Rapid Prototyping of a Wireless Sensor Network Gateway for the Internet of Things Using off-the-shelf Components

C.P. Kruger *, A.M. Abu-Mahfouz * and G.P. Hancke †
*Advanced Sensor Networks Research Group, Council for Scientific and Industrial Research, South Africa.
†Department of Computer Science, City University, Hong Kong, China.
Email: ckruger@ieee.org, A.AbuMahfouz@ieee.org and ghancke@ieee.org

***Abstract*—More than 50 billion devices are estimated to be connected to the Internet by 2020. Incompatibility of devices and protocols (usually proprietary devices and protocols) are one of the major hurdles to be overcome to realise the Internet of Things (IoT) vision. IoT devices are typically constrained devices and this creates dependencies between the hardware, software and protocols used in the device. Open hardware and software platforms to support emerging IoT trends are required. A wireless sensor network gateway was developed in a three-month period using off-the-shelf components. The gateway was based on the Raspberry PI single-board computer; it implemented 6LoWPAN mesh and wireless access point functionality for mobile and low power sensing and actuation devices. The gateway was tested in the following use-case: integrate a battery-operated 6LoWPAN-enabled smart water meter to an IPv6 building network. Several factors that influence the gateway's performance and reliability were identified and should be considered when deploying gateway devices for future endeavours.**

***Index Terms*—Wireless Sensor Networks; Internet of Things; 6LoWPAN; Gateway; Commercial Off-the-shelf (COTS).**

I. Introduction

The Internet of Things (IoT) has the potential to be one of the most disruptive technological advances to date with the possibility of billions of devices connecting and exchanging information on the Internet. More than 50 billion devices are estimated to be connected to the Internet by 2020 [1]. Rapid prototyping is a key requirement to realise and understand the key challenges relating to the IoT vision [2]. Current IoT technologies can be explored from three main perspectives: scientific theory; engineering design; and the user experience [1]. The latter two groupings rely heavily on the construction and implementation of real-world, deployable hardware and software components to build Wireless Sensor Networks (WSN's), Radio Frequency Identification (RFID) and Machine-to-Machine (M2M) systems for IoT applications. Proprietary designs and protocols for IoT systems are a major hurdle to the proliferation and uptake of IoT systems in practice. There are many differentiating factors of WSN devices, e.g., performance, debugging support, cost, power consumption, and form factor [2]; these could all be addressed by proprietary designed systems in ways incompatible to the IoT vision of heterogeneous devices interconnected into scalable networks. Proprietary protocols are a major drawback in WSN deployments [3] due to the difficulty with which such components integrate into larger systems. Proprietary system designs are not only difficult to integrate initially but also remain a problem during the entire life cycle of the system where routine maintenance may be required in the deployed phase. This paper presents a novel WSN gateway device that could be used to research and implement the engineering design and user experience aspects of emerging IoT trends in an open and customisable manner. The gateway was designed using commercial off-the-shelf components (COTS), ensuring a rapid turnaround time and modular implementation that could easily be adapted for varying deployment use-cases without the limitations of a completely proprietary system. Automation is a key aspect of IoT systems needed to create more efficient cities of which smart water and electricity are key focus areas [4]. An initial use-case for the gateway device for a water meter sensing application was developed. Additional use-case applications of the gateway could also include conductor theft monitoring, detecting sagging transmission lines as well as assist with general fault detection for smart grid applications [5].

