GENSEN: A Novel Combination of Product, Application and Technology Platform Development in the Context of Wireless Automation

Reino Virrankoski
University of Vaasa, Department of Computer Science,
Telecommunication Engineering Group
P.O. Box 700, FI-65101 Vaasa, Finland
Tel. +358-6-324-8694, Fax. +358-6-324-8467
reino.virrankoski@uwasa.fi

Simo Keskinen
University of Vaasa, Department of Production,
P.O. Box 700, FI-65101 Vaasa, Finland
Tel. +358-6-324-8482, Fax. +358-6-324-8467
simo.keskinen@uwasa.fi

Keywords: multi use platform, wireless automation, wireless sensor networks

Abstract

During the last decade, several wireless sensor platforms capable to use commercial sensors have been developed for wireless monitoring and automation. However, the field of wireless automation applications is still immature, and there are no complete generic solutions widely available. Many applied wireless solutions have been developed just for cable replacement in simple star topology setups. They measure and transmit data without any processing or any system adaptivity. These kind of “dummy” wireless sensor networks usually run into troubles with limited transmission capacity and limited energy resources if they are applied to wireless automation, where the application might also pose such requirements for the sensor networking platform that time-consuming application-specific tailoring and configurations would be needed. Many of the developed network architectures have been small with just a couple of nodes manually placed in a single-hop star topology. They are unfeasible for big automation systems, because they neither scale up with the number of nodes nor adapt to the changes in the network topology. Compatibility with the rest of the automation system is also forgotten such that many of the wireless automation application prototypes have became stand alone systems instead of integrating to the rest of the automation system. Usually the most expensive and time-consuming phase in the product development is the process to develop a platform up to such a level that allows a fast production of different applications on top of the platform. In this paper we describe a product-making process in the context of research and development project Generic Sensor Systems for Wireless Automation (GENSEN). In this project a combination of product, application and technology platform development is applied to create an application platform that enables a fast production of different kinds of wireless automation applications. Our solution will fill the existing gaps between the current wireless sensor networking platforms and the various needs of different applications of wireless automation. The developed platform enables automatic network configuration and data processing in the network. It supports complex, multi-hop network structures and scales up to hundreds of sensor nodes.
Introduction

Wireless sensor systems enabled by wireless communication provide new opportunities for industrial automation. Since wireless sensor systems are remarkably cheaper than the cabled ones, a bigger number of measurement points can be applied. This allows us to collect more precise and more redundant data from the system, which enables advanced control system development. Moreover, wireless connection provides us access to such places that are difficult or impossible to cable, such as rotating machine parts, several measurement points in one motor, high-voltage locations in an electric power transmission system and locations in the middle of greenhouse flora.

There are several challenges to be solved to make the wireless sensor systems fill the automation requirements. The data transmission rate must be high enough to fill the real time operation requirement. The energy supply to the wireless sensor nodes must be organized in such a way that it does not require continuous human intervention. Since the data transmitted over the wireless channel is typically incomplete and can also include misleading information, automatic fault diagnostics must be applied to be able to eliminate measurement outliers and to complete missing information. In time-critical applications one must also be able to handle the time-varying transmission delays (Eriksson 2008).

If we have such an application that includes tens or even hundreds of measurement points, like a sensor node deployment in a modern greenhouse (Ahonen 2008), automatic network initialization is required. Automatic self-healing from the possible network malfunctioning situations is also needed to make the system reliable. Tens or hundreds of measurement points can produce a huge amount of data. Since energy consumption and data transmission rate are both remarkable bottlenecks in the practical feasibility of sensor networks, and since each sensor node is equipped with microprocessor and some memory, some data processing should be performed locally. Instead of using the network just to collect the data, one should compress the data in the network such that only useful part of the information will be transmitted all the way from the local area network to the centralized network control. It is also possible to perform event-based networking such that the nodes are in the low-power operating mode most of the time and switch on their radios just if they have something important to transmit. In addition to single sensor data compression, one must also perform data fusion in such systems, which use information measured by several different types of sensors. This kind of data fusion is typically performed either in a centralized knowledge after collecting all the data from the network or in a locally centralized manner in network cluster heads if a hierarchical network architecture is applied.

