Internet Metadata Framework for Plug and Play Wireless Sensor Networks

Jongwoo Sung, Youngsoo Kim, Taehong Kim, Young-Joo Kim, Daeyoung Kim
Information and Communications University
Daejon, Korea
{jwsung, pineland, damiano, yjkim, kimd}@icu.ac.kr

***Abstract*—Metadata, self-describing data about sensor nodes and sensing data, is needed to understand different capabilities provided by heterogeneous sensor networks. However, storing metadata in resource constrained sensor nodes is infeasible in terms of increased memory costs and multi-hop communication overheads. More than this, sensor networks have difficulties with supporting XMLs and high level models often used for describing metadata. To overcome the problem we propose metadata frameworks that store metadata in distributed servers on the Internet instead of sensor nodes. The metadata overheads in sensor nodes are minimized because sensor nodes store only a few bytes long identifier as a key to look up a metadata on distributed servers. New sensor node discovery that finds specific sensor nodes using desiring attributes against retrieved metadata is also presented. In this paper we present a new approach for enabling plug and play wireless sensor networks.**

Keywords —sensor network plug and play, sensor metadata

I. INTRODUCTION

Although wireless sensor networks have been getting attractive in various application fields, they tended to be application specific, propriety solutions rather than general tasking. Heterogeneous sensor networks system requires individualized display and applications without standardized interfaces or capabilities. Thus applications and users are supposed to have priory information and configurations to access sensor networks. Thus it was impossible to use specific sensor networks or sensor nodes for multiple applications without a prior knowledge.

Plug and play wireless sensor networks, on the other hands, allow general network applications and users to access them in a plug and play manner. They provide a special function to self-describe their capabilities and to advertize them so that network applications or users come to have knowledge required to interface with them. A sensor metadata datasheet is an electronic description consisting of attributes, operations, interfacing message formats and services. The most advantage of using sensor metadata is that a prior knowledge needed in established wireless sensor networks can be replaced with well defined descriptions for late bindings. Thus, they are used for detections, identifications, and configurations of sensor nodes in a running time.

This work was supported by Korea Science & Engineering Foundation through the national research lab (NRL) Program.

Although contents of sensor metadata are varied for different applications, a typical sensor metadata may include sensor type, detailed sensing operations, sensor node message formats, sensing units, sensing accuracy, calibration information, various hardware/software descriptions, and so on. The contents of metadata are also varied for sensor node functionalities even if they are in one sensor network.

Standardizing sensor interfaces has been a popular issue in sensor related research communities. IEEE 1451 has worked on defining a standard for a networked smart transducer [2][3]. As a core of smart transducer, the IEEE 1451 standard defined a transducer electronic data sheets (TEDs) and its data format. ZigBee [4] defined application profiles as agreements for messages, message format and processing actions for interoperable devices. Both works store standard TEDs and profiles in target devices such as transducers or sensor nodes.

However, storing metadata in tiny sensor nodes is unfeasible in some points. Storing metadata in resource constrained sensor nodes results in increased storage costs and high energy consumptions during transferring them over error prone wireless networks. It becomes a more serious problem when sensor metadata is transferred via multi-hop communications which involve a number of sensor nodes in routing paths with. It consequently leads to high energy consumptions in wireless sensor networks which takes energy for the most important performance factors. In tiny sensor networks it is also highly restricted to support XMLs or high level models preferred by networked applications or users. In addition, sensor metadata have to be programmed at the time of sensor node developments.

A plug and play mechanism requires two tightly coupled phases; a new sensor node discovery and metadata retrieving for discovered sensor nodes. However, such a plug and play support is not a simple problem for multiple reasons. First, sensor nodes are too resource constrained to store large size of sensor metadata as mentioned above. Sensor network overhead caused by dealing with sensor metadata should be minimized. Second, since metadata is scattered across unreliable wireless network, sensor node discovery [4] [14], finding specific sensor nodes with some attributes described in metadata, becomes network problems. Even worse, sensor node discovery are

likely to be limited to few service attributes such as sensor types if attributes are queried against networked sensor nodes.

