

# Unified Sensor Gateway Interconnection of sensor Networks

Roman Trchalík\*, Miroslav Švéda\*\*

\* Faculty of Information Technology, Brno University of Technology,  
CO 612 00 CZ (Tel: 420-541-141315; e-mail: trchalik@fit.vutbr.cz).

\*\* Faculty of Information Technology, Brno University of Technology,  
CO 612 00 CZ (e-mail: sveda@fit.vutbr.cz)

---

**Abstract:** Abstract: In recent years, numbers of electronic devices rapidly grow in our surroundings. The demands and claims are changing with the increasing number of electronic devices. It placed emphasis on easier installation and management this devices. Simultaneously, it increases the requests to optimize power consumption and emphasis on the mobility, unification and advanced control and communication, which enables reciprocal communication among these devices. This paper describes a prototype of the universal gateway connecting various architecture of wireless sensor networks which are based on the architecture of intelligent sensors that use the open IEEE 1451 standard which provides the interface among sensors/actuators and instruments, microprocessors and networks.

*Keywords:* Sensor Networks, Gateways, IP translator, Logical Controller

---

## 1. INTRODUCTION

With the growing number of electronics devices applied in household and in business increases the cost on installation and management such equipment. Moreover, this trend significantly extends the attacks on these devices and if they are not secured and not properly updated, it significantly increases abuse risk, either by stealing their resources such as power computation, disk space or to steal user and corporate data. In the case of sensor networks, an attacker can influence the measured values and thus cause a false security alarm, unlock secure doors or measuring instruments to measure the value of a manufacturing line or such values sugar levels in patients in hospital.

This trend leads to increasing pressure on manufacturers of hardware and software to development quality intelligent receivers which already contain advanced technology such as efficient use of bandwidth or the hardware coprocessor with efficient and native encryption support, which saves electric energy.

This pressure is reflected by advances in development of higher programming languages and frameworks in the software field. The software framework facilitates rapid development individual software products. Moreover the frameworks eliminate repetitive errors during the source code writing, such as treatment of input values in a web form, element <INPUT>, session management and database access. Another indisputable asset of frameworks is that they allow creating more complex systems or cover more existing

systems to one system using the same communication standard.

From the reason of increasing a number of network devices, the companies try to simplify the management of these devices and streamlining the work of the administrators. For the systems with different networking technologies significantly increase demands on their management and their eventual integrations into the information system.

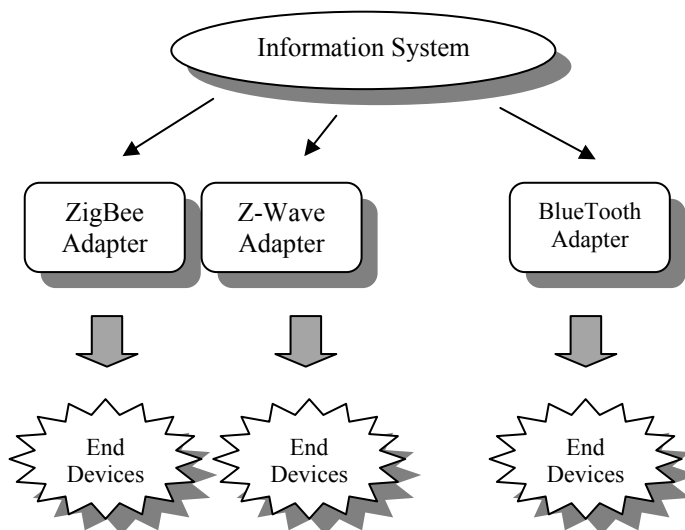


Figure 1: Interconnecting of sensor networks through the Information system

Today there are only two important technologies that allow the creation of advanced network technologies for industrial wireless sensors networks. They are ZigBee and Z-Wave. Wired sensor networks typically use high-speed data bus CAN (Controller Area Network) [13] based on implemented microcontroller and also a separate bus.

## 2. RELATED WORK

Sensor-based applications are the main component of distributed sensor networks that consist of many small spatially dispersed, communicating nodes. The framework, which is base stone of these applications is based on the IEEE 1451 smart transducer interface architecture that enables unify direct coupling to the Ethernet-based Intranets, Intelligent sensors supported by proper networking means can provide features like accuracy, adaptability, reliability, recalibration and also advanced and efficient information processing using data fusion and integration.

