Bluetooth Based Wireless Sensor Networks –Implementation Issues and Solutions

Srdjan Krco

Applied Research Lab, Ericsson Ireland

Invited paper

Abstract – Wireless sensor networks – networks of small devices equipped with sensors, microprocessor and wireless communication interfaces – are a technology that has gained a lot of interest lately. The broad spectrum of new and interesting applications, ranging from personal health-care to environmental monitoring and military applications, is proposed for such networks.

Various wireless technologies, like simple RF, Bluetooth, UWB or infrared might be used for communication between sensors. In this paper the main principles, applications and issues of Bluetooth based wireless sensor networks, as well as an implementation of a simple Bluetooth based sensor network are described. The main problems experienced during the implementation and applied solutions are presented.

1. Introduction

In [1], a vision, called ubiquitous computing, of the world where humans and computers were seamlessly united is described. The essence of the vision was the creation of environments saturated with computing and communication in an unobtrusive way. Recently, WWRF (Wireless World Research Forum) and ISTAG (IST Advisory Group) released their visions of the future communication networks [2], [3]. Both institutions envisage a vast number of various intelligent devices, embedded in the environment, sensing, monitoring and actuating the physical world, communicating with each other and humans.

Today, we are witnessing a rapid proliferation of various wireless devices with considerable computing power and fairly small size. These devices are still not, to paraphrase Marc Weiser [1], indistinguishable from the fabric of everyday life, but are becoming a part of that fabric (Bluetooth headset built as a part of jewellery for example).

Various sensors are already in a broad use today as part of different devices (temperature sensors in home or car heating system, smoke alarms, etc.) or as standalone devices connected to a network, usually to monitor industrial processes, equipment or installations.

The advancements in MEMS (micro-electrical-mechanical systems) technology, wireless communications and electric components have enabled development of small, low-power and low-cost devices, called smart sensor nodes, capable of performing various sensing tasks, processing data and communicating over wireless connections. Such devices, when organized into a network, present a powerful platform that can be used in many interesting applications, like health monitoring, security systems, detection of chemical agents in air and water etc. During the last few years significant research efforts were focused on development of required hardware resources [4], [5], [6], [7] and efficient communication protocols [8], [9], [10], [11], [12], [13], [14], to enable networking and collaboration of smart sensor nodes.

In the next section the main principles of wireless sensor networks are given and research issues are explained. Section 3 presents Bluetooth issues related to its use in sensor networks. An implementation of a Bluetooth based sensor network is described in section 4 along with some of the implementation issues and solutions. Section 5 concludes the paper.

2. Wireless sensor networks

Wireless sensor networks comprise number of small devices equipped with a sensing unit, microprocessor, wireless communication interface and power source. In contrast to the traditional sensor networks that are carefully planned and deployed to the predetermined positions, wireless sensor networks can be deployed in an ad-hoc manner.

Of course, such deployment requires adequate communication protocols that are able to organize the network automatically, without the need for human intervention.

Beside self-organization capability, another important feature of wireless sensor networks is collaboration of network nodes during the task execution. In contrast to the traditional sensor networks where all sensor data is gathered at a server and then analysed and fused, data processing and fusion is now performed by smart nodes themselves. Each node processes raw measurement data in order to decrease amount of data sent over wireless links and forwards only relevant parts to nodes responsible for data fusion.

Data-centric nature of the network is yet another specific characteristic of wireless sensor networks. As deployment of smart sensor nodes is not planned in advance and positions of nodes in the field are not determined, it could happen that some sensor nodes end in such positions that they either cannot perform required measurement or the error probability is high. That is why a redundant number of smart nodes observing the same phenomenon is deployed in the field. These nodes then communicate, collaborate and share data, thus ensuring better results (each sensor has its own view of the phenomenon – when these views are combined a better picture of the phenomenon is obtained). Having this in mind, it is more reasonable for a user to send a data request to all sensors monitoring the phenomenon than to send it to one specific sensor node. Using a multicast routing protocol to send messages to all relevant nodes would require unique addressing scheme in the network. However, due to the sheer number of sensors and user requirements (user needs

information about the phenomenon, does not need information about the phenomenon from a particular sensor), data-centric approach is used where sensors are designated using description of data they can provide instead of using unique IDs. Messages are directed to nodes using routing protocols that can find the route based on the data description contained in the message.
Power efficiency is one of the main requirements for all protocols and algorithms used in these networks. As power resources of each node are limited and required lifetime for many scenarios is measured in months and even years, it is of paramount importance to design system in such a way to ensure power savings whenever possible.

