Work-In-Progress: Industry-Friendly and Native-IP Wireless Communications for Building Automation

Zhibo Pang[1], Yuxin Cheng[2], Morgan E. Johansson[1], Gargi Bag[1]
1, Corporate Research, ABB AB, Västerås, Sweden
2, ICT School, Royal Institute of Technology (KTH), Stockholm, Sweden
{pang.zhibo|morgan.e.johansson|gargi.bag}@se.abb.com, yuxinc@kth.se

Abstract—**Wireless communication technologies for building automation (BA) systems are evolving towards native IP connectivity. More Industry Friendly and Native-IP Wireless Building Automation (IF-NIP WiBA) is needed to address the concerns of the entire value chain of the BA industry including the security, reliability, latency, power consumption, engineering process, and independency. In this paper, a hybrid architecture which can seamless support both Cloud-Based Mode and Stand-Alone Mode is introduced based on the 6LoWPAN WSAN (wireless sensor and actuator networks) technology and verified by a prototyping minimal system. The preliminary experimental results suggest that, 1) both the WSAN and Cloud communications can meet the requirements of non-real-time application of BA systems, 2) the reliability and latency of the WSAN communications is not sufficient for soft real-time applications but it is not far away to meet such requirements by sufficient optimization in the near future, 3) the reliability of Cloud is pretty sufficient but the latency is quite far from the requirement of soft real-time applications. To optimize the latency and power consumption in WSAN, design industrial friendly engineering process, and investigate security mechanisms should be the main focus in the future.**

Keywords—Wireless Sensor and Actuator Networks (WSAN); 6LoWPAN; Native IP Connectivity (NIP); Wireless Building Automation (WiBA)

I. INTRODUCTION

Building Automation (BA) for residential buildings or homes, commercial buildings, and industrial buildings is one of the most promising application area of the Internet-of-Things (IoT) [1]. Thanks to the reduced cost of installation and maintenance and improved user experiences, Wireless Sensor and Actuator Network (WSAN) technologies are being actively applied or developed and therefore the Wireless Building Automation (WiBA) has become the new design paradigm of future BA systems [2]. To bring native support of Internet Protocol (IP), the so-called Native IP (NIP), to lightweight WSAN devices is a promising direction of the evolution of communication technologies for BA systems. It can not only ease the interoperability challenges during the system integration of various devices, sub-systems, and value-added services from different suppliers, but also tear down the walls that are hindering the BA industry to benefit from the vast amount of innovations in internet domain which evolves much faster. In practical standardization efforts, the evolution towards NIP connectivity has been clearly observed in most of the established or emerging BA communication standards, both wireless and wired, e.g. BACnet has released the BACnet/IP [3], KNX has released the KNXnet/IP [4], ZigBee has released the ZigBee IP [5], Bluetooth is developing the 6LoWPAN-over-BLE [6], and DECT ULE is developing the 6LoWPAN-over-ULE [7]. Given the standards which already have NIP connectivity such as the IEEE802.11ah Low Power WiFi [8], Thread Group [9], and the IETF IoT Suite (6LoWPAN, RPL, and CoAP) [10], the BA industry has reached a common consensus to enter the NIP era in near future even though the landscape of standardization is till fragmented.

To realize this vision of NIP-based WiBA, in addition to the technical challenges about reliability, latency, power consumption, and complexity, some important concerns from the standpoint of value chain should be addressed, e.g. "*Is that a good idea to connect everything in buildings to internet or cloud? How to reduce the security and privacy risks when enjoy the benefits from IP connectivity? How to inherit the experiences, best practices and tools for engineering and commissioning? How to strengthen their roles in the existing value chain which seems potentially to be disrupted by new entrants?*" In other words, the BA industry is demanding an Industry-Friendly and Native IP (IF-NIP) communication architecture which will not only meet the critical technical performances but also take care the business benefits of all the stakeholders in the value chain. However the existing efforts in this direction are insufficient due to the misinterpretation of "*NIP connectivity*", or lack of friendliness to system integrators and installers, or lack of support to engineering workflow (see section II for more details).