The IEEE 1451.1 software architecture provides three models of the transducer device environment, which are (i) the model of a network capable application processor, which is the core of a smart networked device; (ii) the data model, which specifies information encoding rules for transmitting data across local and remote object interfaces; and (iii) the network communication model, which support client/server and publishers/subscribers models for communicating information between application processors.

The majority of communication protocols are based on client-server architecture. If the client wants to call some function on server side, it use command for execution and the requested command is decoded and performed on server side. The perform function evaluates the requested function with the given arguments and after that, it return s the resulting values to the client.

The client-server pattern corresponds to remote procedure call (RPC), which is the remote invocation of operations in a distributed context [lit]. The Client is waiting for the server's response before completion RPC actions related to the current call. It's used synchronous communication.

The publishers/subscribers model of communication provides the efficient distribution of measured data. The clients want to receive data from a smart transducer. The clients has to register themselves to the group of its subscribers using the function subscribe. The transducers in the role of publishers have also the ability to advertise the nature of their future events through an advertise function.

The IEEE 1451.5 proposal specifies a wireless communication protocol and transducer electronic data sheet formats. This proposed standard utilizes the IEEE 802 family as a basis of communication protocols.

To satisfy the standard to new requirements, so in the past 10 years expanded to include support for industry network. The attention was focused on alternative physical layer and expansion of data sheets on properties such as support for XML format, hot swapping possibilities and physical layers information. The consensus was to add the following wireless protocols to standard 1451.5: Wi-Fi with 802.11, Bluetooth with 802.15.1 and ZigBee with 802.15.4. The standardization initiative focusing on the proposal P1451.5 ZigBee is coordinated with Wireless Personal Area Network IEEE 802.15 Task Group 4.

## 3. DESIGN OF UNIVERSAL GATEWAY TO SENSOR NETWORKS

Gateways and Bridges offer two different ways how to provide connectivity. First of them provide a full featured connectivity and allow a greater diversity of devices. However bridges are much simpler and can serve a smaller application space. Gateways convert the wireless protocols and sensor data into various formats which are necessary for commercial, industrial and residential systems. Examples of these formats include BACnet and LonWorks for building systems, SCADA and Modbus industrial networks and HTML and XML language for Internet application (Kinney, 2005).

We would like to achieve, to design the universal application gateway to sensor networks. For this reason, we need to establish a virtual communication layer based on the publisher / subscriber principle.

### 3.1 Requirements for universal gateway:

- Independent of the sensor network architecture – the abstraction of the control module above each network architecture maximizes space in terms of extending the functionality of the system
- Easy installation – simplification of the installation equipment for various sensor networks saves costs and reduces the probability of creating an error during installation
- Modularity - the system must allow adding new device or subsystem without affecting other components (devices) that are already part of the system. At the same time of application part the system must be able to adopt a new device and start with work without any major modifications.
- Scalability - The system should by allow managing hundreds of devices, so that a higher number of devices did not affect the performance of each device.

- Security - the system should be contain basic security authentication mechanisms against abuse of the system users and conventional attacks against network

### 3.2 Layered architecture

Architecture design of the gateway is formed on the basis of a set of requirements listed above. The figure below divides the system into layers based on the functional properties.