2.1. *Querying and tasking*

From the user point of view, querying and tasking are two main services provided by wireless sensor networks [14].
Queries are used when user requires only the current value of the observed phenomenon. As wireless sensor networks are data-centric networks, the user does not query a specific node for the information it might provide, but defines data (type, location, accuracy, time, etc.) he/she is interested in and requests it from all nodes that can provide the answer. For example, user can look for "temperature in the north region of the observed area" or for "location of all sensors where chemical agents are present and their level is above the threshold".
Tasking is a more complex operation and is used when a phenomenon has to be observed over a longer period of time. For example, a user can ask a sensor network to detect a specific type of vehicle in the area and to monitor its movement. In order to execute the task, different types of sensors have to collaborate: seismic to detect motion, video and audio to detect type of vehicle etc. Information about the vehicle trajectory is forwarded to the user.
Both queries and tasks are injected to the network by the gateway which also collects replies and forwards them to users.

2.2. *Gateway functionality*

Smart sensor nodes scattered in the field collect data and send it to users via gateway using multiple hop routes (Figure 1).

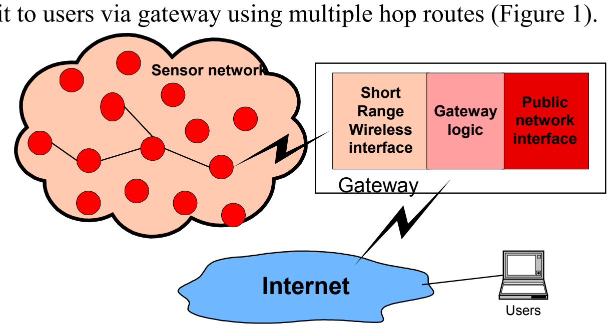

Figure 1. A Wireless Sensor Network

The main functions of a gateway are the following:

– Communication with sensor network - Short-range wireless communication is used (Bluetooth, UWB, RF, IR, etc.). Provides functions like discovery of smart sensor nodes, generic methods for sending and receiving data to and from sensors, routing, etc.;
– Gateway logic – Controls gateway interfaces and data flow to and from sensor network. It provides an abstraction level with API that describes the existing sensors and their characteristics; provides functions for uniform access to sensors regardless of their type, location or network topology, inject queries and tasks and collect replies;
– Communication with users – Gateway communicates with users or other sensor networks over the Internet, wide area networks (GPRS, UMTS), satellite or some short-range communication technology.

It is possible to build a hierarchy of gateways, i.e. to connect gateways described above to a backbone and then to provide a higher-level gateway that is used as a bridge to other networks and users.

2.3. *Applications*

The features described above ensure a wide range of applications for sensor networks. Some of the possible scenarios are given below:

- *Health monitoring* - Wireless sensor networks can be used in various ways to improve or enhance health care services. Monitoring of patients, health diagnostics, drug administration in hospitals, telemonitoring of human physiological data and tracking and monitoring doctors and patients inside a hospital are some of the possible scenarios [16], [17], [18], [19].
  Various sensors (blood pressure, heart monitoring, etc.) can be attached to the patient's body to collect physiological data that can be either stored locally (on a PDA or home PC) or forwarded directly to the hospital server or to the doctor in charge. There are several advantages of such monitoring: it is more comfortable for patients, doctors can have 24 hours access to patients and can better understand the patient's condition and last, but not the least, incurred expenses are much less than when such tests are done in hospitals.
  Wearable sensors can also be used to track patients and doctors in the hospital or to monitor and detect behaviour and health condition of elderly persons and children.
- *Environmental monitoring* - Fire detection, water pollution monitoring, tracking movements of birds, animals or insects, detection of chemical and biological agents are some of the examples of environmental applications of wireless sensor networks [20], [21], [22], [23].
  For example, numerous smart sensor nodes with temperature sensors on board can be dropped from an airplane over a remote forest. After successful landing, these devices will self-organise the network and will monitor temperature profile in the forest. As soon as the fire starts, that information along with the location of fire is transferred to the command centre that can act before the fire spreads to cover a large area.
- *Military and security* - The initial push towards wireless sensor network research came from military agencies. Military applications are various and vary from monitoring soldiers in the field, to tracking vehicles or enemy movement.
  Sensors attached to soldiers, vehicles and equipment can gather information about their condition and location to help planning activities on the battlefield.
  In case of nuclear or biological attacks, sensor fields can gather valuable information about the intensity, radiation