To alleviate the problems we propose a metadata framework that store metadata in distributed servers on the Internet instead of sensor nodes. This frees sensor nodes to store whole metadata. Instead, a sensor node stores a unique identifier as a special key indicating an appropriate metadata location. The identifiers are resolved by resolving systems. Since metadata is not stored in sensor nodes, established sensor node discovery that floods queries to all sensor nodes is not applicable. Instead, identifiers of new sensor nodes are registered to a base station using our simple discovery protocols so that capabilities of them are retrieved from distributed servers. Figure 1 shows our abstracted architecture of plug and play sensor networks without resolving systems.

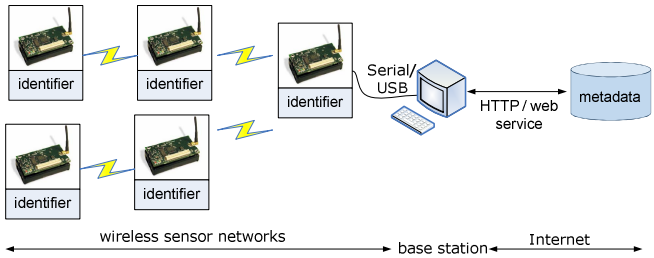

Figure 1 Architecture of plug and play sensor networks

A base station retrieves all registered metadata for sensor nodes, and stores them in it. Our approach changes networked discovery problems, finding specific sensor nodes using attributes based networked queries, to a local search problem against metadata stored in a resource rich base station. For retrieving all metadata the base station needs to simply know identifiers from newly connected sensor nodes. We provide a simple discovery protocol to maintain a list of identifiers in the base station. Then discovery of specific sensor nodes with desired targets is allowed by examining target metadata in the base station. The advantages of our approach are 1) minimizing metadata overhead in sensor nodes; 2) autonomous metadata retrievals without human intervention or prior configuration; 3) enabling effective sensor node discovery using metadata. We utilized 96 bits electronic produce codes (EPCs), defined by EPCglobal [5] for RFIDs, as globally unique identifiers, but we do limit identifiers to EPCs or certain standards for plug and play sensor networks.

However, our approach is not applicable for all cases. We assume a centralized communication model between sensor nodes and one base station rather than localized communication algorithms between sensor nodes. Thus, our approach is the best suitable when a central, networked base stations need metadata to know different sensor node capabilities and to find specific sensor nodes among them.

II. PROPOSED PLUG AND PLAY METHODOLOGY

Our sensor plug and play architecture consists of two different networks: a wireless sensor network and Internet systems. We assume that base stations are connected to Internet, and have role of retrieving sensor metadata for given sensor nodes. This can be easily extended to cover remote networked applications or users communicating with the base station in order to access sensor networks via web services or HTTP protocols as shown in Figure 2. Interfaces between the base station and user system are not address in detains in this paper. Some similar works are found at [12][13]. We put more focus on showing potential advantages of using our metadata framework to reduce metadata overhead in sensor networks.

A. Sensor networks

The basic concept of metadata has been used in literatures such as IEEE 1451 TED (Transducer Electronic Datasheet) [2][3], SensorML [7], TinyML [6], SensorWeb [8], and ZigBee [4]. Because they are specialized to their target system or applications, we define our own metadata for plug and play purpose. However, we do not restrict special metadata for plug and play sensor networks framework. Rather, we provide an architecture model to support metadata for plug and play sensor networks.

We define metadata as static, self describing data for explaining sensor node characteristics such as different capabilities, operations, and presentations. The metadata consists of four categorizes; *sensor*, *sensor node*, *sensor data*, and *application*, and each of them is expressed by {attributes, value} pairs. The *sensor* metadata describes the characteristic of sensors, and the *sensor node* metadata is used to explain hardware and software configurations of sensor nodes. Clients use this sensor attribute to interpret sensing data or to calibrate raw sensing values. As examples, *sensor node* block include battery characteristic, energy model, duty cycle information, network descriptions, and so on.

