RZ 3634 (# 99644) 10/17/2005
Electrical Engineering 5 pages

Research Report

The IBM Wireless Sensor Networking Testbed

Simeon Furrer, Wolfgang Schott, Hong Linh Truong, and Beat Weiss

IBM Research GmbH
Zurich Research Laboratory
8803 Rüschlikon
Switzerland
{sfu,sct,hlt,wei}@zurich.ibm.com

The IBM Wireless Sensor Networking Testbed

Simeon Furrer, Wolfgang Schott, Hong Linh Truong, and Beat Weiss
IBM Zurich Research Laboratory, CH-8803 Rüschlikon, Switzerland
{sfu,sct,hlt,wei}@zurich.ibm.com

Abstract

This paper describes the wireless sensor networking testbed built at the IBM Zurich Research Laboratory. The testbed has been used to address a wealth of exciting research challenges. Performance evaluations have been carried out with short-range wireless communication technologies, which are highly relevant for sensor networking such as IEEE 802.15.4/ZigBee networks, Bluetooth WPANs, and IEEE 802.11b WLANs. With the testbed, the merits of wireless mesh networking for range extension and reliability enhancement have been explored. New light-weight messaging protocols for communication between sensors and an application server have been tested which allow to bring messaging-oriented middleware down to very low-end sensors and actuators. In addition, the testbed has been used to develop sensor applications for remote metering and location-sensing.

1. Introduction

According to various market forecasts and research reports [1,2], sensors and actuators will be massively deployed soon and outnumber traditional electronic appliances such as personal computers and mobile phones in a few years. This trend is already visible in various industry segments such as the automotive industry, retail, as well as in the home environment (Fig. 1), where sensors are used to monitor and control equipment, for automatically collecting data, for providing basic security functions, and for location sensing. Today most of the sensors are connected with wires to a local controller which monitors or processes the sensed information. However, we already see the emergence of advanced wireless sensors, which can forward the sensor data to the controller via a wireless local-area network (LAN) (IEEE 802.11b, etc.), a wireless personal-area network (PAN) (Bluetooth, IEEE 802.15.4, etc.), or a ZigBee [3] compatible mesh

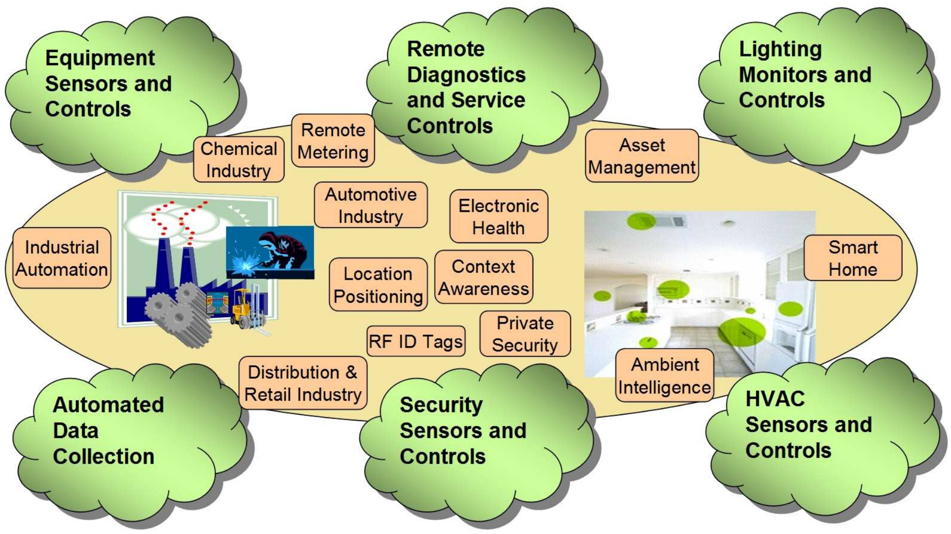

Figure 1. Sensor Network Scenarios and Applications [1]

network. Each of these networks can be viewed as a type-specific wireless sensor network.