levels or type of chemical agents without exposing people to danger.

Seismic, acoustic and video sensors could be deployed to monitor critical terrain and approach routes or reconnaissance of enemy terrain and forces can be done.

- *Industrial safety* - Similar to personal health-care scenarios, wireless sensor networks can be used for "health-care" of buildings, bridges or highways [24]. In such scenarios thousands of various sensors are deployed in and around monitored object and relevant information is gathered and analysed in order to assess condition of the object after a natural or man-made disaster.

  Similarly, sensor can be used to monitor the status of different machines in factories, along with the air pollution or fire monitoring.
- *Other applications* - Home automation, smart environments, environmental control in office spaces, detecting car thefts, vehicle monitoring and tracking, interactive toys are examples of other possible applications [25], [26].

2.4. Research issues

Research issues are numerous and range from hardware issues to design of efficient communication protocols and distributed data processing algorithms. All solutions have to be, as said above, power conscious as well as fault tolerant, scalable, robust, with low production cost etc.

On the hardware side, wireless sensor networks require low-power, low-cost devices that accommodate powerful processor, a sensing unit, wireless communication interface and power source in a robust and tiny package. These devices have to work autonomously, to require no maintenance and to adapt to environment. Obviously these requirements are not easily met. Only since recently technology (MEMS) has become available that enables production of very small sensing units with low power consumption. Several solutions have been proposed so far [4], [5], [6], [7], [36], [37].

Physical layer issues range from power efficient transceiver design to modulation schemes.

MAC layer protocols have to support self-organization of a distributed network and to ensure fair medium access and collision avoidance. Different power modes have to be supported to enable nodes to save energy resources when possible, but without affecting network performance. Changes in network topology due to node malfunction or mobility have to be taken into account and dealt with automatically. Several MAC protocols have been proposed [12][13].

On the network level, routing protocols are required for dissemination of user queries and tasks. Since data-centric approach is used, the existing routing protocols for ad-hoc networks [27] cannot be used and new solutions, capable of routing messages based on data attributes are required. Several routing protocols [9], [11], [28] for different scenarios have been proposed so far.

Another important requirement for routing protocols in sensor networks is collaboration with data aggregation algorithms. Data aggregation is required to avoid network implosion (this may happen when many nodes answer the same query and send replies towards the gateway) and overlap problems in data-centric routing [35].

Based on the predefined methods, responsible nodes analyse gathered data and combine it into a set of meaningful information that is forwarded to users. Data aggregation reduces the amount of network load while preserving validity and amount of information. For certain applications it can be important to know the source of information (position) and in such cases that information has to be forwarded as well. The data aggregation process is shown in Figure 2.

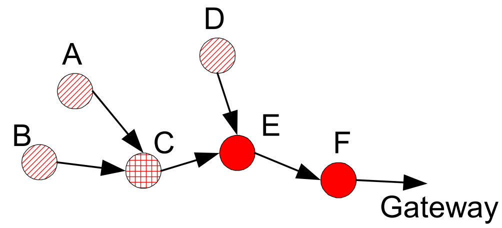

Figure 2. Data aggregation example

On the application level, a framework for attribute-based query definition, task building and their execution at each node as well as collection of replies is required [14], [38]. Sensor network management protocol has to support control of individual nodes (start/stop), network configuration updates, location information data exchange, network clustering, data aggregation rules etc.

Sensor network gateway has to provide tools and functions for presentation of network topology, services, characteristics etc. to users and to connect the network to other networks and users.