As a work in progress, this paper intends to present some preliminary thoughts and findings towards the vision of IF-NIP WiBA systems. In particular, a hybrid communication architecture is proposed to support flexible combination of Cloud-Based Mode and Stand-Alone Mode. Preliminary experimental results of a prototyping minimal system based on 6LoWPAN are presented with special respects with latency and reliability. The technical feasibility of the proposed architecture for non-real-time applications is confirmed and the needs of improvement for soft real time applications are suggested as well. Ongoing work about security, power consumption, and engineering process are discussed but without experimental results since they are out of the scope of this paper. The industrial friendly considerations of the proposed vision and architecture can accelerate the market penetration since the major concerns of the entire industrial value chain are addressed better.

INISCom 2015, March 02-04, Tokyo, Japan

DOI 10.4108/icst.iniscom.2015.258563

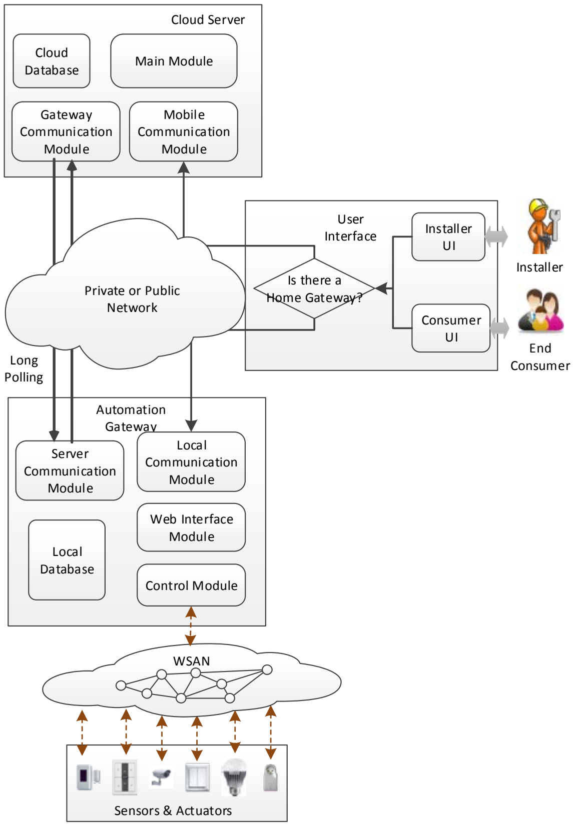
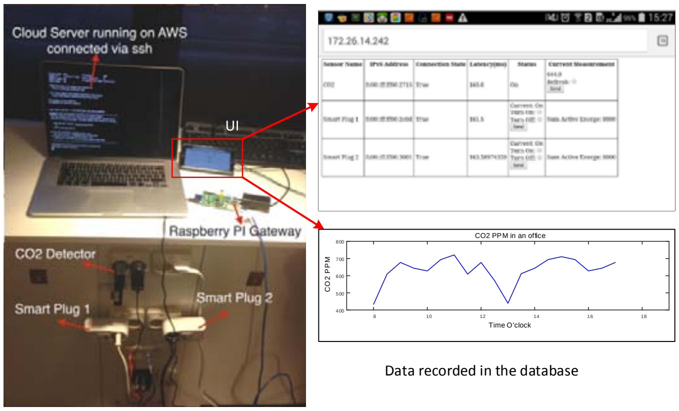


Fig. 1. A minimal system of the proposed hybrid IF-NIP WiBA architecture

II. System Architecture

Given the pros and cons of the Cloud-Based Mode and Stand-Alone Mode, the BA industry is demanding a hybrid WiBA architecture which can take the advantages of the both modes i.e. to simplify the system configuration and maintenance by the Cloud-Based Mode, to reduce the security and privacy issues by the Stand-Alone Mode, and to make the solution more friendly to both system integrators, installers, and dedicated service providers by offering flexible combination of the two modes.