II. Related work

Internet protocol (IP) functionality in a WSN can either be introduced at the sensing node itself or at the gateway device using a proxy; these two main approaches for internet connectivity will merge into one single solution [6] where Plug-and-Play operation of sensor networks is a key requirement to realise IoT trends. A gateway using multiple communication interfaces is presented in [3]. The device has the capability to support Wi-Fi, GPS and Ethernet based network connections and uses a full Linux operating system. The operating system is executed using an Intel Atom processor with a Mini-ATX motherboard that can run from a 12 V supply. The device is however not designed to be operated in an industrial environment and thus may suffer from reliability issues. Size, cost and energy consumption optimisations can also be done to improve the solution. An example of a portable gateway for health applications [7] uses Bluetooth connectivity to communicate with other devices over a short distance. The custom nature of the portable gateway device is not well

978-1-4799-7800-7/15/$31.00 ©2015 IEEE

suited for hybrid designs where one gateway connects multiple sensors for different use-cases. An IoT gateway [8] based on Zigbee and GPRS technology is used to implement a flexible device with the ability to host an application server based on Linux and Python. The gateway uses an ARM9 processor on a custom designed printed circuit board (PCB). A novel feature of the device is the modular design that was used to physically separate the processor, Zigbee and GPRS hardware in the form of modules. The system however still makes use of AT commands to communicate with the Zigbee network, requiring self defined protocols and glue logic software to be implemented. Another example of a Zigbee to IP gateway is given in [9] where automatic address translation from IPv4 to Zigbee node number is performed using proxy software running on the gateway hardware exposing user configurable settings via a web interface. The addition of 6LoWPAN based connectivity can be used to connect existing protocol-specific sensing devices [10] to the internet allowing for easy integration and deployment. Easy access and maintenance can allow complex environmental sensors to become mainstream by reducing implementation cost. In addition to protocol-specific sensors, existing 802.15.4 based sensors [11] can be easily adapted to function as a 6LoWPAN device by only making firmware changes, since the selected standards are based on the 802.15.4 physical and medium access control layers. Developing a sensible and low cost gateway for the IoT using standards based communications interfaces can thus greatly assist in promoting heterogeneous sensing through unified standards. When interpreting the body of knowledge found in the related work cited in this paper, one can make the following conclusions on requirements for gateway designs.

- The ability to transparently bridge a WSN with the internet is the most important property of a gateway device.
- It should be able to integrate various heterogeneous sensor network technologies where legacy equipment still exists.
- It should provide management, data off-loading and routing services for constrained devices where full function device (FFD) capabilities are not practical.

All gateway devices can be broken down into hardware and software components forming the two key components of any gateway design [12]. The designer thus needs to make a decision as to how the key requirements will be implemented and whether or not the functionality will reside in hardware or software.

III. Gateway design

The main function of the gateway device is to allow data to flow from one side of the gateway to the other without breaking end-to-end IP connectivity between servers and end-devices. The logical design of the gateway device is shown in Fig. 1 with the Local Area Network (LAN) connectivity on the left and the wireless connectivity on the right. The network layout of the system is shown in Fig. 2. Wireless connectivity consists of an 802.11 (Wi-Fi) as well as 802.15.4 (6LoWPAN mesh) interface. The Wi-Fi access point is used to connect Wi-Fi enabled devices including laptops, tablets and smart phones to the internet and local network. The 6LoWPAN mesh is used to connect low power sensing and actuation devices (most likely battery-operated devices) to the LAN infrastructure using IPv6. The gateway is thus responsible for compression and decompression of full length IPv6 packets to their shorter 6LoWPAN versions and vice versa. One of the main design objectives was to keep the device as small and integrated as possible while still maintaining the functionality of a desktop computer based gateway device [3]. The use of 6LoWPAN enables battery-operated sensing and actuation devices to communicate directly to other full function devices using IPv6 without the need for glue logic software on the gateway to translate between protocols and commands used in traditional WSN deployments. ContikiOS was chosen to implement the 6LoWPAN mesh instead of TinyOS due as there is a larger pool of programmers globally for C code as opposed to NesC used in TinyOS [13].

