Lucjan GROCHOWSKI

Politechnika Warszawska
Wydział Transportu
ul. Koszykowa 75, 00 662 Warszawa
lgr@it.pw.edu.pl

WIRELESS SENSOR NETWORKS AND THEIR LOGISTICS IMPLEMENTATIONS

Abstract:

In the paper the Wireless Sensor Networks, their operations frameworks, architecture and topologies are presented. An attention is focused on programmable Web service applicability of these sensor networks in implementations referred to two programming levels: low programming level of data sensors and high language programming level in distributed structures addressed to dedicated application. Wireless Sensor Network Systems implementations are illustrated in logistics implementation.

Key words: Computer Programmable Networking, WEB, Wireless Sensors, Logistics.

INTRODUCTION

Wireless Sensor Networks (WSN) are developed as specified area of wireless computing networks general purposes. They make distributed systems, usually composed of embedded devices, each equipped with processing unit, wireless communication interface and a number of sensors or actuators. The implementation of such networks is determined by type of sensors and type of implementations, hardware requirements and programming methods used. So, such sensor network can cover the wide area of implementations from satellite imaging systems, through various purposes measuring systems: seismic, thermal, acoustic..., spatial environment or meteorological monitoring to air quality stations, and security cameras. For growing WSN applications, recently, the special attention is paid to support the WSN services by programming methods and aided using the Web integrating the communication and coordination of the data among the WSN nodes and the trunk lines [1-3].

The transport is one of possible application the Wireless Sensor Networks, mainly, in the area of intelligent road systems and logistics where mobile and wireless communication technology allow in real-time to manage the global interconnections [4-6]. In these systems to process the information coming from many different data sources the SQL (Structure Query Language) stream computing platform usually is implemented. The SQL-stream solution reduces the time and cost to deploy sensor network systems aided by the databases to process large data streams. In these systems any number of data sources can be connected in real-time, mainly, using middleware approach making a standard of scalable sensor network applications. Such sensor application examples can be the Intelligent Transportation systems improving road safety, optimize transportation efficiency, reduce pollution, and providing data to support the national security issues [7-8]. Data exchange among in-vehicle and road side sensors produces a huge amount of real-time information able to be processed as a result of a progress in wireless technologies. Thus, e.g. speeding vehicles or local traffic congestions can be identified by the integrated traffic management systems assuring improvement the traffic smoothness. This is done as a result of:

- processing in real-time a big amount of information from many inputs,
- combining the sensor data, with GPS and data of other transportation databases,
- integrating a number of various mobile transport applications.

Presented wireless systems operate not only in real-time but also in multi-technology environment making so called the Mobile Resource Management (MRM) standard. It incorporates the wireless and GPS technologies with Internet and the other mobile applications by combining the geospatial sensor data with intelligent applications. They are referred to Internet mobile application portals and integrate the sensor data with the ERP management systems, which operate using the standard enterprise data. By this, the wireless sensor network implementations, if aided by the Web, can support the business cross of the globe with the data taken from remote places regardless of their physical location.

1. WIRELESS SENSOR NETWORKS SYSTEM AND PROGRAMMING

Wireless Sensor Networks applications can be categorized in regards of application goal, interaction pattern, mobility, space and time. Application goal resolves itself to sense only, or to sense-and-react under interacting conditions, one-to-many, many-to-many, many-to-one sensors. System mobility of such system can be referred to static or mobile nodes and sinks, while space to global and regional activities managed by periodic or triggered sensor operations. Typical sensor systems consists of hardware, system software assuring routing, localization, time synchronization, storage, reprogramming as well as the application software. Also, the sensor node is supported by volatile memory, external storage, CPU, application program, wireless ports, supply power source, actuators or sensors. Capacity of volatile memory range from 2 KB to 512 KB and it is used to store run-time data during execution of the program. Source program code is stored or in a dedicated memory ranging of 128 KB supported either in the external memory ranging several gigabytes, if it is needed. The sensor node wireless operations usually are arranged in the 2.4 GHz frequency band by the use of the communication protocols determined at the second layer of the OSI model. In contrary to standard computing, in the WSN operating systems a key role play the libraries of which data codes are combined with the application code to produce a binary execution codes. This allows to send the required data directly to the end user keeping under consideration the power supply energy allocated to each node as the power supply energy makes in the WSN system the critical issue of each node managing.

In the WSN systems the programming is addressed to operate the distributed data taken or from the numbers of nodes either from the data of the single nodes. This used to be managed by mentioned above the SQL solution assuring at the node interfaces to avoid unwanted inter-node data interactions.