The hybrid architecture for the IF-NIP-WIBA has been proposed in our previous work [11]. It can seamlessly support both Cloud-Based Mode operation, Stand-Alone Mode operation, and flexible combination of the two basic modes. Benefited from the Cloud-Based Mode, the system integrators can choose to provide remote services for engineering, commissioning, command validation, and maintenance by themselves through a private cloud, and they can choose to partner with 3rd parties for more value-added services like energy management and optimization, in home healthcare, and many more. Benefited from the Stand-Alone Mode, they can also choose to perform the engineering, commissioning, and maintenance in the field by professional installers utilizing exactly the same workflow as the wired BA systems so that the security issues and dependency can be minimized. Additionally, they can combine the Cloud-Based Mode and Stand-Alone Mode in a more flexible way, e.g. using the Cloud-Based Mode during commissioning to ease the engineering and configuration and using the Stand-Alone Mode after commissioning.



172.26.14.242

Sensor Name	IPv6 Address	Connection State	Latency(ms)	Status	Current Measurement
CO2	[illegible]	True	161.6	On	644.0 Refresh: ○ Send
Smart Plug 1	[illegible]	True	161.5	Current: On Turn On: ○ Turn Off: ○ Send	Sum Active Energy: 0000
Smart Plug 2	[illegible]	True	163.58974359	Current: On Turn On: ○ Turn Off: ○ Send	Sum Active Energy: 0000

Fig. 2 Implemented hardware of the minimal system including the Cloud Server, User Interface, Automation Gateway, and the WSAN devices, and the CO2 measurements collected over a normal working day in the office building

To identify the technical requirements and verify the concepts, a minimal system of the proposed hybrid IF-NIP WiBA system architecture is designed as Fig 1. The minimal system is basically formed of four elements: an Automation Gateway, a Cloud Server, a User Interface (UI), and a series of sensors and actuators connected through WSAN. The users (end consumers or installers) can connect to the Cloud Server or the Automation Gateway directly from their UIs e.g. a smart phone or tablet, depending on user's choice and the network connection. The 6LoWPAN technology is used in the WSAN for native support of IPv6 connectivity over low power wireless communications. More details of the minimal system is given in [11].

III. Prototypes

A prototype of the minimal system is being implemented. Some of the hardware setup is shown in Fig. 2. Details are descripted below.

A. Cloud Server

The Cloud Sever application runs in an Ubuntu 12.04 image on Amazon AWS EC2 Platform. The service type is t2.micro, which contains one virtual CPU, 1 GiB memory along with 8 GB store. The server program is written in Python. It forwards any HTTP request (GET, POST) from the web interface to the Automation Gateway, and then sends the received HTTP response from the Automation Gateway back to users.

B. User Interfaces

A static webpage is implemented as the User Interface of our system. From the table in the webpage, users can identify the sensors' name, as well as the IPv6 address, network connection status, average latency within the local WSAN network. Users can also control the sensors from the webpage. In our case, users can turn on/off a smart plug, get the current measurement from CO2 detector, etc. There is no difference between the user interface on cloud server and the user interface on automation gateway. Users can have the same controlling command on both user interfaces.

C. *Automation* Gateway

A low cost Linus host, Raspberry Pi, is used to implement the Automation Gateway. It has 512 MB of RAM, two USB ports and a 100Mb Ethernet port. The Raspbian Image is used as the gateway operating system. The Server Communication Module, Local Communication Module and Control Module are implemented as Python program. A Python network engine is designed in the program. The Database is based on SQLite database running on the Raspberry Pi. A simple Web Interfaces is also provided. After giving the right user name and password, users can have a direct control to the WSAN devices. The Automation Gateway application integrates IPv4 socket and IPv6 socket harmoniously. The IPv4 socket manages the communication with the Could Server, as well as responds to the HTTP request from the Web Interface. The IPv6 socket manages the connection to the USB 6LoWPAN Border Router not only to control the devices within the WSAN but also to enroll a new devices to the network. In the Automation Gateway program, a daemon thread is set to monitor the current network status of the sensor, measuring the average latency of every sensor within the local sensor network.

