

# Chapter 22

## From the Lab to the Factory Floor: Engineering Software for Wireless Sensor Networks

**Christian Scholz**

*University of Applied Sciences Coburg, Germany*

**Thomas Wieland**

*University of Applied Sciences Coburg, Germany*

**Christoph Niedermeier**

*Siemens Corporate Research, Germany*

### ABSTRACT

*Wireless sensor networks consisting of numerous embedded systems with wireless connectivity, called motes, have gained much interest in network-related research involving MAC protocols, routing, or data aggregation. The various projects that have been undertaken so far are mostly in a scientific context; the software development has not taken into consideration appropriate software engineering methods and processes. Practical adoption of this technology in industry, agriculture, building automation, or similar fields of use, requires well-defined software engineering processes. Building a sensor network application is either a systems engineering task if the mote hardware has to be developed as well, or a pure software engineering task if existing hardware is going to be used. Both cases are considered in this chapter. Each phase of the software life cycle is analyzed, and recommendations concerning the proper (software) engineering of wireless sensor networks are made.*

### 1. INTRODUCTION

Wireless sensor networks (WSNs) have gained much attention in recent years due to their ability to support data gathering and monitor a wide variety of environments. WSNs consist of embedded systems called *motes* (Hill, Szewczyk, Woo, Hollar,

Culler, & Pister, 2000), each equipped with a logic processor, some memory, one or more sensors, interfaces to analog or digital external sensors and actuators, and a wireless transmission unit. The widely distributed computing capabilities within such a network together with ad hoc and mesh networking connectivity make this programming environment unique and promising.

DOI: 10.4018/978-1-61520-655-1.ch022

Currently, research activities have mainly focused on technical aspects like energy efficiency, message routing, MAC protocols, or data aggregation. But for using the WSN technology as a product or at least a reliable individual system in industry, an out-of-the-box readiness has to be achieved. The practical and wide-spread adoption of sensor networks seems to require powerful and robust software services running on the nodes. This software should offer a much more abstract programming model such that the industrial developer can concentrate on his own application, without needing to know about all the hardware and protocol details of the mote.

As a consequence, the development of an industrial WSN application for running on factory floors is considerably different from the development of lab applications. In scientific sensor networks protocols and applications are usually built by the same team, following a common and thus closely coupled design idea. An industrial development requires a much more stringent software engineering approach.

In industrial contexts some properties of sensor networks are less important than in other environments. One of this is mobility: As production equipment is generally stationary, the sensors are not supposed to move around in a large radius. They may be mounted on a moving part of a machine, but these normally have just a range of a few meters. In some cases, however, like item tracking between production lines, mobility can be an issue. Since we focus primarily on industrial applications, we will see mobility as a cross-cutting concern, but will not treat mobility aspects like localization or frequent topology rearrangements in detail.

The experiences from business software engineering may be transferred to WSNs to some extent. The use cases and overall conditions in sensor networks are so specific, that many aspects have to be considered individually. In this chapter, we want to highlight these aspects by identifying the respective problems and outlining possible

solutions. Although we are going to structure this chapter in line with the usual software lifecycle, this does not mean that pure waterfall-like processes are recommended. Applications for wireless sensor networks – even in industry – can be and will be developed in agile processes in an iterative manner, too.

## **2. REQUIREMENTS AND PLATFORMS**

Development of applications for WSNs is strongly driven by the requirements imposed by the application domain as well as the specific application context. Industrial applications of WSNs exist in quite a number of different domains such as: Factory automation, process automation, predictive maintenance (condition monitoring of machines and facilities), building automation (HVAC control systems, lighting control systems, automatic meter reading), building security (intrusion detection systems, fire alarm systems), and logistics (inventory management, monitoring of goods and means of transportation).

Functional and non-functional requirements have to be investigated separately for each of these domains as there are considerable differences between them. For instance, some applications in the field of factory automation are subject to hard real-time constraints with strict deadlines whereas fire alarm systems and intrusion detection systems have less severe real-time constraints but must be highly reliable and well protected against external attacks that may affect their integrity. As another example: Building automation systems must permit long maintenance intervals in order to be cost-efficient. Thus energy-efficient design of wireless components resulting in a low frequency of battery changes is vital for this domain.

General purpose solutions taking entirely into account this large variety of requirements are difficult to achieve. Therefore, hardware and software components usually have to be specifi-

17 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

[www.igi-global.com/chapter/lab-factory-floor/66478](http://www.igi-global.com/chapter/lab-factory-floor/66478)

## Related Content

---

### A Tagging Approach to Extract Security Requirements in Non-Traditional Software Development Processes

Annette Tetmeyer, Daniel Heinand Hossein Saiedian (2014). *International Journal of Secure Software Engineering* (pp. 31-47).

[www.irma-international.org/article/a-tagging-approach-to-extract-security-requirements-in-non-traditional-software-development-processes/121681](http://www.irma-international.org/article/a-tagging-approach-to-extract-security-requirements-in-non-traditional-software-development-processes/121681)

### Open Source Software Evaluation

Karin van den Berg (2009). *Software Applications: Concepts, Methodologies, Tools, and Applications* (pp. 52-65).

[www.irma-international.org/chapter/open-source-software-evaluation/29378](http://www.irma-international.org/chapter/open-source-software-evaluation/29378)

### A New Coding Scheme for Data Security in RF based Wireless Communication

Irfan Habib, Atiqul Islam, Suman Chetiaand Samar Jyoti Saikia (2015). *Intelligent Applications for Heterogeneous System Modeling and Design* (pp. 301-319).

[www.irma-international.org/chapter/a-new-coding-scheme-for-data-security-in-rf-based-wireless-communication/135892](http://www.irma-international.org/chapter/a-new-coding-scheme-for-data-security-in-rf-based-wireless-communication/135892)

### Automata-Based Approach to Manage Self-Adaptive Component-Based Architectures

Abdelfetah Saadi, Youcef Hammaland Mourad Chabane Oussalah (2022). *International Journal of Software Innovation* (pp. 1-22).

[www.irma-international.org/article/automata-based-approach-to-manage-self-adaptive-component-based-architectures/297623](http://www.irma-international.org/article/automata-based-approach-to-manage-self-adaptive-component-based-architectures/297623)

### Software Configuration Management in Agile Development

Lars Bendixand Torbjörn Ekman (2009). *Software Applications: Concepts, Methodologies, Tools, and Applications* (pp. 291-308).

[www.irma-international.org/chapter/software-configuration-management-agile-development/29394](http://www.irma-international.org/chapter/software-configuration-management-agile-development/29394)