Bluetooth-based Mobile Gateway for Wireless Sensor Network

Marko Pesko, Mitja Štular
Mobitel d.d.
Ljubljana, Slovenia
firstname.lastname@mobitel.si

Matevž Vučnik, Miha Smolnikar, Mihael Mohorčič
Department of Communication Systems
Jozef Stefan Institute
Ljubljana, Slovenia
firstname.lastname@ijs.si

Abstract—**Sensor nodes are being increasingly deployed for monitoring different phenomena in a stand-alone manner or as a part of larger wireless sensor networks (WSN). In addition to protocols for communication within the WSN, sensor nodes may also support communication with other devices. These devices either represent gateways for exposing data and metadata to external networks, or serve as connection points to WSN for on-site calibration and maintenance. Advanced mobile terminals appear particularly suitable for such manners. This paper presents the implementation of a mobile gateway consisting of a Samsung i8910 smartphone and a Versatile Sensor Node (VSN) connected via Bluetooth, whose role is to interface between WSN using proprietary networking protocol and mobile network. The data obtained from WSN can be enriched using smartphone's embedded sensors before being sent to the remote server. We show this on an example of geo-tagging with smartphone's GPS-based location data.**

Keywords-mobile gateway; wireless sensor networks; metadata; adaptation layer; testbed

I. INTRODUCTION

Interconnection with heterogeneous objects, sensing of various phenomena and gathering of large amounts of data are unawarely becoming part of our daily routine. Wireless Sensor Networks (WSNs) consisting of low power, low data-rate communicating sensor nodes can collect valuable information of various processes under observation over an area of interest [1]. Sensor nodes must be able to communicate with some distant location or be accessible from it. This is typically done via other nodes and through an appropriate gateway(s) to other networks. In such case only these need to support multiple communication interfaces.

In addition to data from sensors it is increasingly important for advanced applications to have an access also to the metadata providing spatial, temporal and thematic information required for discovering, analyzing and managing sensor data. As an example of basic metadata we use in this paper sensor nodes' identification numbers and their geo-locations.

There are various methods how and where metadata can be stored. It can be stored into individual device's memory or appropriate server. The most basic solution is pre-programming on sensor node or gateway during software development. More sophisticated is automatic metadata generation with pre-programmed specific software e.g. based on node's hardware serial numbers or other available information. Metadata programming on running nodes is not straightforward, thus it must be software supported and even enabled. Usually, when nodes are deployed randomly they are not inherently aware of their position unless using Global Positioning System (GPS) or other mostly less accurate radio location techniques [2]. In general, these should be avoided due to higher costs, energy consumption and processing power requirements. For manually deployed sensor networks, however, where their position is pre-planned, location information can be pre-programmed on the nodes during software programming. Still, for large deployment scenarios this is a difficult and time-consuming task that may lead to mistakes.

The aim of this paper is to present solution for nodes metadata storing based on mobile gateway concept [3] in combination with the server-side to avoid the need for pre-programming location metadata on sensor nodes. In the following, Section 2 discuses sensor network scenario and Section 3 describes the proposed mobile gateway structure. The test application on a real sensor network scenario is presented in Section 4, while Section 5 concludes the paper.

II. SENSOR NETWORK SCENARIO WITH MOBILE GATEWAY

The wireless sensor network scenario considered in this paper is depicted in Fig. 1. It consists of fixed sensor nodes, fixed gateway, supporting server side and additional sensor

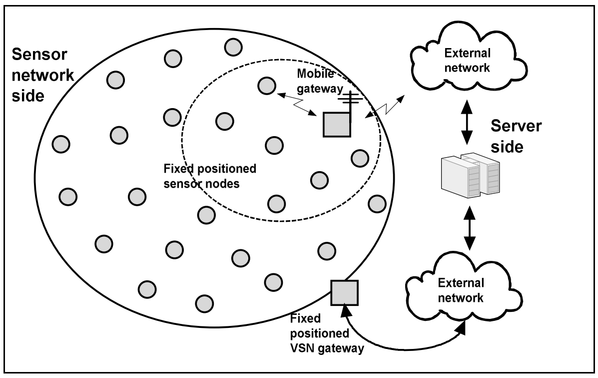

Figure 1. WSN scenario extended with a mobile gateway.

