Evaluating Design Approaches For Smart Building Systems

Evangelatos, Napoleon-Orestis; Samarasinghe, Kasun; Rolim, Jose

Evaluating Design Approaches For Smart Building Systems

Orestis Evangelatos, Kasun Samarasinghe, Jose Rolim
Computer Science Department
University of Geneva
Geneva, Switzerland
{Orestis.Evangelatos, Kasun.Wijesiriwardana, Jose.Rolim}@unige.ch

***Abstract*—As we are moving towards to the Internet of Things (IoT) era, Wireless Sensor Networks (WSN) in smart buildings delineate the heart of such systems' architecture. WSN systems are mature enough to support the IoT vision and different architectural designs and communication protocols are developed to realize this vision. In this paper, two different WSN architectural approaches for smart building systems are presented. In the first one, IPv6 over Low power Wireless Personal Area Networks (6LoWPAN) deployment is used, which is designed specifically for constrained embedded devices. In the second one, the system is developed without the usage of IP. To evaluate these two approaches we implemented a scenario of a smart building environment on top of them. We analyze and compare them, both from theoretical and practical point of view. Finally, as a proof of concept we evaluated them experimentaly in our testbed and we reported our conclusions.**

Keywords – Internet of Things, Wireless Sensor Networks, Smart Buildings, 6LoWPAN, CoAP.

I. INTRODUCTION

With the rapid advancements in processor technologies and hardware platforms, embedded network systems have drawn a lot of attention in the IT research community. Wireless Sensor Networks (WSN), are one of the realizations of networked embedded systems. Subsequently, with the significant research effort both from academia and industry, the WSN combined with IP technology are becoming the future of embedded internet. Millions of tiny devices connected to the internet are taking the pervasive computing to the next level. This line of research envisions a seamless integration of day to day commodities with the internet, namely the Internet of Things (IoT).

IoT technologies provide an infrastructure for wide range of applications such as industry automation, vehicular ad-hoc sensor networks and smart building systems. Among these, smart building systems are becoming more and more vital due to the improvement they provide to the quality of life. One of the key components of a smart building system is a WSN, which provides the necessary information to the smart building system, allowing it to control and monitor the physical environment.

Frequently WSN operate in isolation but towards their collaboration in the IoT technology, interconnectivity between two or more networks is a challenging task. This is mainly due to the fact that different protocols, systems and implementations do not always operate in harmony. An obvious move towards the amalgamation of the isolated WSN is to adapt multiple and diverse sensor networks to the existing protocols and make them work seamlessly together. Recent research and development have incorporated IP technology with WSN, allowing to bridge the gap between heterogeneous networks. Without the use of IP protocols, "smart" gateways which are capable of interconnecting different protocols could be used to overcome the problem of isolation.

In this paper we present an analysis and a comparison of two different approaches in the context of WSN in smart buildings; an IP-enabled approach and a non IP-enabled. Specifically we present these two approaches both from a design and an implementation perspective. Moreover, we present the communication schemes of each approach and how the several components of the network communicate with each other. In the first approach of the IP-based architecture, a key factor is the IPv6 over Low power Wireless Personal Area Networks (6LoWPAN) [12] protocol. On top of that the Constrained Application Protocol (CoAP) [6] allows direct and simple interactions. In the second approach without the use of IP, the heart of the system is the mesh routing protocol RIME [14].To the best of our knowledge this is the first time such an analysis is being done.

The outline of the paper is as follows; in the section II we briefly describe the background material. In section III we present a system's architecture for smart building applications, followed by the description of the two approaches we analyze in section IV. An evaluation of the schemes is presented in section V based on a scenario in the WSN. In section VI we present the evaluation of the approaches after a testbed implementation. Finally in section VII we present our conclusions.

II. BACKGROUND

As we are moving towards the IoT era we can clearly see the impact of WSN in the future. One of the main aspects of the IoT is the improvement of the Information Technologies and their applications which surrounds us and our environment. Important domain of the scientific efforts and research of the IT challenges is the smart building systems. As shown in [15], the challenges of the next-generation wireless sensor networks in the intelligent buildings could be overcame using state-of-the-art technologies. By exploiting them, we could reach a credible future in the development of smart buildings.