Thus, programming procedures can be separately considered in two areas: communication and implementation. First one covers mainly addressing and data accessing, while second one covers real operations: functional, sequential, rule-based, even driven, SQL-like or special purposes operations. The WSN nodes can’t perform any useful task, if they are left alone; to accomplish a higher-level implementation goal a given WSN system has to assure collaboration and coordination of numerous devices. In communication this is referred to node operations: mainly, multi-hop connected and multi-hope disconnected WSN group of the nodes. Therefore, the communication scope can be defined as a set of the data exchange nodes, that accomplishes a required application process using the nodes making:

- physical neighborhood of a given node,
- multi-hop node group: connected or non-connected.

For such specified purposes a term of the Active Messages is introduced. It is defined by a set of interfaces providing basic communication primitives, which for WSN purposes are described in event-driven language, extracted from C language. A new language is called the nesC language, and is used mainly to programming the interconnections of interacting system components provided by the data interfaces.

An example of interface described in the nesC Active Message formalism is shown below:

```
interface AMSend
{
    command error_t send ( am_addr_t addr, message_t * msg, uint8_t len );
    command error_t cancel ( message_t * msg );
    event void sendDone ( message_t * msg, error_t error );
    command uint8_t maxPayloadLength ();
    command void * getPayload ( message_t * msg, uint8_t len );
}
```

The Active Message interfaces meet conditions of the MAC-sublevel of the OSI model and make a platform to both the single-hop cast and the broadcast operations. In such platform, additionally, to overcome the some limits by the expansion of standard Active Message the interfaces adjusted to the multi-hop protocols for data collection and distribution are implemented.

Independently from presented above the Connected Multi-hop group method other programming approaches also is used. One of this is called the EnviroSuite group, and in a given application it allows to build an object-based programming framework for monitoring the program changes. In the environment specified by the object approach the objects represent dynamically created the physical entities as a result of the physical events appearing at the moment of event detection. Such entities are addressed to non-connected multi-hop group and they are automatically destroyed at the moment of system moving out of sensing range. Thus, for a each node the objects can create the Logical Neighborhoods - programming abstractions, which redefine the neighborhood according to actually required logical properties being independent on physical location of the objects. For those purposes a declarative programming language called **Spidey Neighborhoods** is used, it makes extension of existing WSN languages and operates by using Application Programming Interfaces associated with given nodes.

To facilitate such programming operations cross a whole WSN system, also additionally, so called, the TinyDB sub-systems are introduced. Their role is to optimize WSN system energy consumption by controlling where, when, and how often the data are sampled. For such purposes in the form of “injection” into the network, the user is allowed to submit SQL-like requirements on optimized data to the base station. By this he can pay just an attention to the data operations without specifying how to do so, as “injection” is equivalent building a routing tree, which span all nodes across of given network. In consequence, at the base station the volume data to be processed is reduced.

In the WSN implementations important roles play the data communication and sensor addressing. Two classes of addressing are distinguished:

- physical addressing: the given nodes are identified by statically assigned identifiers,
- logical addressing: the given nodes are identified by application program.

Similarly, for program purposes the communication can be divided into explicit communication and implicit communication related to the possibilities of used languages. This allows for the local nodes, group of nodes and global data processing to build up the **Data Access Model of the WSN**, which covers processing of:

- databases: WSN data are referred to a relational database and SQL operations,
- data sharing: WSN data are shared as the remotely accessible variables,
- mobile codes: WSN data are accessed locally to a node,
- message passing: among the nodes the WSN data are exchanged in the form of communicates.

Presented above low level programming of primitives of the WSN makes the necessary initial phase of real programming of WSN services, these later are programmed using high level languages, typically the object programming languages as e.g. the Java is well adjusted to network programming, especially, in the Web environment. The results of low level programming of WSN primitives make the "input data" to higher level service programming in the Java language, usually, applying the SOAP / WebService framework. Initially programmed and pre-processed from WSN systems data have rather aggregated character.

As standard programming methods related to high level languages, mainly, in Java language are widely reported, in presented paper they will be not discussed.

2. WEB SUPPORT OF WSN SYSTEMS, IMPLEMENTATION EXAMPLES

Functionalities of the current WSN implementations can be substantially upgraded if they are supported by the Web systems. Such implementations are characterized the Service Oriented Architecture profile based on various system solutions: as so called Embedded Networks [9], or as Web Services Eventing architectures [10], or as Web Management of WSN [11] either the systems of Small Programmable Objects integrated in the Web [12]. Among these solutions the last one approach to handle Service Oriented Architectures seems to be not only elastic but also promising for a large group of the WSN implementations and from these reasons as the representative system will be presented below.