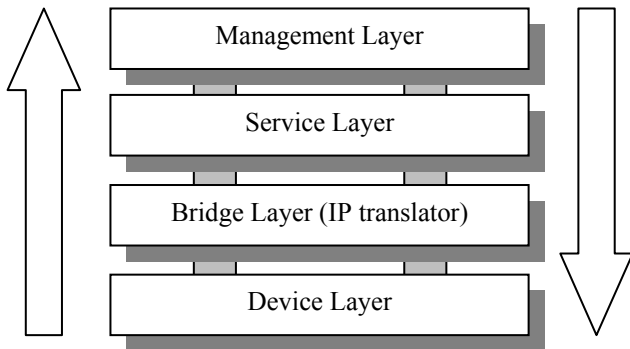


Fig. 2 Layered model

The basic design is dividing the system into 4 layers: (i) Device layer, which contains a set of industrial sensors. These nodes are from some subsystem, such as ZigBee, Bluetooth, Z-Wave, etc. This layer include not only all end nodes of the sensor network, but also network routers, but it can also be main control element of the sensor network like a coordinator, which is a part ZigBee network. (ii) Bridge Layer is responsible for translation between two heterogeneous network protocols. At this layer there is a small application that continuously transforms incoming messages from the one network architecture to another network architecture. In our case, the default network architecture is Ethernet. We have small applications that transform the message into the TCP/IP from each sensor network and other ways. This bridge application is running on network element, which has both physical interfaces: Ethernet and interface to sensor network; Sometime the bridge can be consisting from two components. First component has IP interface and second one has interface to sensor network. Both of them are connected through the serial channel. For example, IP part of bridge can run on standard network router with USB ports and second sensor part can run on ZigBee coordinator; (iii) Service Layer is a area, where is placed the sensor device functionality. In this layer is implemented basic functionality that can create the requirement and invoke its execution on remote smart sensor. Than on the remote site is performed invoked operation. Service layer can accept the answers and trap messages and submit them for evaluation in (iv) Management Layer. This layer is a core of control and logic management functions. There are algorithms which are managing the data changing among end devices which can be placed in different networks with various communication protocols.

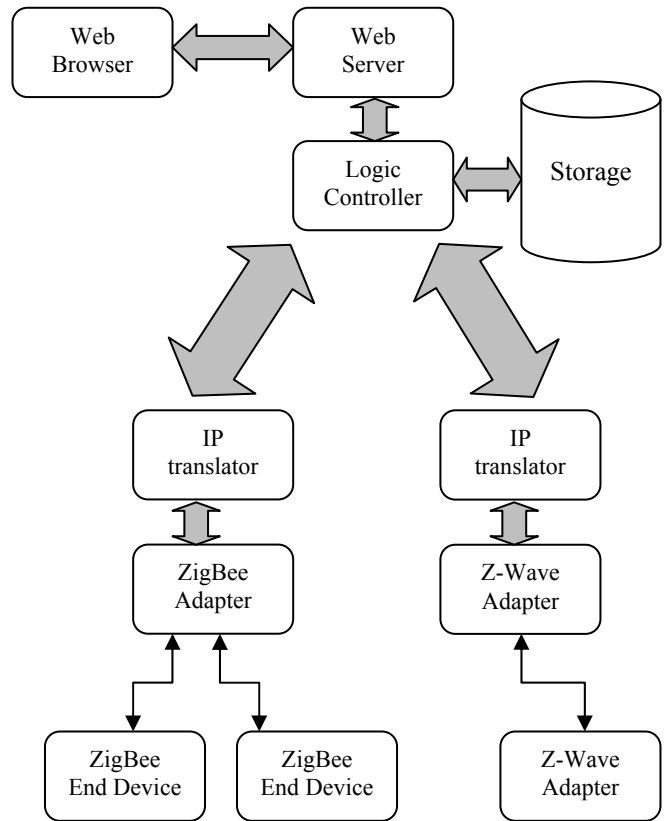


Fig. 3: Components of universal gateway

The figure 3 illustrates a component-oriented flexible gateway designed for multi-domain management system based on smart sensors. Individual components should not be the same network architecture, but several different network architectures and standards such as ZigBee Z-Wave, Bluetooth, etc.

End points are devices, where is placed the smart sensor or more smart sensors – transducer, such as a temperature sensor, pressure sensor, smoke sensor or actuator as a switch, light sensor, etc. This includes – These End points are communicating through the sensor network with their network adapter directly or through the network router or special device like a coordinator.

These endpoints communicate through a sensor network with IP Bridge directly or through the router (coordinator), which is connected to IP Bridge like plug and play device by USB interface or USB/RS232 reduction cable.

The IP Bridge performs encapsulation of frame from sensor networks to TCP or UDP, and sends through the Ethernet network (Internet) on the control logic (controller logic).