To be able to realize such systems that are auto organizing, easily accessible, efficient and energy aware, new protocols and standards have to be deployed. The state-of-the-art protocol suit 6LoWPAN [2,11,12] deployed by the 6LoWPAN working group of Internet Engineering Task Force (IETF) has defined the frame format for transmission of IPv6 packets to be sent and received over IEEE 802.15.4 networks. They have also designed the formation of IPv6 link-local addresses and statelessly autoconfigured addresses on top of IEEE 802.15.4 networks. The 6LoWPAN stack enables each device to be directly connected to the Web. Based on these IP packets, a RESTful API for sensor nodes has been developed. REST (*REpresentational State Transfer)* is a style of software architecture for distributed systems such as the World Wide Web [4]. REST-style architectures consist of clients and servers. Clients initiate requests to the servers, they process these requests and return appropriate responses. In REST, every resource has its own URI and by using these URIs it is possible to access these resources. The resources themselves are conceptually separated from the representations that are returned to the client.

An implementation of CoAP for the Contiki [3] operating system leverages the ContikiMAC low-power duty cycling mechanism to provide power efficiency [5]. The CoAP enables interoperability at the application layer through RESTful Web services [6]. The experimental evaluation in [5] of their low-power CoAP, demonstrates that an existing application layer protocol can be made power-efficient through a generic radio duty cycling mechanism. Furthermore it is shown that the use of ContikiMAC substantially reduces the motes energy consumption while keeping a reasonable end-to-end latency. The ContikiMAC is the MAC protocol that we are using in our implementation which is described in section IV.

Inside a smart building many sensors and actuators are interconnected to form a control system. Nowadays the deployment of a building's control system is complicated due to different communication standards. In reference [7] authors implemented an API to access services on sensor nodes following the architectural style of REST. An approach towards an integration of tiny wireless sensors or actuator nodes into an IPv6 6LoWPAN based network is presented. They propose the use of lightweight web services based on REST and the representation of data in the JSON format together with the stateless address auto-configuration mechanisms provided by the IPv6 protocol.

In the home automation design field a wireless sensor networks system using 6LoWPAN is proposed [8]. Besides, the JavaScript Object Notation (JSON) format is used to encode the data from the sensors which are deployed in the building. An IPv6 address is given to them allowing flexibility to the system. Data is sent over the network in a simple text format and none of the components of the system needs to know which are the capabilities of each individual node since they can be discerned easily from the data that is sent.

A vital part of an architecture design of WSN is a gateway. The gateway provides all the necessary interconnection schemes that makes WSN feasible to connect to other WSN and to the Web. The design and the construction of a wireless sensor and actuator network gateway based on 6LoWPAN are shown in [9]. A new gateway device which enables end-to-end connectivity between 6LoWPAN-based sensors and Ip enabled devices is presented. The 6LoWPAN adaptation layer is the part of the gateway, which is responsible for the compression of packets addressed to the WSN and the decompression of packets targeted to the IPv6 network.

III. WSN SYSTEM ARCHITECTURE FOR SMART BUILDINGS APPLICATIONS

Wireless sensor networks are being used widely in smart building applications. Multiple sensors deployed around an area can transfer diverse information of their resources to the system while other sensors can receive data to drive appliances connected to them. The key requirement for a smart building is that all sensors and actuators are accessible over the network from humans or other devices in an efficient and reliable way. With the current development efforts, IPv6 has become feasible in sensor networks and nodes are connected to the network using the IEEE 802.15.4 standard which is specifying the physical layer and the media access control for energy efficient communication with low data rates. On top of the IEEE standard, 6LoWPAN is used which allows IPv6 packets to be sent and received throughout the network. IPv4 and IPv6 are the work horses for data propagation. In our work, between the wireless sensor network and the broader network, a gateway is used to forward the packets from one