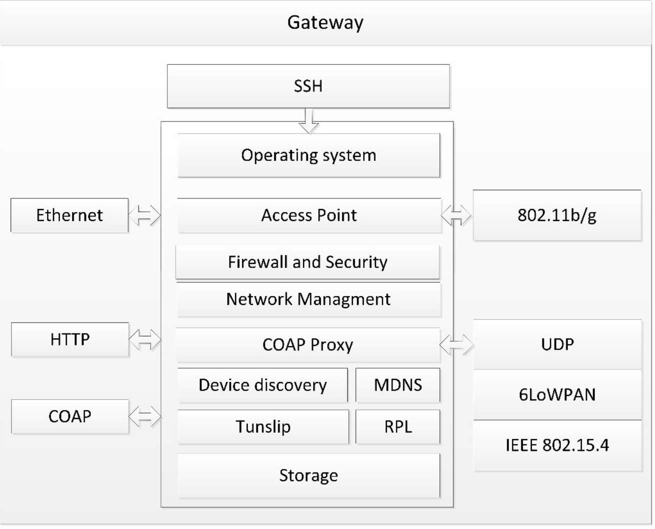

Fig. 1: Functional system diagram of the gateway device.

The integration of the Wi-Fi and 6LoWPAN connectivity into a single device can possibly solve the co-existence problems experienced in [14] where separate physical devices were used for the 6LoWPAN and Wi-Fi interfaces by actively synchronising the Wi-Fi and 6LoWPAN connectivity to use different spectrum at alternating time periods. An important component of the gateway design is the ability to host a Java-based CoAP proxy [15]. The proxy translates RESTful commands from HTTP to CoAP as well as CoAP from an IPv4 network to CoAP for an IPv6 network eliminating the need for a 6to4 tunnel as used in [16] to connect device to the internet. The gateway design is thus similar to [3] in design, but has the added benefit of using CoAP to handle fragmentation and re-assembly of data packets between the motes and application at the application layer. The application layer itself can thus vary the amount of data being sent to optimise the network throughput without regard for physical layer constraints. Secure Shell (SSH) access to a gateway

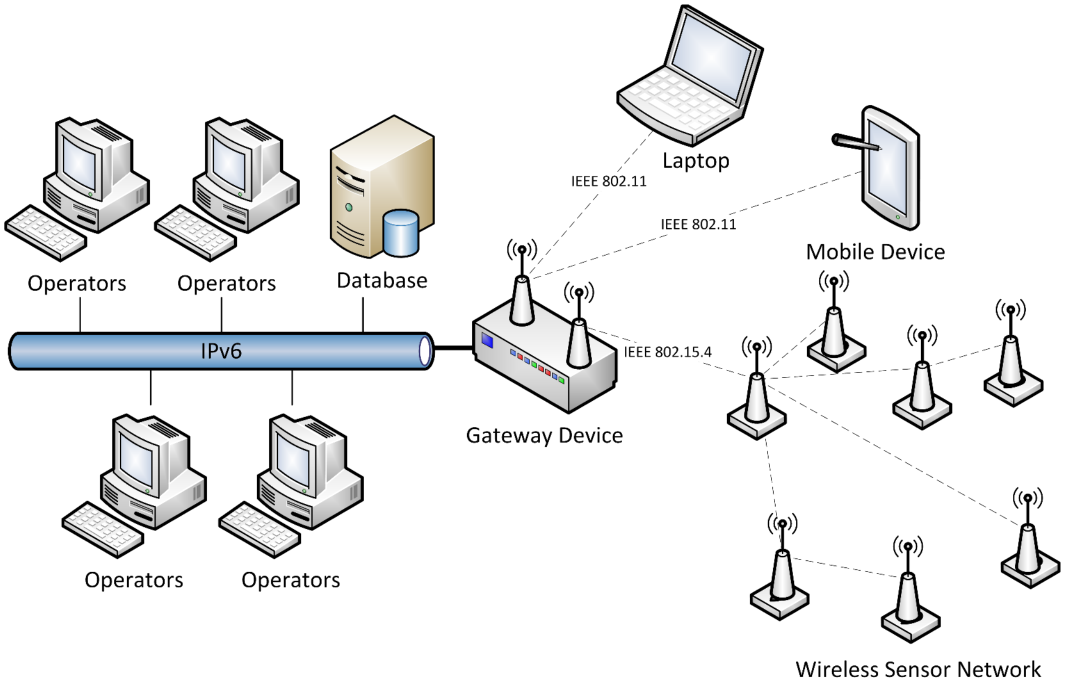

Fig. 2: Network layout of the implemented system.