This systems integrating in the Web small programmable objects characterize themselves the multilayer architecture referred to the use of Standard and Micro Editions of Java language [12-14]. Data operations and system functionality of such system is assured by the programmable objects fulfilling requirements of Web Services aided by XML scripts used to service managing. In the integrated Web programmable environment in this approach each sensor node is able to use the Java Servlet assuring a good interoperability without expanded gateway hierarchy. Main system advantage makes using the Discovery Protocol, that is able to detect and to register the new sensors as well as to use the Phone for direct communication with the sensor nodes. Also, such system architecture allows to integrate a number of sensor sub- networks physically located in different areas, regardless of sub-networks technologies, or configurations. Architecture of such system is shown in fig.1, it characterize itself a layer hierarchy, in which at the lowest system layer the programmable objects are located.

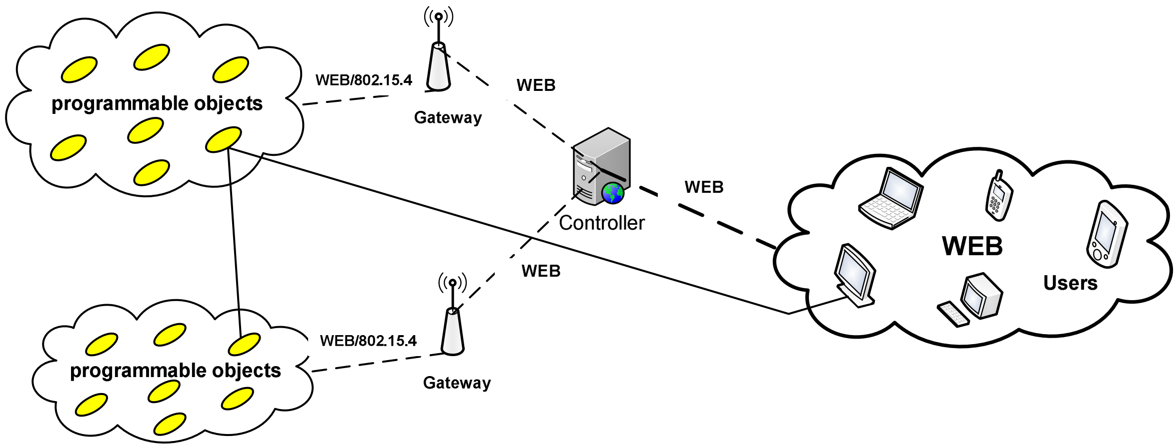

Fig.1 System Architecture - Web interworking structure comprising sub-sensor networks supporting functional programmable objects.

As it is shown in fig.1 each wireless sensor is running software component supported by sensor detecting interface and special communication protocols extending the HTTP protocol. Similarly to commonly used in Java language programming the terms servlet or midlet for sensor purposes a term of a **senselet** is introduced. The **senselet** has meaning the specific programmable function to be executed in the Wireless Sensor Network.

In the system shown in fig.1 the controller plays control and administration roles using three characteristic layers. The lowest layer comprises the WSN controller, which administrating the physical sub-network of programmable objects , the proxy - to store the MAC addresses of registered programmable objects and to list the available senselets. At this layer also the routing operations are located. The second one, the middle layer plays a role of process authentication, while the third one, the highest layer makes the real Web environment. All these layers form the stack of characteristic operations of system controller. When a gateway receives a broadcast message from a programmable object it executes the GETINFO senselet, parses the returned XML file and stores the extracted list of senselets.

By similar way to stated above, the specified stack of programmable object can be introduced, it also consists of the three layers: the senselets making the highest one layer, second one middle layer making the WSN runtime object environment together with so called small HTTP server, and the lowest layer being well known the Java Virtual Machine.

The system routing is the one that handles the virtual topology setups among a number of network physical programmable objects [14]. Built up the virtual network links can operate or among nodes inside a single programmable object network either among programmable object networks physically differentiated by their features. In considered networks special options can make the operations among selected WSN nodes of large WSN networks. Mentioned above the WSN Runtime module, in turn, makes the end point of the WSN Controller. In this module to reach a given implementation goal all required functionalities are invoked in each one programmable object.