3. Bluetooth based sensor networks

Bluetooth is a low-cost, short-range wireless technology with small footprint, small power consumption, reasonable throughput and hence suitable for various small, battery-driven devices like mobile phones, PDAs, cameras, laptops etc. Development of the Bluetooth started several years ago with the intention to replace all sorts of cables used to connect different devices. In meantime the idea has evolved and Bluetooth is now developing not just as a point-to-point, but as a network technology as well.

Bluetooth has gone through periods of big hype when it was considered as the best short-range technology as well as through periods when it was considered a failure. However, the last year could be seen as the turning point year for Bluetooth. A lot of various Bluetooth devices and accessories appeared on the market, broad range of users is able to use it and first experiences are generally positive. The main challenge in front of Bluetooth developers now is to prove interoperability between different manufacturers' devices and to provide numerous interesting applications. An example of such applications are wireless sensor networks.

Low-cost (target price below 5EUR), low-power Bluetooth modules seem to meet the requirements of wireless sensor networks well. Ad-hoc connection establishment capability, reasonable throughput (up to 721kbit/s in uplink and 56kbit/s downlink), usage of frequency-hopping scheme with TDD (time division duplex) to minimize the impact of interference in the ISM band (2.4 GHz), existence of different power saving modes along with its availability and standardized specification are the main advantages of Bluetooth over other wireless technologies. However, there are several issues, like connection establishment delay and networking functionality, that have to be solved before Bluetooth can be deployed in large sensor networks.

In the following paragraphs, an initial implementation of a simple Bluetooth based sensor network is described. The main issues experienced during the implementation and solutions proposed are presented.

3.1. Sensor network related Bluetooth issues

Bluetooth operates in the 2.4GHz frequency band and uses frequency hopping spread spectrum technique. There are 79 channels, each 1MHz wide, available for hopping [29].

A Bluetooth device has to be member of a piconet to be able to communicate with other devices. A piconet is a collection of up to 8 devices that frequency hop together. Each piconet has one master, usually the device that initiated establishment of the piconet, and up to 7 slave devices. Master's Bluetooth address is used for definition of the frequency hopping sequence. Slave devices use the master's clock to synchronize their clocks to be able to hop simultaneously.

When a device wants to establish a piconet it has to perform Inquiry to discover other Bluetooth devices (these has to perform inquiry scanning at the same time) in the range. Inquiry procedure is defined in such a way to ensure that two devices will after some time visit the same frequency at the same time. When that happens, required information is exchanged (Bluetooth address and clock of the device that will be master of the piconet) and devices can use paging procedure to establish connection. Time required for communication establishment can be rather lengthy, taking on average around 5s (minimum is 0.00375s and maximum is 12.8s-33.28s). This delay can be the limiting factor for applications that require instant connection establishment.

When more than 7 devices needs to communicate there are two options. The first one is to put one or more devices into the park state. Bluetooth defines 3 low power modes: sniff, hold and park. When a device is in the park mode then it disassociates from the piconet, but still maintains timing synchronization with it. The master of the piconet periodically broadcasts beacons to invite the slave to rejoin the piconet or to allow the slave to request to rejoin. Of course, the slave can rejoin the piconet only if there are less than 7 slaves already in the piconet. If this is not the case, then master has to park one of the active slaves first. All these actions cause delays and for some applications it can be unacceptable (process control that requires immediate response from the command centre for example).

The other option is to build a scatternet. Scatternet consist of several piconets connected by devices participating in multiple piconets. These devices can be slaves in all piconets or master in one piconet and slave in other piconets. Using scatternets, higher throughput is available and multi-hop connections between devices in different piconets are possible. Unfortunately, hardware currently available on the market still does not support this functionality due to several reasons:

- Specification gives no way for a slave to demand park, hold or sniff mode, but can only request it from the master so there is no guarantee that the slave will be allowed to leave one piconet and join the other;
- Each time a device switch between piconets it might lose up to two slots for communication due to difference in piconet's clocks;
- Scheduling switches between piconets in such a way to maintain communication links with devices uninterrupted is very difficult, etc.

Several schemes for scatternet operation are proposed [30], [31], but so far none of them is implemented. A possible solution, before scatternet is supported by Bluetooth hardware, could be to perform switching between piconets on the application level.