The *sensor data* metadata consists of events information and communication message representations for interfaces. The event information defines events list, threshold value for certain events, and their descriptions. The message representations give a list of available query/actuating interfacing commands provided by sensor nodes and their application level message formats. The *Application* metadata includes data required for applications. Table 1 shows typical sensor metadata examples.

Table 1 Classifications of sensor metadata

Metadata	Example
Sensor	Type, sensing unit, sensing range, sensitivity, coefficient, description, manufacturer, calibration information
Sensor node	Sensor list, hardware/software/network description, duty cycle range, battery model, service list
Sensor data	Event description, threshold value for event, interface formats, operations
Application	Image, icon, contact information

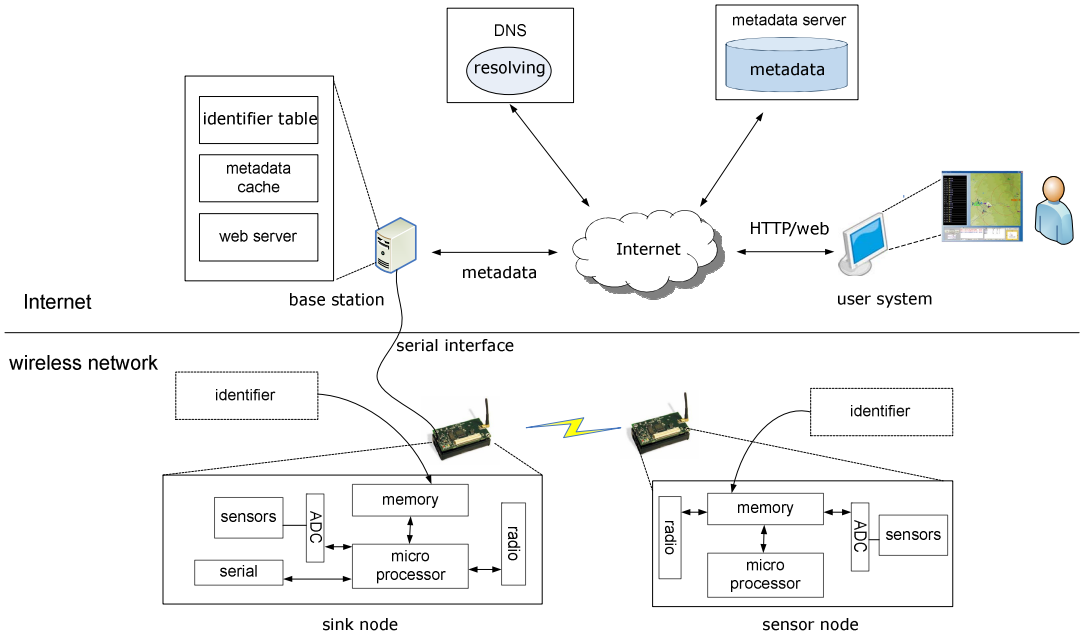

Figure 2 plug and play sensor network

Sensor metadata is often described in XMLs because it is preferred by networked applications or users for interoperability and human readability. Because XML description is around 10 times bigger than identical meanings expressed in a binary representation [9], we store them in distributed servers on the Internet. To retrieve appropriate locations of metadata for specific sensor nodes it is required to look up them. We provide a clue between sensor nodes and their metadata servers by assigning unique keys to sensor nodes, and provide resolution servers changing them to authorized locations of sensor metadata. The resolution process is explained later in more details.

Since identifiers are used as keys to query sensor metadata for desired sensor nodes they should be assured to be unique, but length of them should be small enough to be stored in sensor node. In addition, the identification needs to distinguish sensor nodes in serial level (each sensor node) and to support hierarchical structure like global organization, specific company, sensor nodes type, and serial numbers. To meet these requirements we adopt electronic product codes (EPCs), which were originally developed for RFID and EPCglobal Network [5]. Each sensor node only store small (e.g., 96 bits length) identifiers instead of large metadata. Although we used EPCs and EPC based resolutions for plug and play sensor networks, metadata concepts and sensor networks are originally not addressed by EPCglobal network [5]. In addition, there have been some works on using Internet servers for metadata, but they either depend on pre-assigned servers to lookup metadata [10] or manual searching [11] by users.