### 3.3 Sensor profile

Sensor profile defines the basic characteristics of the sensor and is assigned to each node. These characteristics are used on logical component to identify the sensor type, to determine the list of permitted operations, the sensor calibration data and lots of other statistical properties that depend on the current using of the sensor. Sensor profile is best store in XML format (Figure 4) which can be easily customizable by the user or application requirements. Our profile consists of three parts: (i) AttributeProfile that describe basic characteristics of the sensor as the sensor type, location, manufacturer, accuracy, A / D resolution, etc., (ii) DataProfile that describes the data format generated by sensors; the value can be a whole number (integer) or real number, one value or multiple values – respectively specifies the number of bytes used to represent the properties (sample). (iii) OperationList tells the control unit (logical component).

```
<?xml version="1.0" encoding="UTF-8"?>
<sensor>
  <attributeProfile>
    <attribute name="type">
      Temperature
    </attribute name>
    <attribute name="location">
      Reception
    </attribute name>
    <attribute name="number">
      1
    </attribute name>
  </attributeProfile>
  <dataProfile>
    <dataAttribute>
      <sampleSize>
        4
      </sampleSize>
      <samplingRate>
        4
      </samplingRate>
    </dataAttribute>
  </dataProfile>
</sensor>
```

Figure 4: Example of sensor profile

### 3.4 IP translator

Each IP translator is composed of two shared buffers that are used for communication between client threads and the thread which is providing the communication with networks adapter of sensor network through the serial interface. Each entry in the buffer is assigned an identification number of the client that which put the message into the buffer and the message itself. Administration header contains only the length of the APS framework and for these framework is the message itself. The internal structure of the buffer is shown in Figure 5.

DATA

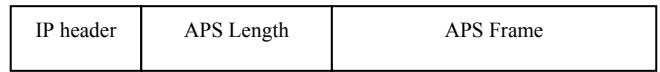


Figure 5: Internal structure of Internet communication

The figure 6 illustrates communication of client thread and thread of serial interface via input and output shared buffer.

When the command *com* is received on Input shared buffer, the client thread inserts command *com* into the input buffer using the method *AddMessage()* (action 1) in the format which is shown on the figure 6. Serial interface thread periodically checks the input buffer using the function *hasNewMessage()* (action 2) and when it receives a positive answer, it removes message from buffer by the method *getMessage()* (action 3) and the message will be processed.

Output shared buffer is used for communication amount the client threads and the thread of serial interface with the direction of communication commands from sensor networks to clients. Serial interface thread receives the APS frame with length of APS frame from the coordinator through the serial interface (Action 4). Message is inserted into the output shared buffer, which is controlled in regular intervals by the client thread (Action 6). There is a possibility to obtain of asynchronous events occurrence that may occur in the sensor network. The length of interval can be set as the shared input buffer. Output shared buffer is monitored with the client thread for the events and of course the case of the thread receives a command from the client *CHCK* (check authentication) (action 5). If the client thread receives a positive response from the method *hasIDMessage()*, whose parameter is the identification number of the client, it can remove the message from the buffer using the method *getMessage()* (action 7), and send the message content to the client.

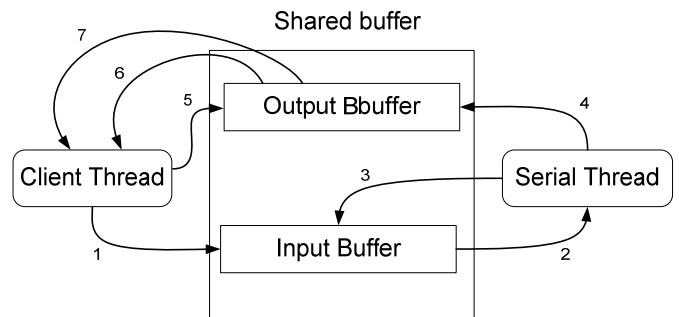


Figure 6: Detailed Shared Buffer of IP translator

### 3.5 Network adapter

Serial thread forwards the message through the serial connection to the network adapter, which handles all communication with the sensor network (ZigBee, Z-Wave and Bluetooth) and data collection from sensor networks. It makes the network adapter a main key member for the management and control of sensor networks. Network allows two-way communication. It is able to receive commands for the sensor network from server, process them and send them to relevant sensors. On the other hand, network adapter can receive information from sensors, evaluate it and send it to the IP translator. Flow chart coordinator is graphically illustrated in Figure 6.