Proposed scatternet building mechanisms assume that all nodes in the network are peer nodes. A mitigating circumstance for sensor networks is that gateway can be used to direct establishment of the scatternet. Using such centralized approach it is possible to generate more optimum network topology and solve scheduling, bandwidth allocation and routing easier [24].

However, there are some additional requests for scatternets in sensor networks that complicate scatternet building. Various sensor types produce different amount of data (video sensor and temperature sensor for example). If too many high-output sensors are connected to the same branch in the scatternet it can cause link congestion or buffer overflow in intermediate nodes. Hence, parameters like number of sensors, amount of data generated by sensor per measurement and buffer size have to be taken into account during building scatternet topology.

4. Sensor Network Implementation

The main goal of our implementation was to build a hardware platform and generic software solutions that can serve as the basis and a testbed for the research of wireless sensor network protocols. It supports ad-hoc deployment of sensors, sensor characteristics are automatically collected and presented in a structured way using XML, there are no limits in terms of sensor type and number of sensors, generic functions for querying sensors and collecting replies and the basis for attribute-based routing are provided. Software architecture is designed in such a way that new protocols can be added easily without affecting current functionality.

During implementation some compromises were made either to reduce development time or because of lack of appropriate hardware and software.

Implemented sensor network consists of several smart sensor nodes and a gateway. Each smart node can have several sensors and is equipped with a micro-controller and a Bluetooth radio module. Gateway has two wireless interfaces: Bluetooth for communication with sensors and GPRS for communication with users. Gateway and smart nodes are members of one piconet and hence maximum 7 smart nodes can exist simultaneously in the network.

4.1. Smart sensor node implementation

The smart sensor node comprises 3 functional blocks: sensing, data processing and communication (Figure 3).

One or more sensors can be attached to the micro controller. Temperature, heart monitor and smart fabric sensors were used.

Intel 8051 micro controller is responsible for the smart sensor node logic. An application is developed that gathers data from sensors and controls Bluetooth module and communication with the gateway. It also stores sensors' profiles and data. This controller was chosen because of our previous experience with it, which should have reduced development time.

Bluetooth functionality is implemented using an off-the-shelf Bluetooth device [32]. This device provides set of AT-like commands for control of Bluetooth connections over an RS232 interface.

It was selected because it enables quick prototyping, without the need for development of a Bluetooth stack suitable for the micro controller. However, there are several drawbacks of this solution:

– AT commands provide the basic control over the Bluetooth device, but do not support the full control of the Bluetooth protocol and settings. Two main problems are that SDP (service discovery protocol) is not supported and it is not possible to assign major and minor device class, and service class of the device. These deficiencies have stipulated some of the applied solutions, like algorithm for smart sensor node discovery;

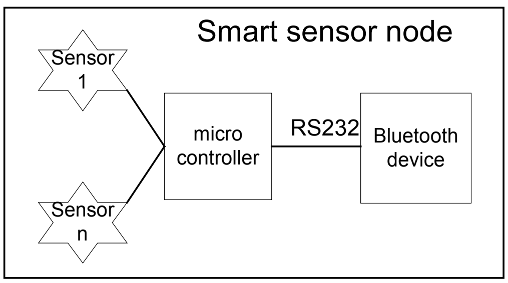

Figure 3. Smart sensor node implementation

– The size of the smart sensor node is not negligible (5cmx5cmx5cm), i.e. it is much bigger than required for many applications;
– Only serial port profile is supported.

However, using this off-the-shelf solution we were able to rapidly develop the prototype and to develop and test higher-level protocols.

4.2. Sensor Network Gateway Implementation

Gateway is implemented on a Pentium laptop with Linux operating system. As complete software functionality is developed in Java it can easily be transferred to other operating systems. A Bluetooth PCMCIA card is used for the Bluetooth interface. GPRS connection is built either over a GPRS PCMCIA card or using infrared connection with GPRS enabled mobile phone. We have also tried to connect GPRS phone using Bluetooth, but due to lack of Java PPP stack we were not able to control connection properly, so the first two solutions were used.