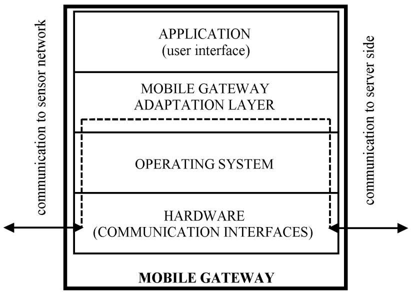

Figure 2. General mobile gateway structure.

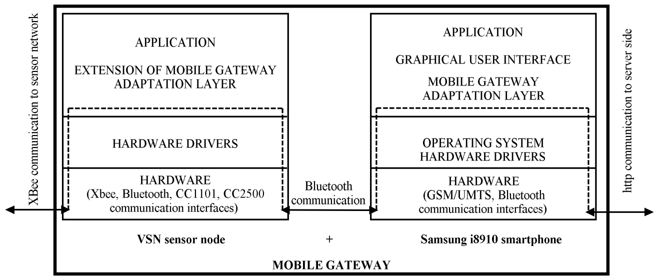

Figure 3. Practical implementation of a mobile gateway.

node(s) performing mobile gateway functions, making the entire setup more advanced than typical wireless sensor network.

It could be assumed that sensor nodes have pre-planned fixed deployment position. Using low-power, low data-rate wireless interfaces and sensor network protocols, these can communicate with fixed positioned gateway of the same network or any other device in the vicinity which supports the same communication interfaces and protocols. In general, fixed gateway is used to forward information from sensor nodes to the server side consisting of one or multiple servers handling all communication requests and appropriate number of database structures for storing data and metadata obtained from sensor nodes.

In this paper we introduce slightly different concept of sensor networks with the usage of mobile gateway which extends the ordinary gateway functionalities and can operate in parallel with existing gateway(s) of any compatible sensor network. Mobile gateway is designed for gathering and forwarding sensorial data from any available sensor node in the vicinity and additionally for metadata and parameter settings extraction or storing into node's memory. It is also well suited for the in-field sensor calibration, over-the-air (re)programming or other sensor network maintenance tasks. In practice, mobile gateway enables potential end-user or sensor network administrator to come near fully operating sensor network and connect to any sensor node having such connection supported and enabled. Based on security parameters of the sensor network and level of local access rights, end-user, administrator or specific mobile gateway advanced automatic software can thus access, retrieve or change nodes' shared capabilities.

III. Mobile Gateway Structure

A general mobile gateway structure that suits various sensor networks with homogeneous or heterogeneous nodes from the hardware and software point of view is depicted in Fig. 2. Such mobile gateway should integrate interfaces to sensor network technologies such as ZigBee, WirelessHART, 6LoWPAN and ANT on one hand and interfaces to other well established data communication technologies including Bluetooth, Wi-Fi and GSM/UMTS on the other.

In this study, we considered the WSN side of the testbed [4] consisting of Versatile Sensor Nodes (VSNs)[1] [5], running a proprietary networking protocol. In general, any mobile device with appropriate WSN communication interface(s) on one side and suitable communication interfaces towards the server side could be used for mobile gateway implementation. General purpose devices such as mobile phones, smartphones, PDAs, tablets and laptop computers, wireless routers, etc. all fit in this category due to multiple communication interfaces and large computing capabilities compared to sensor nodes. Mobile gateway characteristics such as number of communication interfaces, number of embedded sensors, memory capacity, processing unit(s), power supplies, programming/debugging interfaces, etc. are therefore mostly dependent on the hosting device.