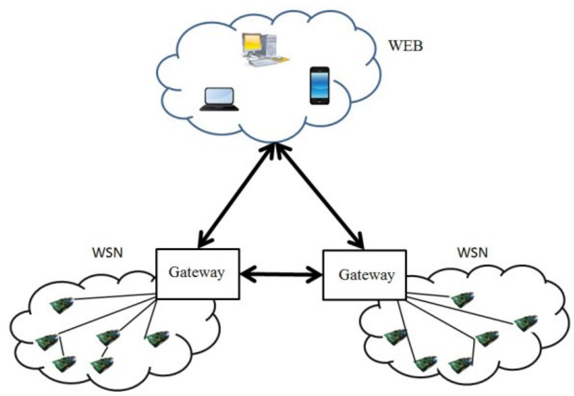

Figure1. Interconnection of WSN using gateways.

subnetwork to the otherand to the Web. The gateway is represented and implemented by a node connected via a serial port to a computer, which is connected to the Web or to other networks either wired or wireless. The main architecture of our system is presented in Fig. 1.

IV. WSN DESIGN APPROACHES; IPV6 VS NON-IP

In this section we present two different approaches for the system architecture of smart building applications. The choice of protocol to be chosen for developing a wireless sensor network and the general system itself is critical as the devices are tightly constrained in terms of energy, payload, communication bandwidth and memory. Questions such as; which communication protocol is more energy efficient, needs less overhead and is more feasible, need to be answered.

We present two different approaches using different communication schemes. The main aim of both approaches remains the same; to deploy a wireless sensor network in a smart building and to develop an infrastructure for accessing the sensor network from the web or another network in a feasible way. The two systems that we present differ in the communication patterns and the protocols that they use, but also in the format of the packets that are being transmitted over the network. On the other hand they use the same 802.15.4 protocols for the Physical and Link layer. Table I shows the layers of the communication protocols of these two approaches.

A. WSN System designed with IPv6 over 6LoWPAN.

In the first approach the system is being deployed with IPv6 over 6LoWPAN protocol for communication of constrained embedded devices together with CoAP. In the link layer the 802.15.4 standard is used, while in the Internet layer the IPv6 – 6LoWPAN protocol is used. In the transport layer the UDP protocol is used while in the application layer we use the CoAP protocol. Short description of the layers follows:

1) 802.15.4

The main characteristics of the 802.15.4 protocol are its low power consumption, support for low latency devices, dynamic device addressing and very low complexity. Data rates are available at 20 kb/s, 40kb/s and 250 kb/s. [10]

2) IPv6 over 6LoWPAN

Following the revolution of Ubiquitous Computing which started in 1990s, and the Internet of Things subsequently, the IETF 6LowPAN group developed a standard and defined mechanisms that allow IPv6 packets to be send to and received over Low-Power and Lossy Networks (LLNs) such as those based on IEEE 802.15.4 networks [11]. 6LoWPAN is the efficient extension of IPv6 into the wireless embedded domain, thus enabling end-to-end IP networking and features for a wide range of embedded applications. Issues such as power and duty cycle, multicast communications, mesh topologies, bandwidth and frame size have been extensively addressed.

3) UDP

The UDP protocol is used in between of the 6LoWPAN and the CoAP protocol in the transportation layer. It uses a simple transmission model avoiding a big overhead. Error correction mechanisms are used in other layers to ensure correct delivery of packets.

4) CoAP

The CoAP application protocol which runs on top of UDP layer is designed to easily translate to HTTP for simple integration with the Web. Its main characteristics are constrained machine-to-machine web protocol, simple proxy and caching capabilities, low header overhead and parsing complexity, and reliable unicast and multicast support [6]. Low overhead, multicast, efficiency and simplicity are extremely important for the Internet of Things.

B. WSN System designed without IP.

In this approach, we employ a much simpler network layer protocol for WSN connectivity in contrast to TCP/IP protocol suit. Furthermore we identify the absence of established transport/application

Table I. Comparison of the communication protocols used by the two different approaches; with IPv6 and without IP.

	Comparison of communication protocols of IPv6 and Non-IP	
	IPv6 over 6LoWPAN	***Non-IP (RIME)***
Layer	**Protocol**	**Protocol**
Application	CoAP	JSON
Transportation	UDP	RIME
Network	IPv6 – 6LoWPAN	RIME
Link – Physical	IEEE 802.15.4	IEEE 802.15.4

layers in such configurations, and we propose simple alternatives as described below. In the link and physical layer the 802.15.4 protocol is used.