A Java Bluetooth stack [33] with standard Java Bluetooth API [34] was used. There are many Bluetooth stacks available on the market. They are developed for different operating systems, support different HCI (host controller interface) interfaces (serial port, USB, etc.) and provide different APIs. This makes it very difficult to make a proper choice of the stack and later to port developed application to different platforms. Java community undertook the first effort towards Bluetooth stack API standardization and it has recently resulted in the first standardized specification - the JSR-82 JABWT (Java API for Bluetooth Wireless Technology).

Gateway plays the role of the piconet's master in the sensor network. It controls establishment of the network, gathers information about the existing smart sensor nodes and sensors attached to them and provide access to them.

Software architecture of the gateway is shown in Figure 4.

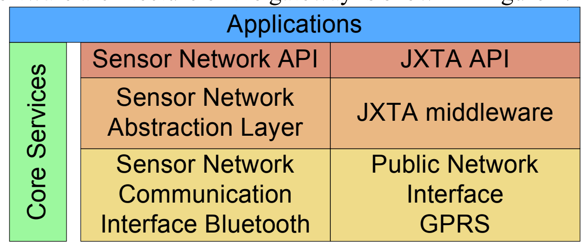

Figure 4. The architecture of the implemented gateway

A set of core services is developed to take care of common procedures and services required by all layers. Logging, scheduling, event subscription and services required for automatic application starting and restarting are supported.

Sensor network communication interface handles communication with sensors and can also control connections to mobile phone for GPRS connection or to local users that are using Bluetooth to access the network. Depending on the available hardware resources more than one sensor network can be attached to the gateway (using Bluetooth or any other communication interface).

The sensor network abstraction layer and its API are independent of the underlying communication technology and provide information about and access to all available sensors in the network.

Public network interface provide access to GPRS services of a public network.

JXTA middleware is used as a framework for establishment of a P2P network of gateway over a wide area network.

Application layer designates either local applications that reside on the gateway itself or remote applications that access the network over the GPRS network.

4.3. Discovery of smart sensor nodes

Smart sensor node discovery is the first procedure that is executed upon the gateway initialisation. Its goal is to discover all sensor nodes in the area and to build a list of sensor's characteristics and network topology. Afterwards, it is executed periodically to facilitate addition of new or removal of the existing sensors. The following algorithm is proposed.

When the gateway is initialised, it performs Bluetooth Inquiry procedure. When a new Bluetooth device is discovered, its major and minor device class are checked (these parameters are obtained along with the Bluetooth address and other parameters). These parameters are set by each smart node to define type of the device (Major class = "smart sensor node") and type(s) of attached sensors (Minor class = "temperature"). Service class field can be used to give some additional description of offered services (sensor accuracy, manufacturer, etc.). If discovered device is not smart node it is discarded. Otherwise, SDP is invoked and service database of the discovered smart node is searched for sensor services. As currently there is no specific sensor profile, then database is searched for the serial port profile connection parameters.

Once connection string is obtained from the device, Bluetooth link is established and data exchange with smart node can start.

Due to the deficiencies of the Bluetooth device we used in smart sensor nodes, it was not possible to implement proposed algorithm and some shortcuts were used. As SDP is not supported by these devices, connection string is built by concatenation of the Bluetooth address of the discovered device and server channel number used by the smart node's Bluetooth module (always set to 1). After building the connection string, an attempt is made to establish the connection. If establishment succeeds, the XMLSense protocol (the protocol we developed for data exchange with sensors) is invoked and communication with micro controller application attempted.

Obviously this solution causes establishment of unnecessary links (discovered device is not smart sensor node or is a smart sensor node, but does not have required sensors) which wastes energy and causes delays.

4.4. Communication with smart sensor nodes

On top of the Bluetooth link, a simple, but very flexible data exchange protocol, the XMLSense (XML based protocol for data exchange with sensors), has been developed for sending and receiving data to and from sensors.

Using two types of messages gateway can request either list of sensors attached to the particular smart sensor node or sensor data. Reply messages have very flexible structure and can relay information about any number or type of sensor. Sensor profiles are defined by sensor vendors, according to an XML scheme, and are sent as XML strings in reply messages. As all relevant information about a sensor is contained in its profile (sensor type, measuring unit, accuracy, manufacturer, calibration date, etc.), gateway can automatically build knowledge of sensor network and its characteristics, i.e. sensors can be deployed in an ad-hoc fashion.