device is one of several advantages obtained when using a full Linux Kernel based operating system (OS) and allows remote access to the device even over delay intolerant networks. SSH can be used in conjunction with infrastructure management software listed below to achieve highly scalable and manageable deployments that can be maintained by IT professionals as part of the existing network:

- Puppet - A configuration management platform that allows configuration files to be pushed to multiple devices using a template [17].
- Pandora FMS (Flexible Monitoring Software) - A network monitoring tool that can graphically depict the connectivity status of a network and the various devices and services running on the network [18].
- Clonezilla - A disk imaging tool that can be used to clone flash drives for rapid deployment and backup of gateways in case of failure [19].

The use of a Linux based operating system also allows for various software packages to be installed and removed on the fly as the role and requirement of the gateway device changes. Firewall and security measures are implemented by installing an iptables firewall for both IPv4 and IPv6 traffic using well known and proven configuration syntax which greatly improves the security and flexibility of the device from a traffic routing and blocking perspective. Routing and automatic IPv6 configuration can be achieved using various software utilities available from the built in repositories listed below:

- Quagga Routing Software Suite - A routing daemon for Linux with support for IPv6 [20].
- Linux IPv6 Router Advertisement Daemon (radvd) - The linux Router Advertisement Daemon for advertising IPv6 routers to host devices [21].
- 6to4 tunnels - A tunneling protocol for transporting IPv6 packet over IPv4 networks if end-to-end IPv6 connectivity is not available.

IV. Real-world deployment

The real world use-case for the gateway was constructed around the idea of providing connectivity for a solar powered water meter [22] of which the first prototype is shown in Fig. 3. The gateway was required to interface the 802.15.4 low power mesh interface of the water meter to the building local area network (LAN). The networked water meter allowed a JBoss based visualisation platform to collect and graph the water meter readings. The gateway device is the by-product of the testing done in [23] and combines the root node and personal computer into a single device capable of being pole or wall mounted in outdoor applications. The gateway device was realised using a Raspberry PI connected to a STM32W108 based wireless 802.15.4 module. The module uses ContikiOS for the operating system and interacted with the Linux Kernel on the Raspberry PI using a SLIP tunnel (tunslip6). Wi-Fi access point functionality is implemented using a TP-Link Wireless dongle of which the enclosure was removed. The COTS components were re-packaged in a weatherproof IP-66 rated enclosure with a wall mount bracket pre-fitted. The initial prototype of the gateway is shown in Fig. 4 with a custom built adapter board for the RF module (Fig. 5) that was used to connect the Raspberry PI to the RF module in a secure manner. The initial prototype was used to better understand the practical and environmental factors

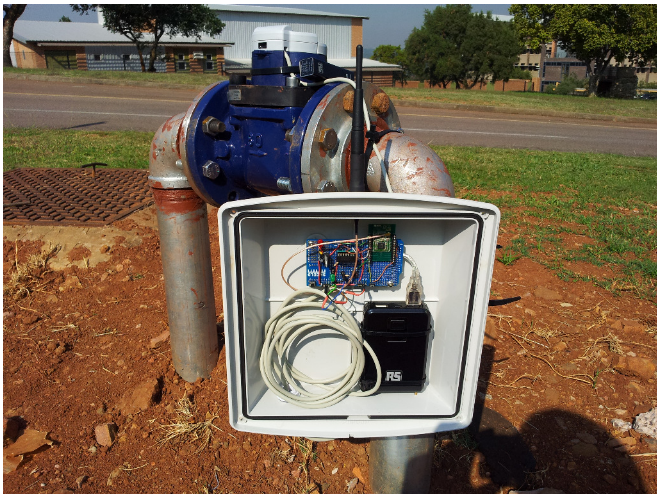

Fig. 3: The prototype water meter connected to the gateway device.

