sensor node. The following metrics are presented for energy consumption:

1. Low Power Mode (LPM): LPM refers to overall power consumption in the low power mode, also known as the sleep mode where not much activity takes place inside the sensor mote. Radio activity is completely cut off.
2. Central Processing Unit (CPU): CPU section of the graph shows consumption of energy by CPU based activities.
3. Radio Transmit: It is the amount of energy spent overall for total radio transmission.
4. Radio Listen: It is the amount of energy spent in the radio listening mode.

Interesting thing to observe is the LPM power consumption of the sink is large as compared to the CPU. This is because of the time that the sink spent in the LPM mode is significantly high as compared to the time the CPU spent in calculations. Figure 4 displays the radio duty cycle that was implemented in the node and the sink for radio listen. The sink remained at roughly 100% listening duty cycle throughout the experiment, whereas the duty cycle for a sensor node is much lower in comparison. The sensor node will only utilize the radio during its short bursts of transmission which will be responsible for a majority of its current expenditure.

Figure 5 compares the average power consumption arisen from a wide variance in the duty cycle in the sink and an end node. Clearly the energy requirements of a sink are far higher than the sensor node. A breakdown of this energy consumption reveals the biggest factor in energy consumption. Figure 6 explains the current consumption inside of the two nodes, and demonstrates that radio transmission is responsible for the highest consumption, followed by LPM mode, then CPU then Radio Listening mode.

6. Discussion

Achieving functionality and maximizing the lifetime of individual sensors are conflicting goals. Also, HNs aren’t truly remote as they as always attached to a person and replacing batteries is not an issue, but timely readings and an appropriate frequency of sensing is an important requirement. In addition, requirements in HNs deal with the medical sensors being capable of monitoring vital signs, being attached to the body with reduced data footprints and the being able to analyze the data (Kranen, et al., 2008). The most important requirement would be to limit the size of the data in an effort to preserve battery, and increase efficiency, as compromises may not be made in the duty cycle which reflects on the frequency of the readings made. The second method to preserve battery is to limit the transmission radius to a meter or so. Complex middleware implementations have been avoided, and focus is laid on a simple method for the medical sensors to interact with the central device, referred to as a data sink.

A. Data Interfacing

Clearly the requirements of a sink and sensor node vary significantly. A sink node will generally not have any sensing functionality and will be always in the listening/reception mode. The radio duty cycle numbers for the HN as a whole have been shown earlier. Instead of the basic message format that we employed in demonstration on our testbed, the collected data can be forwarded in a suitable format such as XML, JSON or CSV to a remote web service. Once the sinks collect data for a certain period of time or until a certain portion of their allocated memory has been used for the same, they will forward the data to the access point/border device that they are attached to. Thus, all sensor nodes will be tied to the sink, while as the sink will be responsible to interface to the outside world via an access point or a device such as a border router.

The sink may also use application level protocols such as MQTT, or REST to push the data into an appropriate web-service such as Twitter or cloud service providers such as Xively (Xively). If the sink is unable to connect to any network then it will keep on storing the data until it finds a network to connect to. If the sink is unable to find an access point, while its memory is running out, then the sink will prefer to store fresh data as opposed to the oldest data which will be discarded. Whereas this does simplify the working significantly and assumes a safe scenario, but work is definitely needed to brush the algorithm to cater to certain critical data for this kind of digital forgetting.

The problem whether to save and archive all the data is up for question. A list of privacy concerns and a proposed architecture have been documented in (Barnickel, Karahan, & Meyer, 2010) and its authors recommend keeping no central data storage. However, in the case of a HN, this will not enable long term monitoring and thus being able to draw inferences for effects of suitable medications or physiotherapy routines will be impossible. Thus, the ability to store information about the monitored body data is essential particularly when it comes to diseases requiring constant monitoring such as CF (Ramsey, 1996) or Diabetes (Klonoff, 2005). Actual data may have a wide variety of requirements, but the message format for delivering sensor data in the demonstration just dealt with the value of the temperature parameter and the time. The high frequency of sensing such information may prove to be burdensome when it comes to storing medical records over a long period of time.