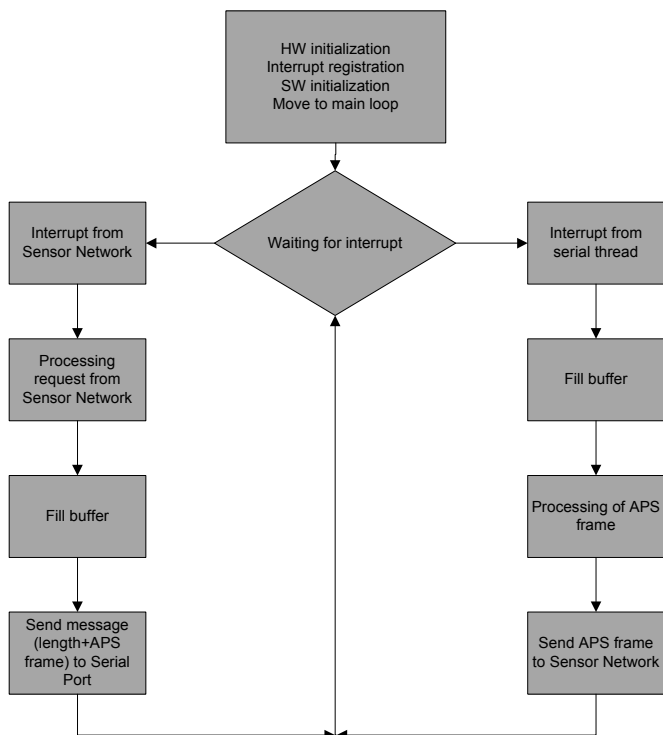


Fig. 6: Logic Controller

This component contains the application logic and service calls. It communicates with network adapter through the IP translator. It performs the following tasks: (i) it receives and processes messages from the network adapter – trap events or responses to commands (check the status light, the current temperature, pressure, etc.); (ii) it analyses the received information and the values entered in XML (figure 7), which are describe the state of the network. It directs the instructions from the user and forwards the network adapter.

```

<!ELEMENT network(PAN, ScanDuration,
BeaconOrd, SuperFrameOrd, BatteryLifeExt,
NumNodes, node+)>
  <!ELEMENT PAN ( #PCDATA ) >
  <!ELEMENT ScanDuration ( #PCDATA ) >

```

```

<!ELEMENT BeaconOrd ( #PCDATA ) >
<!ELEMENT SuperFrameOrd ( #PCDATA ) >
<!ELEMENT BatteryLifeExt ( #PCDATA ) >
<!ELEMENT NumNodes ( #PCDATA ) >
<!ELEMENT node ( ieee, mac, logtype, eplist ) >
  <!ELEMENT ieee ( #PCDATA ) >
  <!ELEMENT mac ( #PCDATA ) >
  <!ELEMENT logtype ( #PCDATA ) >
  <!ELEMENT eplist ( #PCDATA ) >
  <!ELEMENT eplist ( EP+ ) >
  <!ELEMENT EP ( id, name, value,
includcount, includclusters, outcludcount,
outcludclusters ) >
    <!ELEMENT id ( #PCDATA ) >
    <!ELEMENT name ( #PCDATA ) >
    <!ELEMENT value ( #PCDATA ) >
    <!ELEMENT includcount ( #PCDATA ) >
    <!ELEMENT includclusters ( includcluster? ) >
    <!ELEMENT includcluster ( #PCDATA ) >
    <!ELEMENT outcludcount ( #PCDATA ) >
    <!ELEMENT outcludclusters ( outcludcluster? ) >
    <!ELEMENT outcludcluster ( #PCDATA ) >

```

Figure 7: DTD for XML file:

### 4. EXAMPLE

We performed an experiment with ZigBee sensor network. We used two sets of development kits called Picdem Z developed by Microchip Company. These devices are able to perform the work as RFD (Reduced Function Device) devices or FDD (Full Function Device) devices like a coordinators or network routers. We used Zena analyzer for frames sniffing and analyzing of wireless communications. The coordinator of the first ZigBee sensor network is connected via a USB-serial reduction to the first standard network router (Access point) Asus RT-N16, which has two USB interfaces. The same connection we have made on the second router Asus W500g, which has also two USB interfaces. On both routers are installed embedded Linux OS, they are connected to the Internet.