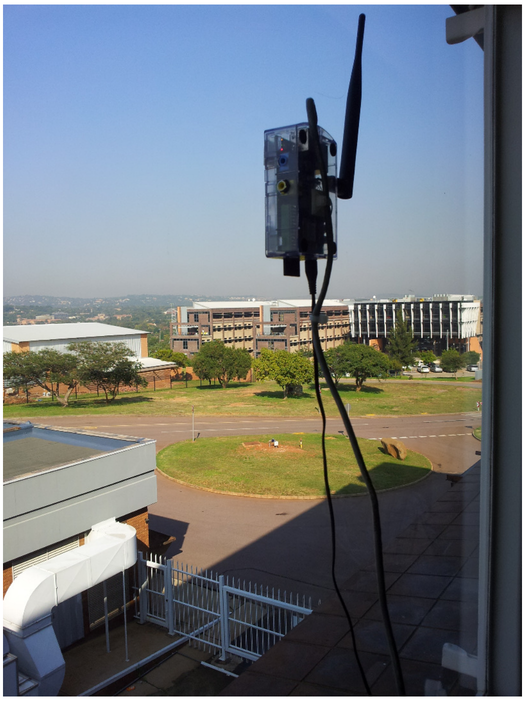

Fig. 4: The first prototype of the gateway device.

that influence the reliability of the communication before the final unit was to be built. Testing the gateway device in a real world deployment scenario revealed several key practical considerations regarding the deployment and installation of gateway devices summarised in bullet form below:

- Wi-Fi interference in a fairly congested office building can have a considerably negative effect on the performance of the 802.15.4 devices. The use of a directional antenna is highly recommended to increase sensitivity and reduce noise, making sure that the line-of-sight (LOS) of the two devices are aligned.
- The ability to remotely access and initiate remote firmware upgrades is a key selection criteria when selecting commercial off-the-shelf components to implement a gateway device. Extensive device management software considerably reduces the complexity and logistics of maintaining a gateway device.
- Installing new devices in office buildings will require building management to install power and data connections that can be quite costly and time consuming if third party contractors are involved. It is thus important that the gateway device has the capability to be powered from a power over Ethernet (POE) power source that can utilise the network data cable to provide power to the device even if this increases the initial device cost. Passive POE works well and is relatively easy to implement.

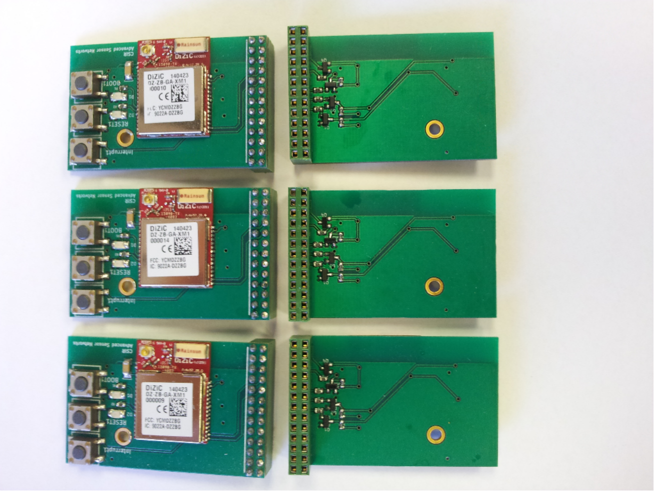

Fig. 5: The adapter board that was specifically designed to interface the Raspberry PI with the STM32W108CC radio.