B. Single Hop Routing Constraint

The routing method employed was single hop based where the sensor nodes directly communicated with the sink. The reason being that the nodes are not required to be transmitting at ranges of the order of tens or hundreds of meters, as the area of deployment is very small. Reducing range to approximately a single hop resulted in saving in power to the extent that LPM consumed more power than actual CPU activity. We chose single-hop based routing to result in a star topology. The shorter transmission radius is achieved by configuring the transmission power of the sensor nodes. As mentioned earlier, star topology is inherently supported in IEEE 802.15.4 and thus there was no trouble in setting up the network for the same. The sink node will, however, have to run on full reception to have the ability to listen to all the sensor devices and not compromise on reliability of the HN as the sensor devices may forward the data randomly. Furthermore, the sink also needs to transmit at the highest power level it can afford to make sure the access point is able to receive its message. It’s more efficient to transmit a packet with the highest power than to try to transmit and increase power slowly until the AP gets the message and sends an acknowledgement.

C. Suitability of IPv6 Addressing

As has been mentioned earlier, there are two ways for addressing the end sensor nodes. The communication between the sensor nodes and the sink could have been done using functions at the link layer, and then forwarding the information to the AP at the Internet layer, with suitable IP information. Second way is to address all the nodes in the sensor network by IPv6 addresses. The suitability completely depends on whether the HN is to be considered private and isolated, or private to what extent. If the medical sensors are required to be kept locally accessible, such as by putting these IoT devices behind a firewall and Network Address Translator (NAT), the benefits of IPv6 may not be so apparent. In other cases, it may be required that the data be directly addressable on the public Internet and openly interacting with web services. In those cases IPv6 will obviously help and may supplement the security measures positively. IPv6 support may not address failover requirements but it is well suited for the IoT environment and is most probably the best candidate for a full-fledged IoT deployment.

D. Optimizing the Algorithms

It is impossible to guarantee that sensors will be installed on the patient’s bodies at all times. It would be extremely inefficient if sensors wake up and collect data around them after predetermined intervals, or even randomly; when the patients are not around. For implementing telemetry, a suitable method such as a proximity sensor should be used to determine the presence of a person. The HN can be suitably activated or deactivated such that the HN doesn’t start collection of sensor data even when the patient is not wearing the sensors. The proposed sensor management module can cater to these requirements.

In the long term, continuous monitoring gives a better insight towards deciding appropriate medications and other medical procedures. Sensors can also be used to monitor general health status and effects of exercises on diseases such as cystic fibrosis (CF) where motion sensors can act as clinical assessment tools (Bradley, Kent, Elborn, & O'Neill, 2010). The information exchange between patients, nursing staff and doctors as well the general hospital infrastructure is made possible by the streaming of data in this system to a central location where it is aggregated and analyzed. This kind of central design is important to maintain a level of control on the HN. Round the clock monitoring is also possible with automatic generation and logging of events at the central device which may prove to be important indicator in future medications and diagnosis. Critical events can be detected by using classification algorithms or by triggering a warning when a certain parameters exceed its normal threshold. This will have to be done on the web-service where any appropriate method to warn the doctors, or relatives can be implemented.

7. Conclusion

To achieve interconnection of HNs and IoT, HNs are required to have suitable IP interfaces. We discussed requirements in a messaging scheme for carrying real-time data from HNs over the IoT, such as using dedicated sensor motes instead of mobile phones as central data sinks. To provide modularity for interfacing, a messaging communication system was discussed. We also discussed requirements for medical data management, and device addressing. Preliminary tests for duty cycle performance and power consumption suggest future applications.

8. Acknowledgements

This research was supported by the Ministry of Higher Education Malaysia (MOHE) under Exploratory Research Grant Scheme (ERGS) number ERGS13-023-0055.

References

Ashraf, Q. M., Habaebi, M. H., Sinniah, G. R., & Chebil, J. (2014). Broadcast based registration technique for heterogeneous nodes in the IoT. International Conference on Control, Engineering & Information Technology (CEIT '14). Sousse.

Barnickel, J., Karahan, H., & Meyer, U. (2010). Security and privacy for mobile electronic health monitoring and recording systems . IEEE International Symposium on World of Wireless Mobile and Multimedia Networks (WoWMoM)). Montreal, QC.

Bradley, J. M., Kent, L., Elborn, J. S., & O'Neill, B. (2010). Motion sensors for monitoring physical activity in cystic fibrosis: what is the next step? Physical Therapy Reviews, 15(3): 197-203.

Chu-Pak Lau. (2008). Hemodynamic sensors in heart failure devices. In Devices for Cardiac Resynchronization- Technologic and Clinical Aspects. Springer US. pp: 253-268

Devarapalli, V., Wakikawa, R., Petrescu, A., & Thubert, P. (2005). Network Mobility (NEMO) Basic Support Protocl. Retrieved August 2, 2013, from http://tools.ietf.org/html/rfc3963

Digi International Inc. (n.d.). ZigBee Wireless Standard- Technology. Retrieved September 1, 2013, from http://www.digi.com/technology/rf-articles/wireless-zigbee

Fok, C. L., Roman, G. C., & Lu, C. (2009). Agila: A mobile agent middleware for self-adaptive wireless sensor networks. ACM Transactions on Autonomous and Adaptive Systems (TAAS), 4(3), Article No. 16.