B. Sensor metadata Internet architecture

Sensor metadata server may be operated by manufacturers of sensor nodes. Sensor metadata servers maintain metadata for different sensor nodes and allow clients to query relevant metadata descriptions using given sensor node identifiers. Sensor metadata servers have operations for sensor metadata download and search via web servers or web service interfaces.

Because sensor metadata are scattered over distributed sensor metadata servers, mechanisms is needed to link a target sensor node to relevant locations of stored sensor metadata. We adopt domain name server (DNS) which converts URLs to IP address and vice versa for metadata server resolution purpose. We configured a domain name server (DNS) to maintain mapping from EPCs to corresponding metadata servers, and to return the location in pre-defined template. A DNS response has pointers to various locations of appropriate sensor metadata servers according to different services such as HTTP or web services. The resolution process using domain name server (DNS) is not unique. EPCglobal [5] used domain name service, called object name server (ONS) [1], to resolve EPCs to associated servers. However, they are used for neither sensor networks nor metadata.

DNS based resolving provides several advantages. Since DNS provide hierarchical structures, sensor metadata servers can be configured and maintained in hierarchical manners. Furthermore, resolutions are hidden to users or networks systems. The network address of DNS servers is automatically configured via DHCP, if it is preferred [5]. Users do not need to recognize the process. In this aspect, it can be considered as software black box or functions, which implements distributed translations and returns relevant address $f(N)$ of authorized metadata servers for given EPC N. Figure 3 illustrates a DNS based resolution process.

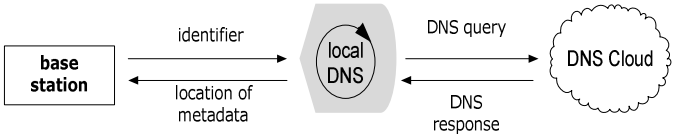

Figure 3 identifier resolution

C. *Sensor network plug and play*

To access sensor networks in plug and play manner, new sensor nodes need to be discovered by users or network applications. Our sensor node discovery protocol allows available sensor nodes to be registered to the base station. If new sensor nodes are joined to a sensor network, they send a report to a base station with a *registration* message. The messages include pairs of an identifier and an assigned network address. The base station maintains a list of identifications, which mean discovered sensor nodes. The table is continuously up-to-dated by our service discovery protocols.

Update procedure is same with the one of *registration*, but it is issued only when sensor nodes have any change (such as leaving) or update time, set by users, is expired. There are possibilities which sensor nodes leave the network without update messages, or registration failed with unexpected errors. To figure out this situation actively and quickly as soon as possible, base station sends actively *discovery* message to sensor networks in a periodic time.

Network applications or users, who want to access sensor nodes, first get an identifier table from the base station. Then, the identifiers are resolved and metadata is retrieved from servers after querying a resolution server. Communication between network applications or users and the base station can be implemented using standard web service or HTTP [12][13].

III. SIMULATION RESULT

Evaluations focused on metadata overhead and effects of our proposed scheme which uses only small identifiers in sensor nodes. We compare simulation results of metadata overhead and our proposed scheme using QualNet (version 4.5) simulator. We use IEEE 802.15.4 for PHY and MAC protocols, and ZigBee is configured for a routing layer. Since QualNet does not support a ZigBee routing we implemented a ZigBee routing layer. For wireless communication we used 2.4GHz channel frequency and 250Kb/s bit rate with O-QPSK modulation. Radio power is set to -10dB in 600m x 600m environments. Wireless sensor nodes are configured to construct ZigBee mesh networks consisting of one PAN coordinator and remaining devices. We assume sensor nodes are placed uniformly through the experiments, and we increased number of sensor nodes from 5 to 85 in 5 steps. As a control group wireless sensor networks are configured to store metadata and to transfer them. Size of metadata is set to 560 bytes and 210 bytes which were delivered using 8 and 3 continuous transmissions of 70 bytes simulation payload with 1 second interval.