1) RIME

RIME is Contiki's inbuilt network layer, which provides addressing and multi-hop networking primitives such as unicast and broadcast [14]. RIME addresses are 16 bits and should be manually configured in contrast to auto configurable IPv6 addresses. Despite the fact that IP is more comprehensive, RIME carries relatively less overhead in terms of message headers and occupies a less amount of bytes in RAM. Therefore RIME-stack is a good choice as the network layer for local subnetworks of an architecture for WSN.

2) JSON

Compared to the TCP/IP stack RIME does not provide transport layer service, therefore a combined transport and application layer has to handle the application data accordingly. As a well-established messaging format, which is developed and tested for constrained environments, JSON is used to handle the application messages. We constraint our messages' size to fit RIME packets so that special transport control is not needed.

C. Comparison of the two approaches.

From the deployment and programming perspectives, both of these approaches have their own pros and cons.

In the first approach, setting up the IP network is much simpler mainly due to the auto-configurable addressing, compared to the manually configurable RIME addresses. Furthermore, IP is more comprehensive in the context of internet, since its inbuilt support for auxiliary services, such as DHCP and DNS. In addition to, transport layer supports UDP and applications can easily adapt to the RESTFull nature of World Wide Web along with application layer protocols such as COAP.

On the non-IP approach we argue about the fact that WSN subnets can use much simpler communication stacks in local area. These sub-networks can be mediated by the gateway to communicate with the internet hosts. This can greatly reduce the memory burden on WSN nodes and communication overheads. Despite this advantage, RIME stack does not provide any transport layer support which makes the applications to take care of transport level aspects. We use a message oriented approach with JSON messaging to communicate with the Web.

We summarize the comparison of the two approaches in the Table II.

Table II. Comparison of 6LoWPAN vs non-IP scheme.

Comparison of IPv6 enabled vs Non-IP scheme	
IPv6 over 6LoWPAN	***Non-IP (RIME)***
Auto address configuration	Manual address configuration
Simple service discovery	Difficult service discovery
Larger overhead	Small overhead
Easy implementation	Difficult implementation
Easy integration with WEB	Difficult integration with WEB (custom gateway needed)

V. IMPLEMENTATION OF APPROACHES

In order to evaluate our approaches, we implemented a scenario in a wireless sensor network in the context of smart buildings. Our argument is which of the two approaches is more appropriate to use for implementing this scenario. Is IPv6-6LoWPAN along with CoAP more efficient than simple RIME? A comparison of these two approaches after we implemented them in the simulator and in our testbed is presented in the next section. Problems such as feasibility of technology, efficiency in the embedded systems, relation with the Internet of Things, schemes for interoperability and others have been tackled. We have chosen the scenario of profiling because it comprises of general communication patterns and components required by most of the smart building scenarios. The technology, the protocols and the systems we are evaluating behind this scenario could be applied to several other scenarios in the smart buildings systems.

A. Description of case study

We implemented the profiling scenario with the above described approaches (with IPv6 and without IP) to evaluate the feasibility, energy consumption, memory footprint and latency of each one of them. The scenario consists of a profiling system in a WSN which is able to identify people, take decisions according to their profile and make the resources of the WSN available to the Web.

Upon the arrival of a person in the proximity of an *agent* a message is transmitted to the profiling server through the *gateway*. The message contains the profile ID of the person hence the profile service decides whether to allow the access of the person in the room or not. In addition, it provides the *agents* and the *node-actuators* customized information based on the user's profile. According to the profile and the needs of each person, several actions take place such as switching on/off a light and turning on/off a fan. The interconnections and the communication pattern of the nodes of the scenario are shown in Fig. 2.

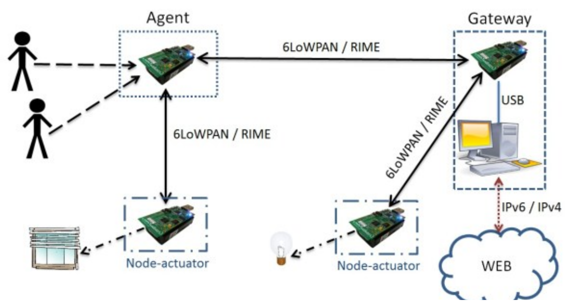

Figure 2. Interconnections and communication pattern in the profiling scenario.