WSN in Logistic Application

For transport logistic services, in recent time, the challenge makes growing competition at increased atomization of goods to be transferred. Both these factors are imposed by:

- changes in delivery chains,
- operating by mean the virtual enterprises allowing just-in-time management,
- increasing traffic and congestion on the transport routes,
- increased environmental awareness.

In such service environment using traditional central planning is difficult and inefficient, so the WSN technology can be an important alternative improving the effectiveness of logistic services. For these purposes a term the **intelligent transport goods** is introduced. It determines technically imposed on the goods the activities related to their transfer, mainly, choosing the transport vehicles under condition of expected arrival time at given destination, risk not to arrive in time, suitability of transport vehicle for the dedicated transport purposes and transport expenses. All these factors can be easily monitored and controlled using the WSN technology, especially, if used system is supported by the RFID tags or more advanced JavaCards recognized by the computer readers supported by access to Internet e.g. [8].

The **intelligent transport goods** approach based on the use of the WSN technology assures to the goods mutual communication and communication of the goods with the vehicle's board computer. Schematically this is demonstrated in fig. 2.

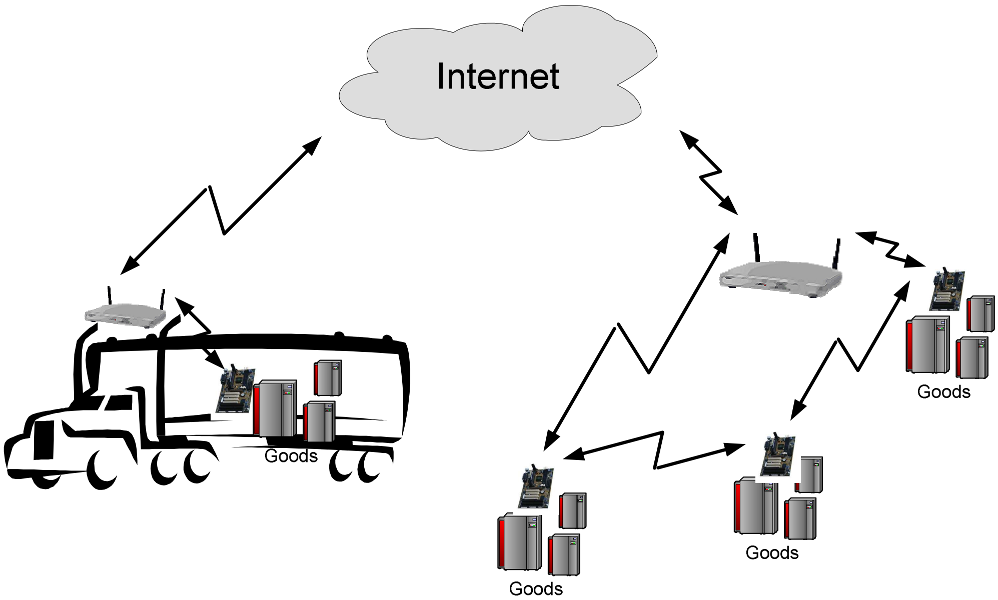

Fig. 2 An idea to use the wireless sensors in logistic management.

Shown in fig.2 the transport goods by mean wireless sensors can "advise" to the vehicle's board computer how to configure and monitoring requested at each moment of the time the best transport conditions. On the transport vehicles this can be managed by full system identification of the goods and intercepting from the Internet the transport managing commands. Handling the goods by that manner assure the WSN dedicated network consisting of a given number of vehicle and good sensors.

Other manner to handle logistic operations by the use of the WSN so called the **Agent Representation of Transport Goods** are. In the computer networks this is given by the autonomous program agents, which identify the goods, the good attributes and associated with them services described by a given program. Such data packet is able to migrate cross of the Internet.

For logistic purposes the agents usually fulfill condition of the FIPA standard [9], in which the agents are combined with wireless sensor networks and allow to transfer the agent codes into agent platform covering the transport vehicles and the goods. In transport environment under correctly specified conditions such mobile agents are able to migrate to the good data bases making a support of goods handling and related services cross of the world. Advantage of such solution is high speed of operations, disadvantage advanced programming and system platforms in WEB/computer networks.

Other one area implementing the WSN technology for logistics purposes the advanced CRC systems are [14].

3. CONCLUSIONS

In the paper the principles, system architecture and topologies of the Wireless Network Sensors (WNS) were presented. Especial attention was paid to the programming methods used in the sensor system implementations. Related programming was categorized into two groups: the low level language programming addressed to managing the data of the sensor networks and the high language level programming describing real service implementation referred to pre-processed aggregated sensor data. These data are used in programming of dedicated implementations of services by mean the distributed programming in the Web environment. Mostly, programming is referred to the use of the Java language standards or more universal standards as .NET and handling the services by the SOAP / WebService approach.