Bluetooth links are maintained as long as the gateway and smart sensor node are in the range. This approach has its advantages and disadvantages. Good side is that as soon as an event happens on the sensor side, the sensor can send information to gateway about it. Drawbacks are that power resources are wasted on maintenance of communication link and it is not possible to have more than 7 smart nodes in one piconet.

The additional implementation problem, when JAVA APIs for Bluetooth is used, is that the power saving modes related functions are not supported.

4.5. Abstraction layer

Gateway's abstraction layer uses sensor profiles to create list of objects that represent each sensor in the network. Each object provides methods that enable sending and receiving data to and from sensor. Specifics of actual data transmission are hidden from users.

Applications can access sensor objects using queries that describe data they are interested in. Comparing data description in the query and profiles associated with each sensor, gateway determines which sensor can have an answer and sends data requests to them using methods provided by each object. Collected replies are formatted into a structured XML document and forwarded to users.

Currently this is a rather simple functionality because all smart nodes have direct communication links to the gateway and no routing is preformed. If larger network and scatternet topology were used, then similar functionality would have had to be provided by each master in the scatternet and appropriate attribute-based routing solutions would be required to disseminate queries. Data aggregation rules would have to be defined and could be easily combined with the XML structured description of the answer.

5. Conclusions

Wireless sensor networks are an interesting research area with many possible applications. They are based on collaborative effort of many small devices capable of communicating and processing data. There are still many open issues ranging from the choice of physical and MAC layer to design of routing and application level protocols.

Bluetooth is a possible choice for data communication in sensor networks. Good throughput, low-power, low-cost, standardized specification and hardware availability are Bluetooth advantages, while slow connection establishment and lack of scatternet support are some of the deficiencies.

An initial implementation of a Bluetooth based sensor network platform is presented. Implemented functionality and various problems experienced during the implementation are described. Implemented platform presents a good environment for further research and development of sensor network protocols and algorithms.

6. References

[1] Mark Weiser, "The Computer for the Twenty-First Century", Scientific American, 1991.

[2] Wireless World Research Forum, "Book of Visions", http://www.wireless-world-research.org

[3] Scenarios for Ambient Intelligence, EU IST Advisory Group

[4] http://www.janet.ucla.edu/WINS/lwim-innovative.htm

[5] J. M. Kahn, R. H. Katz and K. S. J. Pister, *"Mobile Networking for Smart Dust"*, ACM/IEEE Intl. Conf. on Mobile Computing and Networking (MobiCom 99), Seattle, WA, August 17-19, 1999.

[6] G.J. Pottie, W.J. Kaiser, *"Wireless Integrated Network Sensors"*, Communications of the ACM, May 2000, Vol. 43, No. 5

[7] J.M. Rabaey, M.J. Ammer, J.L. da Silva Jr., D. Patel, S. Roundy, *"PicoRadio Supports Ad Hoc Ultra-Low Power Wireless Networking"*, IEEE Computer Magazine, July 2000

[8] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. Culler, K. Pister, "*System Architecture Directions for Networked Sensors"*, Proceedings of the ASPLOS 2000

[9] C. Intanagonwiwat, R. Govindan and D. Estrin, "Directed diffusion: A scalable and robust communication paradigm for sensor networks", Proceedings of the Sixth Annual International Conference on Mobile Computing and Networking (MobiCOM '00), August 2000, Boston, Massachussetts

[10] W. Heinzelman, J. Kulik, and H. Balakrishnan, *"Adaptive Protocols for Information Dissemination in Wireless Sensor Networks"*, Proceedings of the 5th ACM/IEEE Mobicom Conference, Seattle, WA, August, 1999

[11] D. Braginsky, D. Estrin, *"Rumor Routing Algorithm for Sensor Networks"*, Proceedings of the Int. Conference on Distributed Computing Systems (ICDCS-22), November 2001

[12] W. Ye, J. Heidemann, D. Estrin, *"An Energy-Efficient MAC Protocol for Wireless Sensor Networks"*, Proceedings of the Infocom 02, New York June 2002