In a distributed version of the profiling scenario the *agent* is designed to take decisions according to its inputs and drive the *node-actuators* directly. In this case it is propagating the data to the *gateway* so that they are accessible from the Web.

In a centralized version of the profiling scenario the *agent* is acting as a forwarder without taking any decisions or actions. Then it is the responsibility of the profiling server connected to the *gateway* to run the necessary services and drive the *node-actuators*.

A larger scale figure of the proposed scenario is shown in the Fig. 3, where multiple motes are being placed in each of the four rooms of a building comprising a wireless sensor network connected to the Web. The actuator nodes are placed in the corner of each room, driving the lights and the blinds of the windows. Next to each door of the building an *agent* is placed which is connected to the *node-actuators* and the *gateway* as well. The main *gateway* is handling all the information it is receiving from the *agent* and the Web while at the same time is able to drive the *node-actuators*.

B. *Experimental Setup*

1) *Hardware*

For our implementation we used TelosB motes [13] (see nodes in Fig.2 and Fig.3). The TelosB mote is an open source platform designed to enable cutting-edge experimentation. The mote supports; IEEE 802.15.4 compliant RF transceiver, 250 kbps data rate, integrated onboard antenna, the MSP430 micro-controller with 10kB RAM, low current consumption, programming and data collection via USB, sensor suite including integrated light temperature and humidity sensor.

As a *gateway* a TelosB mote acting as a border router was connected to a computer running an Ubuntu distribution of Linux.

In the *node-actuator*, we used the General I/O pins of the TelosB motes to drive a table lamp and a fan.

2) *Software*

To program our sensors we used the newest version 2.6 of Contiki OS [3]. For the simulations, the Cooja simulator developed by the Contiki community was used. Cooja allows large and small networks of Contiki motes to be simulated. Motes can be simulated at the hardware level, which is slower but allows precise inspection of the system behavior, or at a less detailed level, which is faster and allows simulation of larger networks. We used the Contiki's CoAP API to implement a CoAP client and server in the agent and node-actuators. We used JSON as the message format with the JSON support of Contiki in both approaches.

VI. EVALUATION

In this section we evaluate the two implemented approaches, based on their ROM memory footprint, energy consumption and latency per transaction. We implemented a use case of the profiling scenario interconnecting the three different components; *agent, gateway* and *node-actuator* according to Fig.2. The *agent* initiates a "transaction" requesting the profile from the server and transmitting the appropriate message to the *node-actuator*. We evaluated the above mentioned metrics performing 100 transactions for 20 times. The nodes were placed in 3 meters away from each other in an office environment. The transmission power of the nodes was set to their maximum. The evaluation of the two approaches took place at the same with the same conditions, allowing us to make accurate comparisons.

A. *Memory footprint*

Fig.4 shows the memory footprint in bytes of the ROM occupied by the different components of the system. As shown in the table, the ROM memory footprint of the non-IP approach is significantly smaller

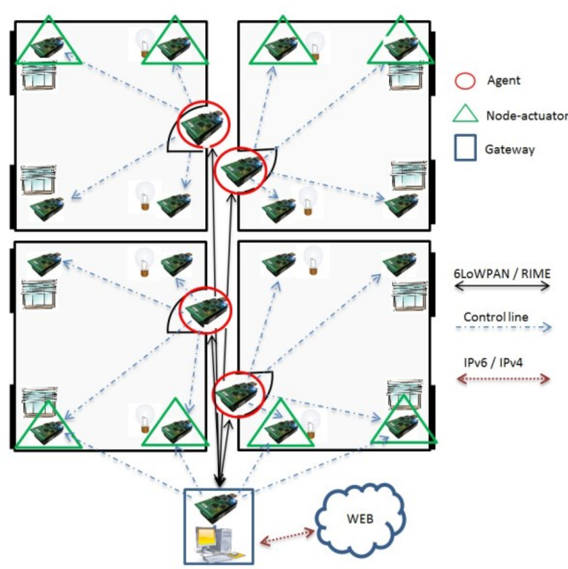

Figure 3. Wireless sensor network deployment of the profiling scenario in a smart building