The growing number of sensors and actuators will also enable a plethora of novel attractive services and applications that can be employed in industrial automation, asset management, environmental monitoring, transportation business, and for remote diagnostic and service control. Many of these future applications require forwarding of the sensor information from the sensor network via an enterprise network to a remote application server. These sensor applications will increase the data traffic in the network significantly, and introduce a directional shift in the network traffic because most of the traffic in sensor networks will flow from the sensors to the application server and no longer from the network to the user as in traditional networks. To cope with the growth and directional shift in data traffic, significant architectural changes have to be incorporated into the network to efficiently support the sensor data traffic. The changes encompass, firstly, the placement of a computing device ("edge server") between the sensor and enterprise network to aggregate, abstract, and filter the sensor data before they enter the backbone network, and, secondly, the use of an asynchronous publish/subscribe-based messaging system to reduce the degree of coupling between the sensor network components, edge server, and application server [4,5]. Moreover, sensor networks may be deployed in remote locations, which require remote system management for control and configuration as well as support for software updates.

In this paper, we describe a wireless sensor networking testbed that has been built at the IBM Zurich Research Laboratory. The testbed prototypes a complete end-to-end solution, reaching from the hardware of the sensors and actuators to the application software executed on an enterprise server. It comprises various types of wireless sensor networks, a sensor gateway and middleware connecting the sensor world to the enterprise computing environment, and the sensor application software. The testbed has been built to evaluate the performance of short-range wireless communication technologies, which are highly relevant for sensor networking such as IEEE 802.15.4/ZigBee networks, Bluetooth WPANs, and IEEE 802.11b WLANs, to test new light-weight messaging protocols for asynchronous communication between sensors and the application server, and to develop applications such as remote metering and location sensing. A photograph of the key components of the wireless sensor networking testbed is shown in Fig. 2.

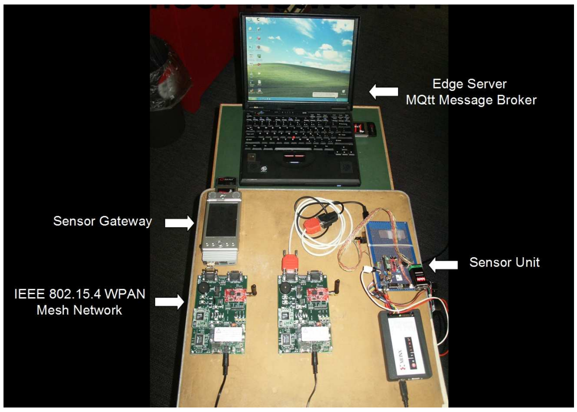

Figure 2. Wireless Sensor Networking Testbed

2. Wireless Sensor Networking Testbed

Fig. 3 gives an overview on the wireless sensor networking testbed. It integrates a wireless sensor network with an enterprise network in order to bring data generated by low-end wireless sensors to an application server.

The testbed consists of the following components*:*

- A sensor unit equipped with several types of sensors;
- A wireless sensor network;
- A sensor gateway connecting the wireless sensor world with the enterprise computing environment;
- Middleware components for distributing the sensor data to sensor applications; and
- Sensor application software.

2.1. Sensor Unit

The sensor unit contains several types of sensors for reliably measuring its own orientation and movement in a given co-ordinate system, namely a compass, accelerometers, gyroscopes, and a thermometer. The compass measures the orientation of the sensor unit in an earth-fixed co-ordinate system, while the accelerometers and gyroscopes sense the acceleration and angular rate of the moving unit in a Cartesian co-ordinate system that is body-fixed to the sensor unit. Offsets in the sensed data caused by temperature drifts can be compensated by taking into account the temperature measurements. The sensor unit also contains an A/D converter for converting analog sensor signals into a digital format, a field programmable gate