Last part of the paper demonstrates the examples of service implementations of the Wireless Sensor Networks in logistics. They are referred to implementations based on the intelligent transport goods approach and introducing the programmable agent representation to transported goods. Both of these implementations operate using sensor networks in wireless environment and if supported by the Web significantly upgrade the level of services.

This allows to state that the wireless sensor networks make powerful technology with the potential to be truly universal computing environment. However, despite of the advancements allowing building the network oriented smaller devices with stronger computation power, and new possibilities, programming of WSN systems remains the weakest point of wider deployments. So far, the WSN programming platforms are as much advanced as sensors assure acceptable and predictable levels of performance and reliability.

BIBLIOGRAPHY

[1] Mottola L., Picco G.P.: Programming Wireless Sensor Networks…,Fundamental Concepts and State of the Art - this paper will be published in ACM Computing Surveys, www.sics.se/~luca/papers/mottola - 2010 .

[2] Reumerman H-J. et al.: The application-based clustering concept and requirements for intervehicular networks, IEEE Communication Magazine, April 2005, pp. 108-113,

[3] Grochowski L.: Programowanie komponentowe w środowisku WEB, Studia Informatyczne, 2010, 1, str. 23-33.

[4] Gaynor M. et al, Integrating wireless sensor networks with the Grid, IEEE Internet Computing (special issue on wireless grid) July/August 2004, 8, pp. 32-39.

[5] Becker M. et al.,.: Challenges of Applying Wireless Sensor Networks in Logistics, www.citeseerx.ist.psu.edu/viewdoc/download?, 2010.

[6] Grochowski L., Programmable Services in Transport Implementations, International Conference TRANSCOMP, monograph 2008, pp.91-98, Zakopane.

[7] Nekovee M.: Sensor networks on the road: the promises and challenges of vehicular ad hoc networks and grids, www.semanticfrid.org/ubinesc/ubi-v1.pdf, 2010

[8] Varsheney U.,: Vehicular mobile commerce, IEEE Computer, Magazine Online, December 2004.

[9] Becker M., et al.,.: Analysis of Mobile Agents considering the Fan Out – Mobile Agents for Autonomous Logistics, IEEE International Conference on Service Operations and Logistics, and Informatics, 2006, Shanghai, China.

[10] Scholz A., et al., Service Oriented Architectures adapted for Embedded Networks, Proc.of 2009 7th IEEE International Conference on Industrial Informatics (INDIN 2009) pp. 599 - 605

[11] Nissanka B. Priyantha, et al., Tiny Web Services: Design and Implementation of Interoperable and Evolvable Sensor Networks, Proc. of SenSys'08, November 5–7, 2008, Raleigh, North Carolina, USA
[12] Orestis Akribopoulos et al., A Web services-oriented Architecture for Integrating Small Programmable Objects in the Web of Things, Proc. of 2010 Developments in E-systems Engineering IEEE, pp.70 – 75.
[13] Baumgartner, et al., Virtualising Testbeds to Support Large-Scale Reconfigurable Experimental Facilities. Proc. of 7th European Conference on Wireless Sensor Networks (EWSN 2010), Coimbra, Portugal, 2010. Lecture Notes in Computer Science, v. 5970, pp. 210-223 (Springer-Verlag).
[14] Apostolos Malatras, et al.. Web Enabled Wireless Sensor Networks for Facilities Management, IEEE System Journal, v.2, 4, December 2008

SIECI BEZPRZEWODOWYCH SENSORÓW I ICH ZASTOSOWANIA W LOGISTYCE

Streszczenie:

W artykule zostały zaprezentowane bezprzewodowe sieci sensorów wraz z ich szkieletowymi rozwiązaniami systemowymi, architekturą i topologiami połączeń. Uwaga skupia się na programowalnych aplikacjach Web opartych o wykorzystanie sensorów bezprzewodowych w odniesieniu do dwu poziomów programowania: niższego obejmującego programowanie danych sensorów i wyższego wykorzystującego języki wysokiego poziomu w strukturach rozproszonych do tworzenia dedykowanych aplikacji. Zaprezentowane systemy bezprzewodowych sieci sensorów ilustrowane są ich zastosowaniami w logistyce.

Słowa kluczowe: Programowanie w Rozproszonych Strukturach Komputerowych, WEB, Bezprzewodowe Sensory, Logistyka.