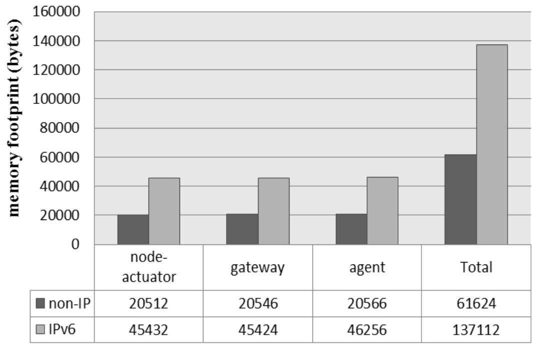

Figure 4. ROM memory footprint of the non-IP and the IPv6 approach. The non-IP has more than half less memory footprint.

than the IP-based implementation. This is due to the complexity of the COAP and 6LoWPAN implementations compared to the much simpler RIME stack.

B. *Energy consumption*

We used the software based online energy estimation mechanism proposed for Contiki [16] in order to calculate the energy consumptions of the three components. Fig.5 shows the total and the individual energy consumptions in micro-Joule (mJ). We observe that the IP-based approach consumes slightly less energy than the non-IP approach.

C. *Latency*

Fig. 6 shows the latency in seconds of a transaction in the two approaches. We calculate the latency based on the number of clock ticks spent for a transaction. We observe that the IP-based implementation performs faster.

VII. CONCLUSIONS

We presented two differnet design approaches for smart building systems using WSN. One approach is based on IPv6 while the other one is based on a custom non-IP networks stack called RIME. After the systems' architecture presentation we analyzed the advantages and disadvantages of the approaches. We showed the differences of the two approaches from the theoretical and implementation perspective.

We conclude that IPv6 approach is more advantageous for systems of large scale WSN. This is mainly due to the auto-configurable networking setup and the interoperability that the IPv6 technology provides with the standard protocols. Furthermore, the support in the transport layer with UDP, allows easy integration of Web applications via protocols such as CoAP.

Even though in the WSN the non-IP implementation is simpler, COAP/6LoWPAN design provides more flexibility when connectivity with the internet is needed. This is due to the fact that in the non-IP

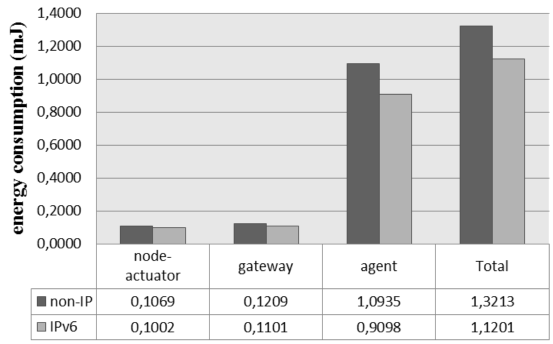

Figure 5. Energy consumption of the non-IP and the IPv6 approach. The IPv6 approach has slightly lower energy consumption.

approach, a “smart” custom gateway is needed to interconnect the simple RIME addresses of the WSN with the IP addresses of the Web. Additionally, without IP addresses in the WSN, the individual nodes are difficult to be accessed from outside the WSN they belong. Moreover, when a large number of nodes need to be deployed in a WSN, by using the non-IP approach the set up of the network would require many man hours to be deployed. For the fact that our lab does not possess hundreds of nodes to conduct large scale experiments, the investigation of the feasibility in the deployment of a large scale WSN can be considered as future work.

On the other hand non-IP design is good for small scale WSN because of its light complexity of the protocols and communication layers that are used. Furthermore, the small memory footprint of this design suites well to the constrained embedded devices. In addition, not all the devices of the IoT need necessarily to be connected to the internet favouring the non-IP implementation.