Typically a wireless sensor network, which operates under IEEE 802.15.4 communication protocol, forms just a part of the communication architecture applied in automation system. Some other parts can be based on IEEE 802.11 wireless communication, or they can be cabled. Thus, one design challenge is to make the wireless sensor network compatible with the rest of the system. It is, for example, known that IEEE 802.11-based WLAN network can be an efficient jammer to IEEE 802.15.4-based network. Thus, in the research point of view, we must pay attention to interference minimization and radio resource management. Even without an existence of other wireless communication network, these issues become important in harsh environments such as electric grid substations or spaces that include a lot of metal.

There are a huge amount of industrial sensors as well as several sensor networking platforms consisting of wireless sensor nodes (sometimes called sensor motes) commercially available (Crossbow 2009, Polastre 2005, Lymberopoulos 2005, Imote2 2009, Sensinode 2009). However,
one is still missing such a generic wireless sensor network concept, which enables a rapid production of different kinds of automation applications. Quite often the applications need to be built from the scratch and tailoring the nodes, protocols and other software takes a lot of time and effort. We are targeting to fill this lack in GENSEN-project, and produce a generic sensor network concept for wireless automation. We will apply such sensor platforms produced by Telos (Polastre 2005) and Sensinode (Sensinode 2009), which support Contiki operating system (Contiki 2009). We will also consider Sensinode Microseries and Nanoseries platforms up to such versions of Nanoseries, which still offer an open source code. To guarantee the compatibility with the other parts of the automation system, our solution supports IP-based networking with IPv4 and IPv6. This is accomplished by using a 6LoWPAN compliant protocol stack (either NanoStack or Contiki μIPv6) (6LoWPAN 2009). In some applications we may also develop our own hardware which is fully compatible with the pre-mentioned ones. A sensor development is out of the focus of this project because we are applying commercially available sensors.

The core part of the technology platform development is the creation of novel algorithms for networking and data processing and their implementation to the protocol stack. Related to the product platform, special attention will be paid to system reliability and overall system usability including user interfaces. Also product covering to create different system parts an appearance suitable for marketing will be considered. Application platform is developed through five different applications: trolley crane automation, wind turbine generator monitoring and control, cattlehouse automation, greenhouse automation and distributed energy production system monitoring and control. In the case of each application, customer needs, application specific technical requirements, application environment and system integration to other subsystems will be evaluated by using various technical and customer satisfaction criteria. Quality Function Deployment (QFD) can be used as a one tool in this process.

After going over the basics of the platform development and related work, we describe the way how product development is done by using a combination of product, technology and application platforms in the context of GENSEN-project. The project is funded by Finnish Agency for Technology and Innovation (Tekes) and by the participating companies. Participating research units come from the University of Vaasa, Helsinki University of Technology and Seinäjoki University of Applied Sciences. Finally we give a discussion about the benefits achieved through combined platform development process and point some directions to future work.

Related Work

Platforms

Typically automation system consists of operatively independent units called modules. Hardware, software and mechanical architecture must all be taken into account in the modular structure. Also production structures are based on modular (or platform) architecture in which the modules can also be called platforms. In modular production architecture one can freely combine different types of platforms. This property can be exploited in several production sectors where the physical compatibility is the only requirement.

The situation is much more challenging in intelligent embedded systems, such as wireless sensor networks. The associated challenges can also be seen as an opportunity, if one can systematically foresee the current and close future requirements and direct his product development accordingly. Several requirements must be filled to make the modules compatible
with certain generation products and to enable transmission from current to next generation. These requirements can be fulfilled by applying platform structuring and platform-based planning.