D. WSAN *Devices*

The 6LoWPAN devices from Watteco NKE Electronics are selected to implement the WSAN devices. Two smart plugs and one CO2 detector are used in the prototype system. All the devices are operates at 868MHz ISM band. The IPv6 adaptation layer is based on the IETF 6LoWPAN standard. The IETF RPL (routing protocol of the IPv6 packets over low-power and lossy network) protocol is used for mesh networking. In the application layer, the ZigBee Cluster Library (ZCL) format packets are inserted as the payload of UDP packets. Besides the sensor devices, a USB 6LoWPAN Border Router is used to provide the radio connection between the Automation Gateway and the sensor devices. It can be plugged on a Linux host and creates the link between standard IPv6 applications and 6LoWPAN devices. It also takes the role to setup the WSAN and in turn to allow new devices joining in.

IV. Preliminary Results

A. *Experiment Setup*

As mentioned, latency and reliability are of the concerns when IP-based communication is applied due to the lack of real-time mechanisms. As shown in Fig.3, the hardware is setup in a corridor of our office building which is about 100 meters long, and the Automation Gateway is installed in the office aside. The WSAN is configured to be three hops. The Cloud Server which is deployed on the Amazon PaaS located in Ireland.

B. *Definition of Evaluation Criteria*

In this experiment, the latency is measured by Round Trip Time (RTT). As shown in Fig. 4, two types of RTTs are defined. The RTTs are denoted by the number of hops in the WSAN. For example, the RTT between the User Interface and the 3rd hop device is denoted as RTT_{UI-h3}. They a

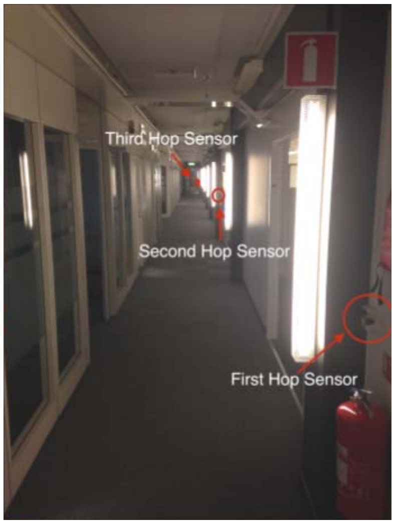


Fig. 3. Test setup in the office building

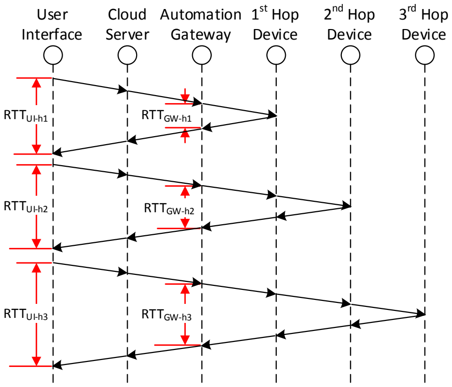


Fig. 4. Definition of Round Trip Time (RTT) between UI and Gateway and between Gateway and Devices with different number of hops

- *RTT_{UI}*: it starts from the moment when the User Interface sends out a command to one of the WSAN devices, and ends at the moment when the User Interface receives a response from the device. During the test, the User Interface software sends out a command to e.g. the 3rd hop Device, then the command is forwarded by the Cloud Server, Automation Gateway, 1st hop Device, and 2nd hop Device sequentially to the 3rd hop Device, then the 3rd hop Device sends out its response, then the response forwarded by the 2nd hop Device, 1st hop Device, Automation Gateway and Cloud Server sequentially back to the User Interface, finally the User Interface receives the response and records the time duration as $RTT_{UI-hop3}$.
- *RTT_{GW}*: it starts from the moment when the Automation Gateway sends out a command such as Read or Write to one of the WSAN device, and ends at the moment when the Automation Gateway receives a response from the device. During the test, the Automation Gateway software sends out a command to e.g. the 3rd hop Device, then the command is forwarded by the 1st hop Device, and 2nd hop Device sequentially to the 3rd hop Device, then the 3rd hop Device sends out its response, then the response forwarded