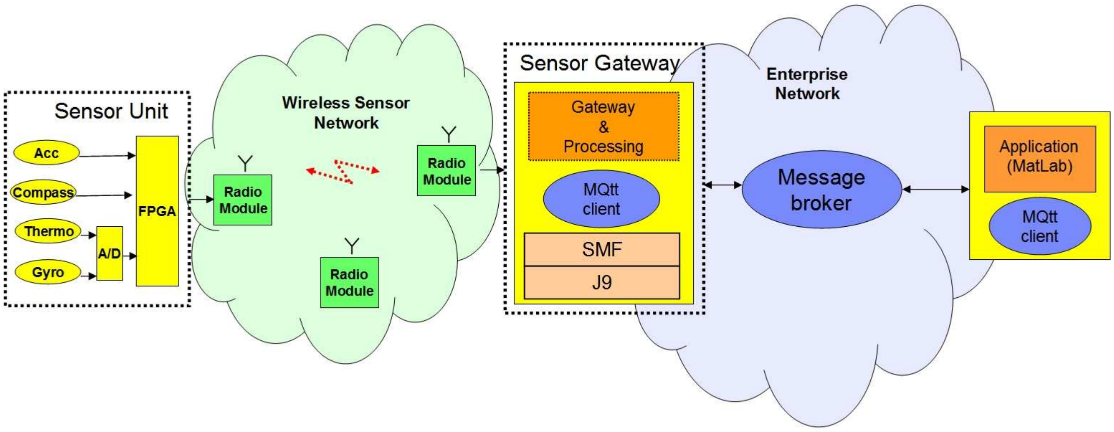

Figure 3. Wireless Sensor Networking Testbed

array (FPGA), and a RS232 serial port interface. The main functions of the FPGA are:

- collect data generated by the sensors;
- assemble data into a data frame; and
- send the resulting data frames to the serial interface.

The serial interface connects the sensor unit to a radio module which converts the sensor data frames to radio messages in order to forward them to the enterprise network.

2.2. Wireless Sensor Network

In our testbed, a wireless sensor network (WSN) is used to transfer the sensor data frames from the sensor unit over a radio interface to the sensor gateway. For this purpose, the sensor unit and gateway are attached via a serial interface to radio modules. If a radio link can be established between these modules for peer-to-peer communication, the radio modules put each sensor data frame into a radio message, send the message over the radio link, and extract the sensor data frame from the received radio message. If the two radio modules cannot establish a direct radio link, additional radio modules are randomly placed in the area between the sensor unit and gateway in order to form a so-called wireless mesh network. This network organizes itself and is self-healing, i.e. network nodes automatically establish and maintain connectivity among themselves. Each module thus operates not only as a data source or sink, but also as message forwarder for other nodes that do not have direct connectivity with their communication peers. Since today's mesh networking protocols were not designed for wireless sensor networks, we are investigating methods to achieve scalability to a large number of nodes and to take into account the limitations imposed by low cost, low processing, and low power characteristics of wireless sensor nodes.

In our wireless sensor networking testbed, we tested and evaluated the performance and scalability of various promising mesh networking protocols, notably EmberNet [6], ZigBee, and tinyDB [7]. Moreover, we have successfully employed the following radio modules:

- Bluetooth IEEE 802.15.1 and WLAN IEEE 802.11.b modules;
- Ember development kit with IEEE 802.15.4 radio and Zigbee networking stack; and
- Crossbow "MICAz" Motes with IEEE 802.15.4 radio [8].

2.3. Sensor Gateway

The sensor gateway is implemented on an embedded Linux application platform, which is suitable for the HW/SW development of embedded processor applications. The core hardware component of the platform is a PowerPC 405 processor, which has been designed for very low power consumption. It is thus well-suited for sensor applications. The processor can access via a high-speed bus many peripheral devices, among them built-in A/D and D/A converters, interfaces to the Ethernet, an IEEE 802.11b WLAN PCMCIA card, and a Bluetooth SDIO card. Fig. 4 shows the software environment that is executed on the embedded processor. MontaVista embedded Linux CEE 3.0 is used as operating system, which fully exploits the power-save features of the processor. On the operating system, the Webshpere Micro Environment [9] is installed, which includes a J9 Java virtual machine representing the programming platform for additional middleware components and application software.