The initial prototype developed into a second design (Fig. 6) and a final design (Fig. 7) that was installed on the outside of a building at the Council for Scientific and Industrial Research (CSIR). The final version of the device used a weatherproof IP-66 rated enclosure with a built-in 2.4 GHz flat panel antenna to reduce Wi-Fi interference. The device connected to the building network using a shielded Ethernet cable with sunlight and corrosion resistant properties. The quality of the link between the WSN node and gateway device was assessed using an automated script in conjunction with the ping6 command. Round-trip times were measured between the gateway and WSN node using varying ICMP packet sizes. The round-trip time (RTT) for one hundred 84 byte ICMP packets is shown in Fig. 8. The measurements show that the majority of packets are received within 43 ms and that the maximum jitter on the link was 55 ms. These measurements compare well with the 30 ms obtained in [24] where RTT‘s were extensively tested for various topologies. The 84 byte packet was the upper limit of the allowable ICMP packet size if fragmentation is not allowed. The experiment was repeated by varying the packet size from 24 to 84 bytes and graphing the minimum, average and maximum RTT‘s as

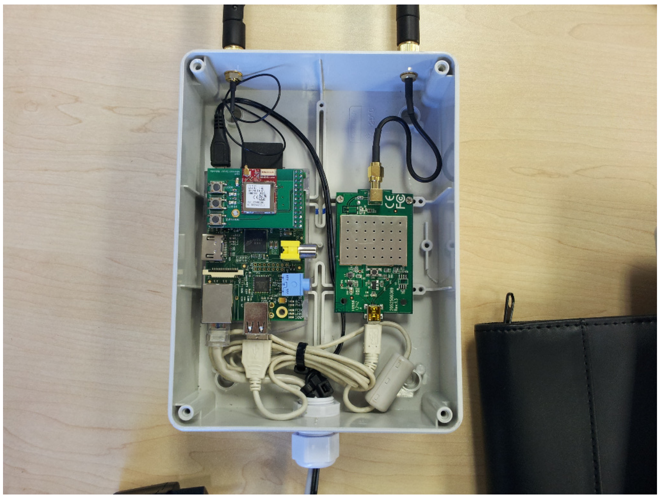

Fig. 6: The second prototype of the gateway device.

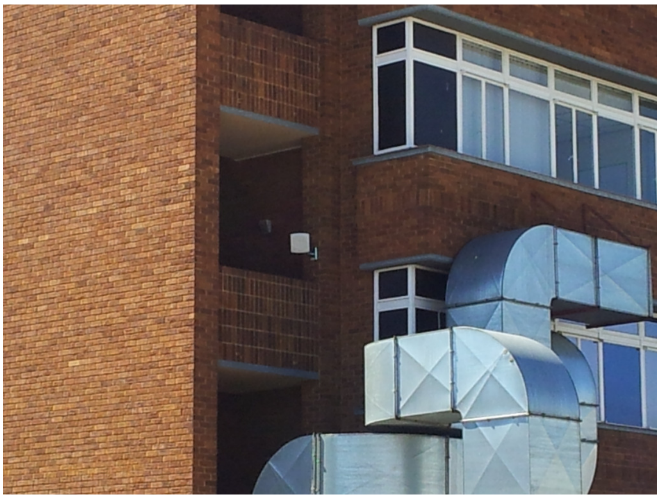

Fig. 7: The gateway installed on the side of a building with the panel antenna facing toward the water meter.

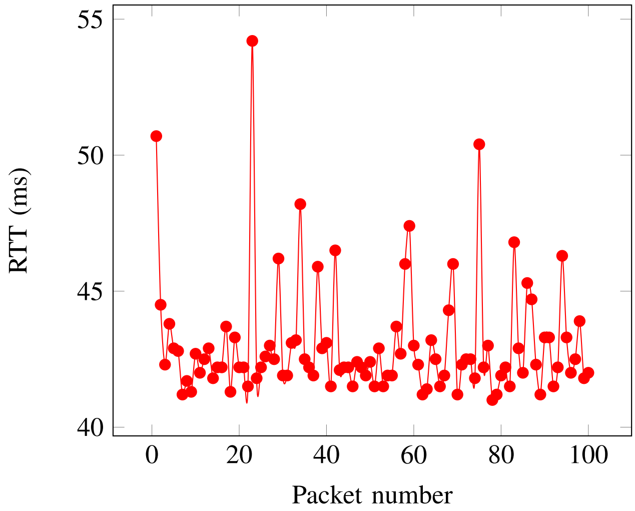

Fig. 8: Round-trip time to the water meter for an 84 byte packet.