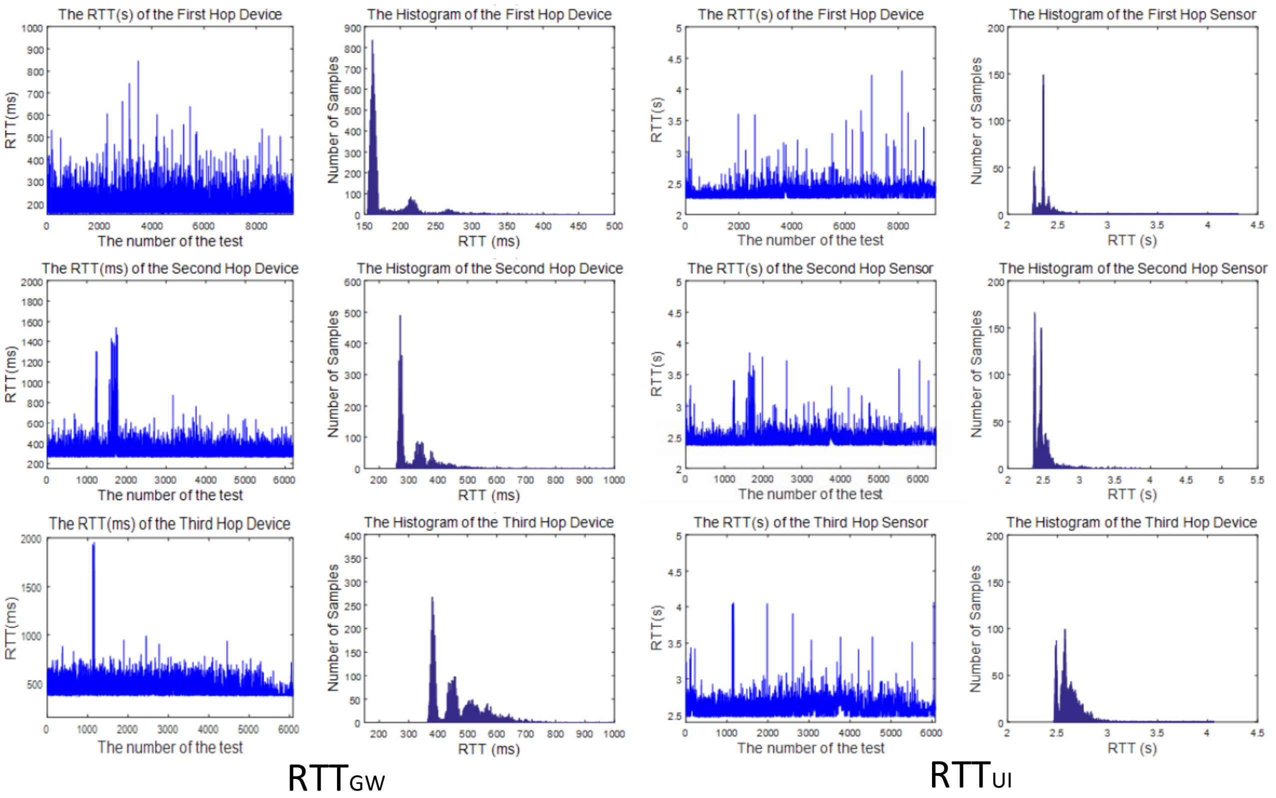


Fig. 5. Round Trip Time between Gateway and Devices (RTT_{GW}) and between User Interface and Devices (RTT_{UI}), and their histograms

by the 2nd hop Device and 1st hop Device sequentially back to the Automation Gateway, finally the Automation Gateway receives the response and records the time duration as $RTT_{GW\text{-}hop3}$.

Reliability is measured by the Round Trip Packet Error Rate (RT-PER) which is the percentage of the commands that are not responded correctly before timeout among the total commands sent during the period of test. The RT-PER are measured at the User Interface and Automation Gateway and denoted as RT-PER_{UI} and RT-PER_{GW} respectively.

C. Data Analysis

The test results from an experiment that lasted for about 15 hours are plotted in Fig. 5 and statistics of the data is collected in Table I. Some observations are described below.