As shown in Fig. 2, a mobile gateway has to implement protocols adaptation layer which is crucial to adapt/bridge various sensor networks nodes with various server side communication technologies.

A. *Mobile gateway hardware implementation*

Mobile gateway should ideally be hosted on a small and portable device, suitable for daily use, such as a smartphone or PDA. The majority of smartphones and PDAs currently do not support typical WSN or Near Field Communication (NFC) interfaces, but there already are some exceptions[2,3,4] which are announcing their comprehensive implementation.

For our practical implementation a typical smartphone was chosen for a hosting device. The temporarily lack of any WSN interfaces was overcome with the use of additional VSN sensor node, which extends smartphone adaptation layer and features appropriate sensor network interfaces as depicted in Fig. 3. In particular, the mobile gateway was realized on a Samsung i8910 smartphone[5] and a VSN sensor node communicating via Bluetooth. Until smartphones and other hosting devices start massively supporting selected WSN interfaces, the proposed approach seems a reasonable alternative solution.

---

[1] http://videolectures.net/wsn2010_mihelin_vsn/
[2] http://se-xperia.com/sony-ericsson-first-smartphone-to-integrate-ant-technology/
[3] http://www1.ericsson.com/res/docs/whitepapers/wp-50-billions.pdf
[4] http://mwc2011.techradar.com/2011/02/motorola-looking-to-deploy-nfc-in-smartphones/
[5] http://www.gsmarena.com/samsung_i8910_omnia_hd-2691.php

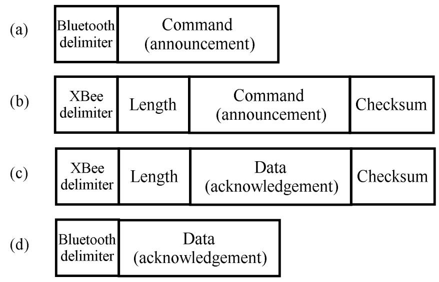

Figure 2. Data frame structures used to take over the control over the specific sensor node in the testbed.

Samsung i8910, used as mobile gateway primary hosting device, features 32-bit 600 MHz ARM Cortex-A8 processor, 256 MB of RAM, 16 GB internal storage, a microSD card slot for up to 32 GB memory, 2G and 3G cellular communication and GPRS, EDGE, 3G, WLAN, Bluetooth, infrared and USB data communication. This hosting device was virtually extended with the VSN sensor node, which is a computationally powerful, modular and open platform for sensor networks, enabling quick adaptation to various sensor networks. In its basic version it features a powerful 32-bit microcontroller with ARM Cortex-M3 core, with maximum clock frequency of 72 MHz, 64 kB of RAM and 512 kB of Flash memory, 128 kB of fast FRAM, miniSD card slot, and XBee, Bluetooth, CC1101, CC2500 communication modules. Optionally, VSN sensor node can also host GSM/GPRS, Ethernet, WLAN and other communication interfaces.

In this respect the implemented mobile gateway unique integrity suits general mobile gateway structure presented in Fig. 2 and at the same time overcomes the lack of WSN interfaces on the smartphone.

B. *Mobile gateway software implementation*

For the communication within and with the WSN testbed we use XBee RF interfaces and a proprietary networking protocol, whereas on the server side a HTTP connection via GPRS, EDGE, or 3G enabled mobile internet access is used. Fig. 3 presents also mobile gateway software distribution, which fundamentally consists of three modules realized on the application level. The first and the second module together form the mobile gateway adaptation layer for translation between HTTP and Bluetooth communication, running on the smartphone as Midlet (J2ME application), and between Bluetooth and XBee communication, running on VSN sensor node as C program application. The third software module is a graphical user interface (GUI) running only on the smartphone.