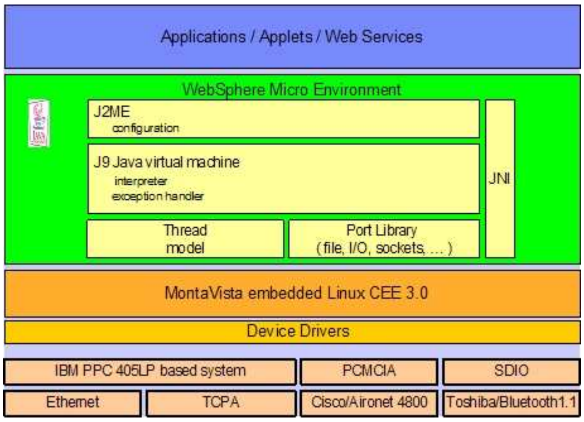

Figure 4. Software Environment

The sensor gateway provides the bridge between the wireless sensor world and the enterprise computing environment. On the wireless sensor side, it is attached via a RS232 interface to a radio module; on the enterprise side, it is connected to the enterprise network via an integrated Ethernet port or the WLAN PCMCIA card. It hosts a java “gateway” application which extracts sensor data frames from the information received from the wireless sensor network and uses a telemetry transport protocol to distribute the sensor data to one or several applications executed in the enterprise network.

2.4. Middleware

In our wireless sensor networking testbed, the sensor gateway uses an advanced version of the MQ telemetry transport protocol (MQtt) [10] to communicate with the sensor applications in the enterprise network. It is an open, light-weight message transfer protocol, specifically designed for remote telemetry applications and optimized for communications over low-bandwidth networks.

The MQtt protocol is based on a publish/subscribe messaging model for distributing information between networked applications by using a message broker. Publishers send (publish) messages to the broker on a specific "topic". Subscribers register (subscribe) their interest in certain topics at the broker. The broker manages the connections to the publishers and subscribers, and distributes the messages it receives from the publishers to the subscribers according to their subscribed topics. The use of a message broker decouples the communication between publishers and subscribers. This decoupling makes the system easily reconfigurable, because the binding between information producers and information consumers is performed via topics, allowing new applications with new combinations of topics to be added and deployed at any time.

The main function of the message broker and the MQtt protocol is to connect the wireless sensor network with the backend computing environment of the enterprise. The data collected by the sensors are thus made available to all applications in the same way as any other enterprise information. The broker also performs certain message processing tasks, such as transforming raw sensor data into an XML-formatted message, which is more suitable for many enterprise applications.

In addition, the middleware provides a platform for the deployment and maintenance of services and applications of networked devices. In our testbed, we use the Service Management Framework (SMF) [11] to remotely start, stop, and manage message filtering applications and software packages running on the sensor gateway without interrupting the operation of the device. SMF is IBM’s implementation of the Open Services Gateway initiative (OSGi) architecture specified by the OSGi Alliance [12]. The OSGi architecture is based on Java to ensure its independence from operating systems and processor architecture. It allows applications to share a single Java Virtual Machine (JVM), and can install, start, stop, update and uninstall applications on the fly without affecting the operation of other applications.

2.5. Sensor Application

The wireless sensor networking testbed has been built to develop sensor applications such as remote metering and location sensing of a moving device. For this purpose, we built a sensor unit that can be viewed as a miniature inertial measurement system, providing multi-dimensional acceleration, angular rate, and earth magnetic-field data. By attaching the sensor unit to a slowly moving vehicle or mobile robot, the orientation, motion, and location of the device can be tracked by processing the sensed inertial measurement data with some proper data-fusion and location-tracking algorithms [13]. This processing can be done either in a computing device attached to the sensor unit, or on a remote application server in the enterprise network.