- *Latency of WSAN*. RTT_{GW} represents the latency caused by the communications within the WSAN. 1) The average RTT_{GW} is on the order of hundreds of milliseconds, e.g. 184ms, 324ms and 465ms for 1st, 2nd hop, and 3rd hop respectively in this experiment. 2) The average RTT_{GW} increases proximately linearly when the number of hops increases. In experiment, the average RTT_{GW} increases by 140ms for each hop. 3) The distribution of RTT_{GW} is quite diverse. In experiment, the maximum RTT_{GW} is on the order of seconds and 4 to 5 times larger than the average RTT_{GW}.
- *Latency of Cloud*. The RTT_{UI} represents the total latency caused by the communications in the WSAN and Cloud. In this experiment, it is reasonable to assume that the statistic characteristics of the WSAN and Cloud environment is stable during the period of the two commands for RTT_{GW} and RTT_{UI} because they are sent almost at the same moment (the error is less than sub second). So the average latency cause by the Cloud (RTT_{Cloud}) can be proximately estimated by *average*(RTT_{UI})- *average*(RTT_{GW}). 1) The average RTT_{Cloud} is quite stable and not affected by the number of hops in WSAN. In this experiment, the average RTT_{Cloud} is all around 2000s for 1st hop, 2nd hop, and 3rd hop devices. 2) The distribution of RTT_{UI} is less diverse compared with the RTT_{UI}. In this experiment, the maximum RTT_{UI} is always less than twice of the minimum RTT_{UI}. 3) Occupying the major part of the total latency, the latency of Cloud is 4 to 10 times larger than the WSAN latency.
- *Reliability of WSAN*. The RT-PER_{GW} represents the command failure caused by the communications within the WSAN. The RT-PER_{GW} increases when the number of hop increases. In this experiment, we not have observed any command failure for the 1st hop device. However we are not sure about whether there is no packet loss because we are not clear if there is any re-transmission mechanism in the lower layers of the protocol implemented by the Watteco devices. The RT-PER_{GW} increase up to 3.15% and 4.32% at the 2nd hop device and 3rd hop device respectively.
- *Reliability of Cloud*. The RT-PER_{UI} represents the total command failure caused by the communications within Cloud and WSAN. Because in this experiment, the communications between the Cloud Server and User Interface and the Automation Gateway are based on TCP which has automatic retransmission and guaranteed end-to-end reliability, there is no command failure caused by the

Table I. Statistics of the Round Trip Time (RTT) and Round Trip Packet Error Rate (RT-PER)

		1st Hop Device	2nd Hop Device	3rd Hop Device
RTT_{UI}	Min(ms)	2247.2	2354.1	2463.9
	Average(ms)	2347.1	2473.6	2612.4
	Max(ms)	4300.7	3849.9	4061.6
	σ (ms)	94.3	118.7	123.2
	avg+σ (ms)	2441.4	2592.3	2735.6
	avg+2σ (ms)	2535.7	2711	2858.8
	avg+3σ (ms)	2630	2829.7	2982
	P@avg+σ	91.41%	91.63%	88.83%
	P@avg+2σ	97.43%	96.86%	97.43%
	P@avg+3σ	98.68%	98.17%	99.00%
RTT_{GW}	Min(ms)	154	258	364
	Average(ms)	184	324	465
	Max(ms)	846	1531	1950
	σ (ms)	47	96	101
	avg+σ (ms)	231	420	566
	avg+2σ (ms)	278	516	667
	avg+3σ (ms)	325	612	768
	P@avg+σ	89.76%	91.58%	86.52%
	P@avg+2σ	95.13%	97.35%	97.78%
	P@avg+3σ	97.87%	98.58%	99.59%
RTT_{Cloud}	Average(ms)	2163.1	2149.6	2147.4
$RT\text{-}PER_{UI}$	%	0.00%	3.15%	4.32%
$RT\text{-}PER_{GW}$	%	0.00%	3.15%	4.32%

Cloud in fact. So the $RT\text{-}PER_{UI}$ is equal to the $RT\text{-}PER_{GW}$ in this experiment.