Adaptation layer monitors communication on all attached and relevant communication interfaces and reacts only on data received interrupts. It interprets all received frames, removes old and adds new headers, and forwards new frames to the appropriate destination interface or data to GUI. The implemented adaptation layer communicates with XBee module through an application programming interface (API) requiring data and commands communicated in frames in a predefined order. This frame headers and payloads principle is also used on top of successfully established Bluetooth communication between the VSN and the smartphone through a RFCOMM and L2CAP connection. However, for the connection to the server side HTTP GET requests are used instead of frames.

IV. Application

The application developed for the demonstration of the implemented mobile gateway concept was tested on scaled down version of a real WSN testbed consisting of fixed VSN sensor nodes and fixed VSN gateway. For the proof of concept there were up to eight VSN end-nodes constantly measuring and sending temperature, humidity, pressure and luminance information to the server side through a fixed VSN gateway. Mobile gateway was used to retrieve metadata (currently only node's ID number) from the specific sensor node, complement it with the GPS location data obtained by the mobile gateway, and send the combined information to the server side where it was stored in the database for the subsequent use by web applications.

A. *Mobile gateway - sensor network side*

For the described test case the data frame structures depicted in Fig. 4 were used. Frames (a) and (d) present Bluetooth communication frames used between the smartphone and the VSN extension module, whereas (b) and (c) are XBee frames for communication with sensor nodes in the mobile gateway vicinity through the XBee interface. It can be noticed that XBee frames have explicit length and checksum fields. This is not necessary for Bluetooth frames because we generate them on the application layer and use them across successfully established underlying Bluetooth connection, which itself implements FEC (Forward Error Correction) or ARQ scheme (Automatic Repeat Request).

Using the described frames, a mobile gateway temporarily takes over the control over specific sensor node in the WSN testbed for the time of information exchange. This is done by wirelessly sending presented data frames between the i8910 smartphone, its VSN extension module and sensor node in the testbed. Since wireless communications are interference sensitive, any data received prior to the predefined frames' start delimiters on all communicating devices is silently discarded. However, if known frame start delimiter is detected, frame is received and its payload is automatically extracted. Afterwards, payload is adapted (i.e. repacked) to the data frame specific for the interface through which data is sent towards its final destination.

For example, to build a mobile gateway's address table of all sensor nodes in the vicinity, the following sequence of frames from Fig. 4 needs to be used. First, announcement command frame (a), i.e. 'new coordinator is present' frame, is formed on the smartphone and sent to the VSN extension module of the mobile gateway. Announcement command is extracted from Bluetooth frame and Xbee command frame (b) is formed and broadcasted to all sensor nodes in the vicinity. All sensor nodes, which receive such frame, reply with the XBee acknowledgement frame (c). Data from the received acknowledgement frames is collected and stored on the VSN module of the mobile gateway until the last response is received. Then the Bluetooth data frame (d) with all addresses

```
String GETrequest = http://"server url"
                  + "?"
                  + "xbee=" + metadata
                  + "&lat=" + latitude
                  + "&lon=" + longitude
                  + "&alt=" + altitude
                  + "&sat=" + satelliteNumber;
```

Figure 3. HTTP GET request format used in test case.

of the sensor nodes in the vicinity is formed and retrieved to the smartphone as an answer to the announcement command. In this manner mobile gateway fills up address table of all sensor nodes in its communication range.

For metadata extraction all the addresses of available sensor nodes are presented in the GUI. User can select specific sensor node from which it wants to retrieve the metadata. After this a request is generated and sent to the selected sensor node in a similar way as the announcement command described above. The main distinction is that broadcast communication between the selected VSN sensor node and the VSN extension module is changed into unicast to preserve unselected sensor nodes in the sensor network from unnecessary energy consumption. In particular, metadata request is sent from the smartphone with a Bluetooth frame which is adapted in the VSN module to the XBee frame and sent further to the selected VSN sensor node. After the VSN sensor node replies the inverse protocol adaptation from XBee to Bluetooth frame is made on the VSN extension module, and Bluetooth frame with metadata is sent to the i8910 smartphone where metadata is extracted and adapted to the HTTP protocol. Meanwhile, the application on the smartphone asynchronously (i.e. whenever data is refreshed) reads GPS data from the embedded GPS sensor, so that current location of the mobile gateway is continuously updated in the application. In general, to combine metadata with correct location coordinates a mobile gateway has to be positioned near the selected sensor node. However, the distance between the VSN extension module and the testbed node should be similar to the GPS location uncertainty range, so the information obtained in such manner is considered adequate for most applications.