[13] K. Sohrabi, J. Gao, V. Ailawadhi, G.J. Pottie, *"Protocols for self-organization of a wireless sensor network"*, IEEE Personal Communications, October 2000
[14] C. Shen, C. Srisathapornphat, C. Jaikeeo, *"Sensor information networking architecture and applications"*, IEEE Personal Communications, August 2001
[15] N. Bulusu, D. Estrin, L. Girod, J. Heidemann, *"Scalable coordination for wireless sensor networks: self-configuring localization systems"*, International Symposium on Communication Theory and Applications (ISCTA 2001), Ambleside, UK, July 2001.
[16] J.M. Kahn, R.H. Katz, K.S.J. Pister, *"Next century challenges: mobile networking for smart dust*", Proceedings of the ACM MobiCom'99, Washington, USA, 1999, pp. 271–278.
[17] N. Noury, T. Herve, V. Rialle, G. Virone, E. Mercier, G. Morey, A. Moro, T. Porcheron, *"Monitoring behavior in home using a smart fall sensor"*, IEEE-EMBS Special Topic Conference on Microtechnologies in Medicine and Biology, October 2000, pp. 607–610.
[18] B.G. Celler et al., *"An instrumentation system for the remote monitoring of changes in functional health status of the elderly"*, International Conference IEEE-EMBS, New York, 1994, pp. 908–909.
[19] G. Coyle et al., *"Home telecare for the elderly"*, Journal of Telemedicine and Telecare 1 (1995) 183–184.
[20] P. Bonnet, J. Gehrke, P. Seshadri, *"Querying the physical world"*, IEEE Personal Communications (October 2000) 10–15.
[21] Cerpa, J. Elson, M. Hamilton, J. Zhao, *"Habitat monitoring: application driver for wireless communications technology"*, ACM SIGCOMM'2000, Costa Rica, April 2001.
[22] http://www.fao.org/sd/EIdirect/EIre0074.htm.
[23] http://www.alertsystems.org.
[24] V. Mehta, M. El Zarki, "Fixed Sensor Networks for Civil Infrastructure Monitoring – An Initial Study", Proceedings of the Medhoc 02, Sardegna, Italy, September 2002
[25] D. Estrin, R. Govindan, J. Heidemann, S. Kumar, "*Next century challenges: scalable coordination in sensor networks*", ACM MobiCom'99, Washington, USA, 1999
[26] D. Estrin, R. Govindan, J. Heidemann, *"Embedding the Internet"*, Communication ACM 43 (2000)
[27] C. Perkins, *"Ad Hoc Networking"*, Addison Wesley 2001.
[28] B. Krishnamachari, D. Estrin, S. Wicker, *"Modelling Data-Centric Routing in Wireless Sensor Networks"*, Proc. of the IEEE Infocom 02, New York, June 2002
[29] Bluetooth specification, http://www.bluetooth.com
[30] T. Salonidis, P. Bhagwat, L. Tassiulas, R. LaMaire, "*Distributed Topology Construction of Bluetooth Personal Area Networks"*
[31] N. Johansson, F. Alriksson, U. Jönsson, *"JUMP Mode - A Dynamic Window-based Scheduling Framework for Bluetooth Scatternets",* Proceeding of the Mobicom 2001, Rome, Italy, July 2001
[32] http://www.stollman.de
[33] http://www.atinav.com
[34] JSR-82 JAVA API for Bluetooth Wireless Technology, http://jcp.org/jsr/detail/082.jsp
[35] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, *"Wireless sensor networks: a survey"*, Computer Networks 38, Elsevier, 2002
[36] G. Asada, M. Dong, T.S. Lin, F. Newberg, G. Pottie, W.J. Kaiser, and H.O. Marcy, "Wireless Integrated Network Sensors: Low Power Systems on a Chip", Proceedings of the 1998 European Solid State Circuits Conference. Invited Paper
[37] Oliver Kasten, Marc Langheinrich, "*First Experiences with Bluetooth in the Smart-Its Distributed Sensor Network",* 2nd International Workshop on Ubiquitous Computing and Communications, Sept. 2001, Barcelona
[38] P. Bonnet, J. E. Gehrke, P. Seshadri, *"Towards Sensor Database Systems"*, Proceedings of the Second International Conference on Mobile Data Management. Hong Kong, January 2001.