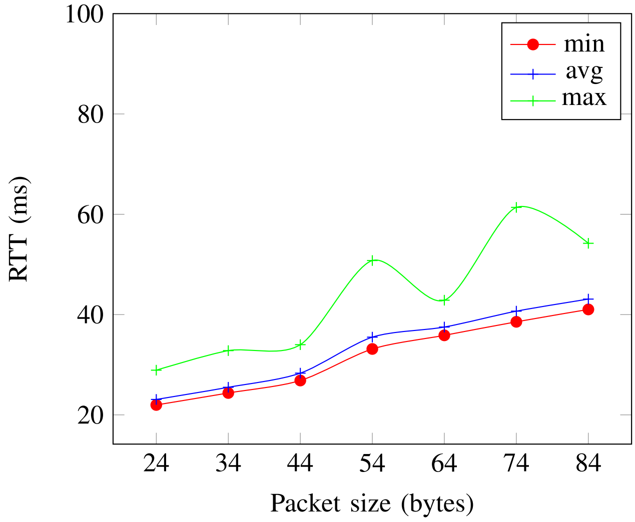

Fig. 9: Latency deviations to the water meter node.

shown in Fig. 9. The graph shows that larger data packets take longer to transmit as expected and that the minimum RTT remains fairly constant in contrast to the maximum RTT that tends to fluctuate more when packets get larger. The results show that large packets should be avoided when optimising for latency. The measurements were stopped at 84 bytes because of the upper limit of 127 byte packets imposed by the 802.15.4 standard without protocol overhead. CoAP automatically implements the fragmentation and reassembly of data at the application layer using block wise transfers and it is thus not necessary to test for larger packet sizes.

V. Conclusions and Future Work

The gateway device was built and prototyped in a three-month period using off-the-shelf components showing that a lengthy product development life cycle is not required to implement gateway devices. The device successfully connected a WSN node using CoAP and 6LoWPAN to the internal IPv6 network of the building where it was monitoring the water consumption of the building. In addition, the device seamlessly integrated into the existing networking infrastructure and did not require any additional power or networking cabling that was not already present. The ability to retrofit existing building infrastructure is a key requirement if Wireless Sensor Networks are to play a significant role in future IoT developments. Future work will include reducing the size of the physical device and to reduce the cost of the device by integrating the off-the-shelf components onto a single PCB design. The gateway device will be made compatible with an open source network monitoring tool to allow network administrators to easily diagnose network related issues that may occur on the gateway device.

VI. Acknowledgements

This work was partially funded by the Department of Science and Technology (DST), South Africa.

References

[1] L. Trappeniers, M. A. Feki, F. Kawsar, and M. Boussard, "The internet of things: the next technological revolution," *Computer*, vol. 46, no. 2, pp. 24–25, 2013.

[2] S. Hodges, S. Taylor, N. Villar, J. Scott, D. Bial, and P. T. Fischer, "Prototyping connected devices for the internet of things," *Computer*, vol. 46, no. 2, pp. 26–34, 2013.

[3] B. da Silva Campos, J. J. Rodrigues, L. D. Mendes, E. F. Nakamura, and C. M. S. Figueiredo, "Design and construction of wireless sensor network gateway with ipv4/ipv6 support," in *International Conference on Communications (ICC)*, 2011, pp. 1–5.

[4] G. P. Hancke, B. de Carvalho e Silva, G. P. Hancke Jr *et al.*, "The role of advanced sensing in smart cities," *Sensors*, vol. 13, no. 1, pp. 393–425, 2012.

[5] S. J. Isaac, G. P. Hancke, H. Madhoo, and A. Khatri, "A survey of wireless sensor network applications from a power utility's distribution perspective," in *AFRICON*, 2011, pp. 1–5.