D. *Performance Assessment*

- *Communications of WSAN.* For non-real-time applications through local network, such as monitoring of status of sensors and configuration of operation parameters (e.g. schedule, work mode) of actuators by an in-home display (IHD) or smart phone, the latency and reliability of WSAN communications is acceptable. For example, in this experiment if we set the up bound of Maximum Acceptable RTT as 768ms which is the *Average_RTT*$_{GW}$+ *3*Standard_Deviation* of the furthest away 3rd hop devices, about 95.68% of the commands can receive correct responses, and 99.59% of the responses arrives within. However this cannot meet the requirement of soft real time applications such as remote control of dimmerable lights or curtains which need the latency to be *imperceptible* for human. According to the rule-of-thumb in practice, *imperceptible latency* usually is defined as >95% of the commands are responded correctly within 150ms. In this experiment, even for the nearest 1st hop device, 95.13% of the responses arrive within 278ms. But it is not far from the acceptable level.
- Communications of Cloud. The reliability of Cloud communications is pretty good for most of the applications. But the latency can only meet the requirement for non-real-time applications. It is quite far (10 time larger) from the requirements of soft real-time applications.

V. Conclusion

Wireless communication technologies for building automation systems are evolving towards native IP connectivity. To realize the vision of Industry Friendly and Native-IP Wireless Building Automation (IF-NIP WiBA) systems, more industry friendly wireless communication technology is needed to address the concerns of the entire value chain of the BA industry, including the security, reliability, latency, power consumption, engineering process, and independency. The preliminary experimental results suggest that, 1) both the WSAN and Cloud communications can meet the requirements of non-real-time application of BA systems, 2) the reliability and latency of the WSAN communications is not sufficient for soft real-time applications but it is not far away to meet such requirements by sufficient optimization in the near future, 3) the reliability of Cloud is pretty sufficient but the latency is quite far from the requirement of soft real-time applications.

In the next step of this work, to optimize the latency and power consumption in WSAN, implement industrial friendly engineering process, and investigate security mechanisms should be the main focus.

References

[1] Zhibo Pang, “Technologies and Architectures of the Internet-of-Things (IoT) for Health and Well-being”, PhD Thesis, Royal Institute of Technology (KTH), Stockholm, Sweden, 2013.

[2] Langhammer, N. ; Kays, R. "Performance Evaluation of Wireless Home Automation Networks in Indoor Scenarios", Smart Grid, IEEE Transactions on, Vol.3, Iss4DOI: 10.1109/TSG.2012.2208770, Page(s): 2252- 2261, 2012

[3] BACnet Offical Website, www.bacnet.org, Online accessed on Oct 21, 2014

[4] Michele Ruta, Floriano Scioscia, Giuseppe Loseto, Eugenio Di Sciascio, "KNX, a worldwide standard protocol for home and building automation: state of the art and perspectives", Industrial Communication Technology Handbook, CRC Press/Taylor & Francis, page 1463--1481 - aug 2014

[5] ZigBee Alliance, "ZigBee IP Specification (ZigBee Public Document 13-002r00)", February 2013

[6] IETF 6Lo Working Group, "Transmission of IPv6 Packets over BLUETOOTH(R) Low Energy (draft-ietf-6lo-btle-03)", September 1, 2014

[7] IETF 6Lo Working Group, " Transmission of IPv6 Packets over DECT Ultra Low Energy (draft-mariager-6lowpan-v6over-dect-ule-03)", July 15, 2013

[8] Low Power WiFi, www.wifi.org, Online accessed on Oct 21, 2014

[9] The Thread Group, http://threadgroup.org Online accessed on Oct 21, 2014

[10] Palattella, M.R. ; Accettura, N. ; Vilajosana, X. ; Watteyne, T. ; Grieco, L.A. ; Boggia, G. ; Dohler, M. "Standardized Protocol Stack for the Internet of (Important) Things", Communications Surveys & Tutorials, IEEE, Vol 15, Issue: 3, DOI: 10.1109/SURV.2012.111412.00158, 2013

[11] Zhibo Pang, Yuxin Cheng, Morgan E. Johansson, Gargi Bag, "Preliminary Study on Wireless Home Automation Systems with Both Cloud-Based Mode and Stand-Alone Mode", to appear in the 13th IEEE Int. Conf. on Ubiquitous Computing and Communications, Dec 2014.