We compare sensor node overheads caused by metadata in sensor nodes with our identification based metadata system. When metadata is stored in sensor nodes, it is needed to send all of them to a remote base station. On the other hands, our proposed scheme allows sensor nodes to store and transfer only identifiers. To support sensor metadata in a sensor network with 85 sensor nodes, total size of sensor metadata is over 45kbytes, but a proposed scheme allows only short identifiers to be transferred in sensor nodes. The identifiers consume only 22 % of sensor metadata size if 24 bytes identifiers are used instead of 560 bytes metadata. Packet transmission overhead for sensor metadata and identifiers are compared in Figure 4.

It becomes a more serious problem when sensor metadata is transmitted using multi-hop communications which involve with multiple intermediate nodes in routing paths with. This consequently leads high energy consumptions in wireless sensor networks which takes energy for the most important performance factors. We present energy consumptions in Figure 5. Figure 5 shows impacts of metadata transmissions using multi-hop wireless communications. Since intermediate sensor nodes need to forward metadata to a base station, actual energy consumptions are bigger than simply increased metadata sizes in Figure 4.

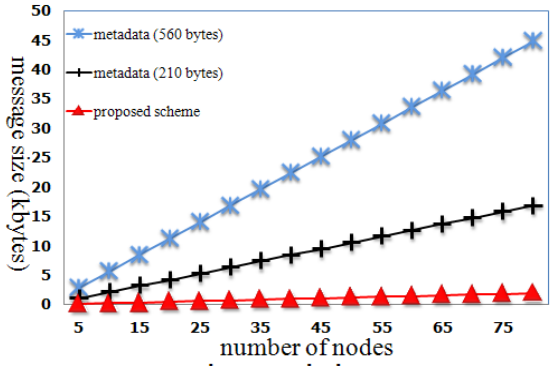

Figure 4 packet transmission

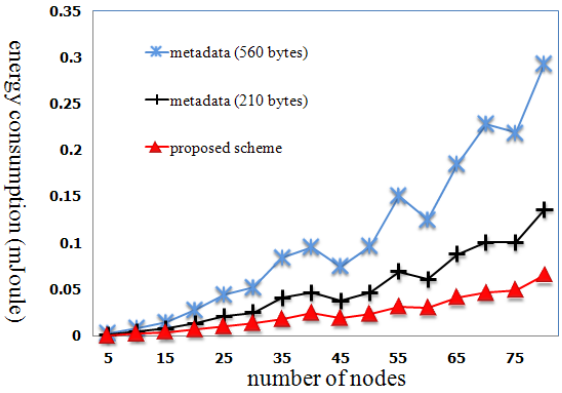

Figure 5 transmission energy

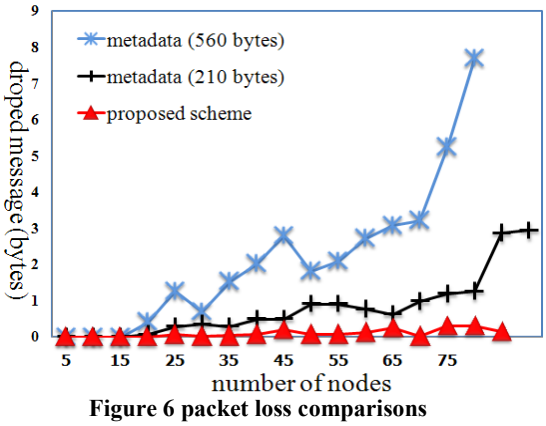

Figure 6 packet loss comparisons

Finally, we compare packet losses for difference cases; transferring sensor metadata and a proposed scheme using identifiers. Intuitively, our identifier approach shows better performance than whole metadata transmissions as depicted in Figure 6. As the number of nodes increases, packet losses are also increased. Sending large size of metadata causes more packet losses during transmitting metadata to the remote base station. In spite of increased number of nodes and potential packet loss rates, the proposed scheme using identifiers shows almost constant packet losses.