Gonzalez-Valenzuela, S., Chen, M., & Leung, V. C. (2010). Programmable middleware for wireless sensor networks applications using mobile-agents. Mobile Networks and Applications, 15(6): 853-865.

Habaebi, M. H., & Elashaal, Z. A. (2011). Transmission power consumption management for zigbee healthnets. In Research Issues in Wireless Communication Networking. Kuala Lumpur: IIUM Press. pp: 300-305

Hesse, J. (2012). Roving Reporter: An intelligent framework for connecting the Internet of (medical) Things. (Intel) Retrieved July 4, 2013, from http://embedded.communities.intel.com/community/en/applications/blog/2012/12/14/roving-reporter-an-intelligent-framework-for-connecting-the-internet-of-medical-things

Iera, A., Floerkemeier, C., Mitsugi, J., & Morabito, G. (2010). Special issue of the internet of things. IEEE Wireless Communcations, 17(6): 8-9.

Jung, S., Chang, A., & Gerla, M. (2007). Comparisons of zigbee personal area network (PAN) interconnection methods . 4th International Symposium on Wireless Communication Systems (ISWCS). Trondheim.

Kephart, J., & Chess, D. (2003). The vision of autonomic computing . Computer, 36(1): 41-50.

Klonoff, D. C. (2005). Continuous glucose monitoring roadmap for 21st century diabetes therapy. Diabetes care, 28(5): 1231-1239.

Koehler, J., Giblin, C., Gantenbein, D., & Hauser, R. (2003). On autonomic computing architectures. IBM Research, Zurich Research Laboratory.

Kranen, P., Kensche, D., Kim, S., Zimmermann, N., Quix, C., Quix, C., . Leonhardt, S. (2008). Mobile mining and information management in healthnet scenarios . 9th International Conference on Mobile Data Management. Beijing.

Lee, S.-H., Jung, S., Chang, A., Cho, D.-K., & Gerla, M. (2008). Bluetooth 2.1 based emergency data delivery system in healthnet . IEEE Wireless Communications and Networking Conference (WCNC). Las Vegas, Nevada.

Masri, W., & Mammeri, Z. (2007). Middleware for wireless sensor networks: a comparative analysis. IFIP International Conference on Network and Parallel Computing Workshops, NPC Workshops. pp: 349-356.

Narmada, A., & Rao, P. S. (2012). Zigbee based WSN with IP connectivity. International Conference on Computational Intelligence, Modelling and Simulation (CIMSiM). Kuantan.

Narten, T., Draves, R., & Krishnan, S. (2007). Privacy extensions for stateless address autoconfiguration in IPv6. Retrieved February 19, 2014, from https://tools.ietf.org/html/rfc4941

Rajasegarar, S., Leckie, C., & Palaniswami, M. (2008). Anomaly detection in wireless sensor networks. IEEE Wireless Communications, 15(4): 34-40.

Ramsey, B. W. (1996). Management of pulmonary disease in patients with cystic fibrosis. The New England Journal of Medicine, 335(3): 179-188.

Shelby, Z., & Bormann, C. (2011). 6LoWPAN: The Wireless Embedded Internet. John Wiley & Sons.

Texas Instruments. (2013). 2.4 GHz IEEE 802.15.4 / ZigBee-ready RF Transceiver. Retrieved August 15, 2013, from http://www.ti.com/lit/ds/symlink/cc2420.pdf

Usman, M., Muthukkumarasamy, V., Wu, X.-W., & Khanum, S. (2012). Securing mobile agent based wireless sensor network applications on middleware. International Symposium on Communications and Information Technologies (ISCIT). Gold Coast, QLD .

We, Y., Bein, D., & Phoha, S. (2010). Middleware for heterogenous sensor networks in urban scenarious. ITNG, Seventh International Conference on Information Technology: New Generations. pp: 654-659.

Xively. (n.d.). Xively- Public Cloud for the Internet of Things. (Log Me In) Retrieved August 22, 2013, from http://www.xively.com

Zheng, J., Simplot-Ryl, D., Bisdikian, C., & Mouftah, H. T. (2011). Ed., The internet of things. IEEE Communications, 49(11): 30-31.