Logic control unit is created by using simple bash and python scripts, which are stored on the first router Asus RT-N16. We don't have implemented and installed the web interface for this demonstration because it is not necessary for verification of the functionality designed unified sensor gateway. Individual tasks (commands) were started from the Linux console. IP translator is running as system demon and control application is connected to the IP translator via loopback interface. IP translator communicates with the network adapter through the serial interface with ZigBee coordinator, which is connected via USB-RS232 reduction directly to router.

The experiment demonstrate that when the button is pressed on the one node in the first sensor network, the LED start lighting on another node from the second wireless sensor network. When the button is pressed on end node (i) it is sent a *toggleMessage()* to ZigBee coordinator of current network,

which is forwarding the message to (ii) IP translator through serial channel. IP translator forwards the message to all IP clients. This message is sent by unicast using the TCP protocol. (iii) message is received by control the controller, which decodes the message and creates a reaction and sends the message by the opposite way to the second sensor networks.

The component of network adapter was created by software modification of ZigBee coordinator. The coordinator is able to forward selected fields from ZigBee frame to IP translator through serial channel. Each sensor network has own IP translator and by its IP address and its port number we are able to identify the type of network and on the basis of the right to choose a template and use the right parsing procedure on the received frame. By the reverse method we are able to create query to any node in the network. For another types of sensor network we can use another set templates and another network adapter.

## 5. CONCLUSION AND FUTURE WORK

This article describes a multi-layer architecture design of the universal gateways to sensor networks with different communication protocols. This type of the gateway would give businesses a considerable choice of sensor probes in building their sensor networks. Moreover this solution allows them to combine different network technologies and to adapt network architecture to local conditions.

Our proposed system allows unifying the control and management into the one central location that lead to reducing the costs of managing and upgrading the network. Further advantage is that it permits the connection of monitoring and logging device directly to the IP translator or it allows the connection of a backup control system which would monitor the network status and in the event of failure of primary control system or updating of primary control system it could take the control of the connected sensor systems.

## 6. ACKNOWLEDGEMENT

The research has been supported by the Czech Ministry of Education in frame of the Research Intention MSM 0021630528: Security-Oriented Research in Information Technology, the IT4Innovations Centre of Excellence MŠMT ED1.1.00/02.0070 and by the projects FIT-S-11-1 and TAČR TA01010632.

## REFERENCES

Baronti P., Pillai P., Chook V.Q.C., Chessa S., Gotta A., Hu Y.F. (2007). Wirellwss sensor networks: A survey on the state of the art and the 802.15.4 and ZigBee standards. In Computer Communication,

Ilyas M., Mahgoub I. (2004). Handbook of Sensor Networks: Compact Wireless and Wired sensing Systems, CRC Press LLC. Boca Raton, FI, USA

IEEE 1451.1, Standard for a Smart Transducer Interface for Sensors and Actuators -- Network Capable Application Processor (NCAP) Information Model, IEEE, New York, USA, 2000.

Jurdak R., Lopes C. V., Baldi P (2004). A framework for modeling sensor networks. Workshop on Building Software for Pervasive Computing, USA

Kinney, P.: Gateways: Beyond the sensors networks. Zigbee Alliance, Website <http://www.zigbee.org/>

Microchip. An1232 zigbee 2006 protocol stack. [Online]. Available: <http://www.microchip.com>

Microchip, Picdem Z. [Online]. Available <http://www.microchip.com/zigbee>.

Sohraby K., Minoli D., Znati T. (2007). Wireless Sensor Networks: Technology, Protocols, and Applications. Wiley-Interscience. USA

ZigBee Alliance, ZigBee specification (2008). [online]. Available: <http://www.zigbee.org>

ZigBee Alliance, Zigbee Cluster Library Specification. (2008). [online] Available: <http://www.zigbee.org>