Generally service discovery in sensor networks finds a set of sensor nodes with specific features offered by sensor nodes [4][9][14]. Although we assume all sensor nodes send their metadata to a base station, it is not necessary for all sensor nodes to transfer them if well defined service discovery protocols are provided [4]. Only sensor nodes that meet service discovery queries need to transfer their metadata. However, if the numbers of qualified sensor nodes is large it will cause same problems; multiple response messages from sensor nodes are sent to a discovery requester in a short period of time, which lead to implode the base station.

Since a base station retrieves all registered metadata for sensor nodes and stores them in it, we changes networked discovery problems, finding specific sensor nodes using attributes based networked queries, to a local search problem against metadata stored in a resource rich base station. To simplify the simulation we do not consider metadata retrieving costs and a searching overhead against stored metadata in a base station. We focus on the fact that a base station and Internet architectures are relatively free from resource constraints compared to sensor networks.

IV. CONCLUSION

A plug and play is an important issue to deal with heterogeneous, application specific sensor networks. However, resource constrained wireless sensor networks require different considerations from established plug and play sensor approaches. Storing sensor metadata in severely resource constrained sensor nodes and transferring them via multi-hop communications increases memory cost and communication overheads. More than this, sensor networks have difficulties in supporting XML or similar high description models useful for applications or users. We proposed new approach to store them in distributed servers on the Internet instead of sensor nodes. It minimizes the metadata overhead and it also enables effective sensor node discovery by changing networked discovery problems to a local attribute search problem.

However, throughout this paper we limited our concerns to static metadata and a discovery support in the centralized sensor network architecture. Generalizing the proposed approach based on stimulated analysis will be our future work.

REFERENCES

[1] The Object Name Service Technical Manual, Version 0.5 (Beta) http://www.autoidlabs.org/whitepapers/MIT-AUTOID-TM-00d4.pdf.

[2] Paul Conway, Donal Heffernan, Brian O'Mara, Prof.Phil Burton, Tremont Miao, “IEEE 1451.2: An interpretation and example implementation”, 2000.

[3] Darold Wobschall, “Networked Sensor Monitoring Using the Universal IEEE 1451 Standard”, IEEE Instrumentation & Measurement Magazine 2008.

[4] ZigBee Alliance, http://www.zigbee.org

[5] EPCglobal and EPCglobal Network specification, web sites: http://epclgoballnc.org.

[6] Ota, Nathan, Kramer, William, T.C.,”TinyML: Meta-data for sWireless Networks,” http://kingkong.me.berkeley.edu/~nota/research/TinyML/project-paper-1.pdf.

[7] Sensor Model Language (SensorML), http://vast.uah.edu/SensorML

[8] Aman Kansal, Suman Nath, Jie Liu, and Feng Zhao, "SenseWeb: An Infrastructure for Shared Sensing," IEEE Multimedia. Vol. 14, No. 4, pp. 8-13, October-December 2007.

[9] Sameer Tilak, Kenneth Chiu, Nael B. Abu-Ghazaleh, Tony Fountain, “Dynamic Resource Discovery for Wireless Sensor Networks," IFIP International Symposium on Network-Centric Ubiquitous Systems (NCUS 2005).

[10] Kosuke Osaka, “An Architecture for Multiple, Large User-specific Data in a Networked RFID,” Master’s Thesis Academic Year 2007.

[11] “Sensors Plug&Play – The New Standard for Automated Sensor Measurements,” http://zone.ni.com/devzone/cda/tut/p/id/4047#toc3.

[12] Eugene Y. Song, Kang B. Lee, “STWS: A Unified Web Service for IEEE 1451 Smart Transducers,” IEEE transactions on instrument and measurement, Vol. 57, No. 8, August 2008.

[13] Marco Sgroi, Adam Wolisz, Alberto Sangiovanni-Vincentelli, Jan M. Rabaey, “A Service-Based Universal Application Interface for Ad Hoc Wireless Sensor and Actuator Networks”, whitepaper, UC Berkeley, 2004.

[14] Christian Frank, Vlado Handziski, Holger Karl, “Service Discovery in Wireless Sensor Networks,” TKN Technical Reports Series, March 2004.