After our experimental evaluations of the profiling scenario we find out that the only drawback of the IP-based solution is the relatively large memory footprint requirement. However, with the advancement of

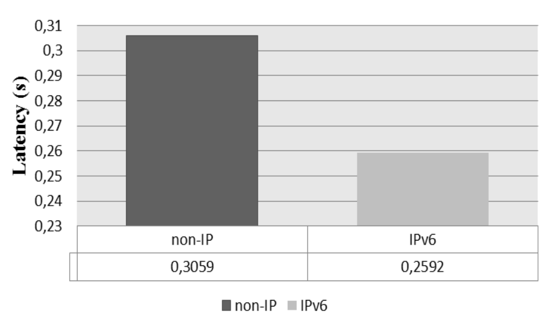

Figure 6. Latency per one transaction of the non-IP and the IPv6 approach. The IPv6 implementation clearly outperforms the non-IP.

hardware electronics, this discrepancy could be eliminated. Even though the design of 6LoWPAN is more advanced and its implementation is complex, it performs better due to its well established and well defined protocols. This results in outperforming the non-IP approach in terms of latency and energy consumption.

Finally we can conclude that considering the IoT vision, inter-connecting large number of embedded devices in WSN and connecting them as well with the Internet, IPv6 prevails as the most scalable and efficient mechanism.

ACKNOWLEDGMENT

This work has been partially supported by the IST Programme of the European Union under contract number ICT/FIRE/-STREP257466 (HOBNET)

REFERENCES

[1] D. Yazar, A. Dunkels, “Efficient application integration in IP-based sensor networks,” in BuildSys ’09 Proceedings of the First ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings, Berkeley California, 2009, pp. 43-48

[2] G. Montenegro, N. Kushalnagar, J. Hui, and D. Culler, “Transmission of IPv6 Packets over IEEE 802.15.4 Networks,” RFC 4944 (Proposed Standard), Internet Engineering Task Force, Sep. 2007

[3] http: //www. contiki-os.org

[4] R. T. Fielding and R. N. Taylor, “Principled Design of the Modern Web Architecture,” ACM Trans. Internet Technol., vol. 2, no. 2, 2002, pp. 115–150.

[5] M. Kovatsch S. Duquennoy A. Dunkels, “ A Low-Power CoAP for Contiki”, MASS '11 Proceedings of the 2011 IEEE Eighth International Conference on Mobile Ad-Hoc and Sensor Systems, 2011 pp. 855-860

[6] Z. Shelby, K. Hartke, C. Bormann, B. Frank, “Constrained Application Protocol (CoAP)”, draft-ietf-core-coap-11, 2012

[7] L. Schor, P. Sommer, R. Wattenhofer, “Towards a zero-configuration wireless sensor network architecture for smart buildings” ,Proceedings of the First ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings, 2009, pp. 31-36

[8] D. S. Tudose et al., “Home automation design using 6LoWPAN wireless sensor networks ” , 2011 International Conference on Distributed Computing in Sensor Systems and Workshops, Barcelona, 2011, pp. 1-6

[9] B. da Silva Campos, et al., "Design and construction of a wireless sensor and actuator network gateway based on 6LoWPAN", EUROCON - International Conference on Computer as a Tool, April 2011

[10] http: //www. ieee802.org/15/pub/TG4.html

[11] Zach Shelby and Carsten Bormann. “6LoWPAN: The Wireless Embedded Internet”, Wiley Publishing, 2010

[12] Mulligan, Geoff, "The 6LoWPAN architecture", EmNets '07: Proceedings of the 4th workshop on Embedded networked sensors, ACM, 2007

[13] TelosB motes datasheet: http://bullseye.xbow.com:81/Products/Product pdf_files/Wireless_pdf/TelosB_Datasheet.pdf

[14] Adam Dunkels. Rime - a lightweight layered communication stack for sensor networks. In Proceedings of the European Conference on Wireless Sensor Networks (EWSN), Poster/Demo session, Delft, The Netherlands, January 2007.

[15] N. Srivastava, "Challenges of Next-Generation Wireless Sensor Networks and its impact on Society", Journal of Telecommunications, Volume 1, Issue 1, February 2010, pp128-133.

[16] A. Dunkels, F. Osterlind, N. Tsiftes, Z. He, “ Software based on-line energy estimation for sensor node”, Proceeding of the 4th Workshop on embedded networked sensors, Cork Ireland, 2007