[6] J. J. Rodrigues and P. A. Neves, "A survey on ip-based wireless sensor network solutions," *International Journal of Communication Systems*, vol. 23, no. 8, pp. 963–981, 2010.

[7] K. Becher, C. Figueiredo, C. Muhle, R. Ruff, P. Mendes, and K. Hoffmann, "Design and realization of a wireless sensor gateway for health monitoring," in *Engineering in Medicine and Biology Society (EMBC)*, 2010, pp. 374–377.

[8] Q. Zhu, R. Wang, Q. Chen, Y. Liu, and W. Qin, "Iot gateway: Bridgingwireless sensor networks into internet of things," in *Embedded and Ubiquitous Computing (EUC)*, 2010, pp. 347–352.

[9] Z. X. Yuan and J. X. Cheng, "The design and realization of wireless sensor network gateway node," *Advanced Materials Research*, vol. 760, pp. 462–466, 2013.

[10] A. Kumar, H. Kim, and G. P. Hancke, "Environmental monitoring systems: A review," *IEEE Sensors Journal*, vol. 13, no. 4, p. 1329, 2013.

[11] A. Kumar and G. Hancke, "Energy efficient environment monitoring system based on the ieee 802.15.4 standard for low cost requirements," *IEEE Sensors Journal*, vol. 14, no. 8, pp. 2557–2566, Aug 2014.

[12] W. Yue, Y. Zhang, Z. Qin, M. Zhu, C. Jin, L. Wang, L. Shu, and C. Chen, "Gatewaying the wireless sensor networks," in *Mobile Ad-hoc and Sensor Networks (MSN)*, 2013, pp. 61–66.

[13] C. Kruger, A. Abu-Mahfouz, and S. Isaac, "Modulo: A modular sensor network node optimised for research and product development," in *IST-Africa Conference and Exhibition*, 2013, pp. 1–9.

[14] M. Najmuddin, M. Hassan, L. Munirah, Z. Ammar, and R. Badlishah, "Integration and deployment of ieee 802.15. 4 wireless sensor networks with a wireless mesh backhaul network," *Advances in Environmental Biology*, vol. 7, no. 12, pp. 3775–3782, 2013.

[15] C. Kruger and G. Hancke, "Benchmarking internet of things devices," in *Industrial Informatics (INDIN) - Accepted for publication*, 2014.

[16] W. Shen, Y. Xu, D. Xie, T. Zhang, and A. Johansson, "Smart border routers for ehealthcare wireless sensor networks," in *Wireless Communications, Networking and Mobile Computing (WiCOM)*, 2011, pp. 1–4.

[17] Puppet open source: It automation software for system administrators. Accessed : 2014-11-03. [Online]. Available: http://puppetlabs.com/puppet/puppet-open-source

[18] Pandora fms - the flexible monitoring software. Accessed : 2014-11-03. [Online]. Available: http://pandorafms.com/Community/Community/en

[19] Clonezilla - the free and open source software for disk imaging and cloning. Accessed : 2014-11-03. [Online]. Available: http://clonezilla.org/

[20] Quagga routing software suite. Accessed : 2014-11-03. [Online]. Available: http://www.nongnu.org/quagga/

[21] Linux ipv6 router advertisement daemon (radvd). Accessed : 2014-11-03. [Online]. Available: http://www.litech.org/radvd/

[22] M. Mudumbe and A. Abu-Mahfouz, "Smart automatic water meter reading based on wireless sensor networks," submitted for publication.

[23] A. G. Dludla, A. M. Abu-Mahfouz, C. P. Kruger, and J. S. Isaac, "Wireless sensor networks testbed: Asntbed," in *IST-Africa Conference and Exhibition (IST-Africa)*, 2013, pp. 1–10.

[24] C. Kruger and G. Hancke, "Implementing the internet of things vision in industrial wireless sensor networks," in *Industrial Informatics (INDIN) - Accepted for publication*, 2014.