In our testbed, the sensor application software is executed on a server. As described in the preceding sections, the application receives the inertial measurement data from the broker for further application-specific processing after having subscribed to these data published by the sensor gateway. To graphically visualize at the server the remotely measured sensor data, we wrote a Matlab-based application program that uses the MQtt protocol to communicate with the broker and get access to the data. Fig. 5 illustrates the output of the program. The diagram on the left indicates the orientation of the sensor unit in the earth-fixed reference co-ordinate system. The right diagram indicates the acceleration vector of the sensor unit measured in the x-y plane of the co-ordinate system that is body-fixed to the sensor board. The value of this vector equals the gravity acceleration g if the measured value lies on the circle. In addition, the temperature and the angular rate of the board measured with respect to the z-axis of the sensor board co-ordinate system are given in numerical representation.

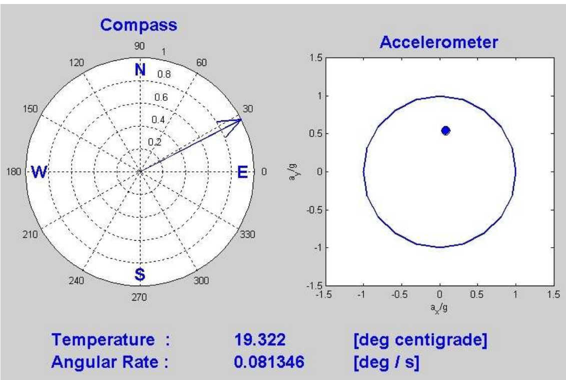

Figure 5. Visualization of Sensor Data

3. Summary

In this paper, we have described the IBM wireless sensor networking testbed, which has been built to evaluate the performance of new wireless sensor technologies, advanced middleware concepts, and novel sensor applications. With the testbed, we have assessed the suitability of various short-range wireless communication technologies for sensor networks, explored the merits of wireless mesh networking, evaluated the efficiency of light-weight message-oriented transport protocols, and developed sensor applications for remote metering and location sensing. Since the testbed encompasses all HW/SW components reaching from low-end sensors to the application executed on an enterprise server, the IBM wireless sensor testbed represents a reference design for end-to-end solutions.

ACKNOWLEDGEMENT

The authors gratefully acknowledge the managerial support given by P. Chevillat and useful discussions on middleware-related issues with D. Bauer and J. Rooney.

REFERENCES

[1] P. Chevillat, P. Coronel, W. Schott, et.al., “WWRF Briefings 2004: Wireless Body Area and Sensor Networks,” Wireless World Research Forum, Dec. 2004.
[2] R. Clauberg, “RFID and Sensor Networks”, RFID workshop, University of St. Gallen, Sept. 2004.
[3] ZigBee Alliance, http://www.zigbee.org
[4] S. Rooney, D. Bauer, and P. Scotton, “Edge Server Software Architecture for Sensor Applications”, IEEE Intern. Symposium on Applications and the Internet, Jan. 2005.
[5] S. Rooney, D. Bauer, and P. Scotton, “Distributed Messaging using Meta Channels and Message Bins”, 9th IFIP/IEEE Intern. Symposium on Integrated Network Management”, May 2005.
[6] Ember Wireless Semiconductor Solutions, http://www.ember.com/
[7] TinyDB, A Declarative Database for Sensor Networks, http://telegraph.cs.berkeley.edu/tinydb/
[8] Crossbow Technology Inc., http://www.xbow.com/
[9] IBM WebSphere Everyplace Micro Environment, http://www-306.ibm.com/software/wireless/weme/ features.html
[10] IBM WebSphere Telemetry Integration, http://www-306.ibm.com/software/integration/mqfamily/ integrator/telemetry/
[11] IBM Service Management Framework, http://www-306.ibm.com/software/wireless/smf/
[12] OSGi Alliance, http://www.osgi.org/
[13] N. Dunais, “Location Tracking of Mobile Devices with Inertial Measurements Units,” IBM Research Report, September 2005.