After both GPS position and metadata are available, mobile gateway combines them into the HTTP GET request depicted in Fig. 5 which is sent to the server side.

B. Mobile gateway - server side

The implementation of the server side was designed to be scalable with future development plans and is thus more general than needed for mobile gateway test application. The main component on the server side is a HTTP server listening on the specific port and periodically receiving measurements or other data from VSN nodes through the fixed or mobile gateway. Server multiplexes received data into MySQL database, which forms part of a Global Sensor Networks (GSN) middleware[6]. MySQL database structure is designed to efficiently manage data and metadata, supports the use of data mining tools on top of the received sensor data, and allows the extension of current concept with semantics.

GSN is a middleware for sensor networks and sensorial data flexible integration, discovery and web representation. It offers fast deployment and addition of custom platforms (e.g. sensors or sensor nodes) using specification in the XML format. GSN input streams from the HTTP server are handled by a CSV (comma-separated values) wrapper, which adjusts received data into the standard GSN data model. A CSV builder generates a CSV file with raw data which can be parsed by the GSN. The metadata is stored in the XML description of the virtual sensor which is also presented to the user on the web. Even more, GSN also enables web browsing between real time data, data related graphs or sensor nodes' locations on maps.

V. Conclusion

This paper discusses the mobile gateway concept for sensor nodes' metadata retrieval, its enrichment with geo-location information and sending combined information to the server side for the use by web applications. A mobile gateway was implemented as a device consisting of a Samsung i8910 smartphone complemented with a VSN extension module to overcome the lack of smartphone's WSN communication interfaces. The mobile gateway adaptation layer was designed for translation between communication protocols used in the test setup, and a simple GUI was developed for the interaction with a user. Communication in the test case was based on specific XBee and Bluetooth frames and HTTP GET requests. On the server side, an HTTP server, MySQL database, and a GSN middleware were implemented for metadata as well as sensorial data reception, storing and web representation.

Acknowledgments

This work has been in part funded by the European Community from the European Social Fund under the Operational Programme Human Resources Development for the period 2007 – 2013, and through the FP7 AgroSense Project (FP7-204472). The authors would also like to thank SensorLab members for their support and valuable contributions.

References

[1] M. Pesko, M. Mohorčič, M. Smolnikar, and M. Zajc, "Implementation of the Bluetooth Wireless Sensor Node," in *Eighteenth International Electrotechnical and Computer Science Conference, ERK 2009*, Portorož, Slovenia, 2009.

[2] L. Vidmar, M. Štular, and M. Pogačnik, "Positioning methods for future location based services," *Elektrotehniški vestnik,* vol. 78, pp. 23-29, 2011.

[3] F. Fummi, D. Quaglia, F. Ricciato, and M. Turolla, "Modeling and simulation of mobile gateways interacting with wireless sensor networks," in *Design, automation and test in Europe: Designers' forum (DATE'06),* Leuven, Belgium, 2006.

[4] M. Smolnikar, C. Fortuna, M. Vučnik, M. Mihelin, and M. Mohorčič, "Wireless Sensor Network Testbed on Public Lighting Infrastructure," in *EcoSense'11*, Belgrade, Serbia, 2011.

[5] M. Mihelin, M. Smolnikar, M. Mohorčič, and U. Platiše, "VSN – An Open Platform for Wireless Sensor Networks," presented at the The Nineteenth International Electrotechnical and Computer Science Conference, ERK 2010, Portorož, Slovenia, 2010.

[6] http://gsn.ijs.si/