Previously a term platform was used in the context of technology to call physical construction elements such as bridges, skeletons, metal plates etc. On 1990s the term became common also in electronics. At the beginning it was used to characterize software architectures and later also other application areas of electronics. Software architecture platform is illustrated in Figure 1.

<table>
<thead>
<tr>
<th>Application layer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application-specific platform</td>
</tr>
<tr>
<td>Common application platform</td>
</tr>
<tr>
<td>Hardware, operating system, database management system, programming environment, communication environment etc.</td>
</tr>
</tbody>
</table>

*Figure 1: Software architecture platform (Jakobson 1993)*

There exists several platform definitions and characterizations (Maier & Lehnerd 1997, McGarth 1995, Sanchez & Machoney 1996). A product platform can be defined to be the common part in certain product group, family or generation. Different kinds of modifications as well as practices to improve process and information management can be built on top of the product platform. It can also be seen as a state up to which the process is already brought to enable the production of customer-specific applications as fast and efficiently as possible, like it is illustrated in Figure 2. Developing a product up to platform level is more expensive and time consuming than customization. Once the product platform exists, customization processes can be done fast, efficiently and parallel for different customers.

In company and company networks point of view platform concept has a rich applicability. Some of the main advantages are as follows:

- Platform concept enables product management and provides tools for product lifetime support.
- By applying the platform concept, one can take advantage of both mass production and customization benefits. In other words the platform concept enables mass customization.
- In addition to products, the platform concept can also be applied to processes, services and to the whole business concepts.
In the context of wireless sensor networks, Hill et al. (Hill 2004) presented a classification to four different platform classes. Even though the hierarchical classification they presented provides one way to divide heterogeneous sensor network to different classes, it does not meet the requirements of wireless automation. Less attention is paid to the data processing in sensor network during data aggregation and too much is counted for TinyOS operating system (TinyOS 2009). TinyOS does not offer such a generic level that it would enable fast and easy modifications depending on the particular application needs. It is also suffering about weak documentation and uncoordinated application development. Moreover, it does not offer compatibility with other embedded systems typically used as a part of the automation system. The role of IP-based networking is also underestimated. IP based networking with IPv4 or IPv6 is playing an important role when wireless and cabled parts of the automation system are integrated together. In wireless sensor and actuator networks this support can be offered, for example, by 6LoWPAN compliant protocol stack (6 LoWPAN 2009).

**Wireless Automation Standards**

Recently, the Hart Communication Foundation (HCF) released the WirelessHART (WirelessHART 2009) standard for wireless automation and some field devices and sensor equipment supporting the specification are already available on the market. The core protocol of WirelessHART is the TSMP (time-synchronized mesh protocol) [Reference], which defines the communication slots in time and frequency for the nodes in the network. The main features of WirelessHART include: time-synchronized communication, self-organization and self-healing, frequency hopping (and channel blacklisting), security (encryption, authentication, integrity), and full mesh networking (path redundancy). WirelessHART is the first standard on wireless communication in industrial automation. Another standard, the ISA100.11a is also coming up during the year 2009 (Isa 2009). It has similar features as WirelessHART, but it is targeted to systems with faster cycle times. The WirelessHART standard currently specifies a minimum communication cycle time of 1 s, and the equipment on the market can support 4 s...
communication rate. It is obvious that the standard as such is not useful in all applications, because of the long duty cycles. Typically in the control systems we consider the duty cycles can be up to 100 times faster. Nevertheless, TSMP has several appealing properties and techniques that would be useful if implemented as light-weight versions on the wireless sensor networking platforms mentioned above.

**Combined Platform Development in GENSEN**

In GENSEN project we combine the requirements set by three different platform development points of view. It is important to notice that these platform requirements effect each other and the best generic solution can be achieved by evaluating the platforms against each other since the early stages of the development process.

**Applications**

In GENSEN we build five different types of applications: wireless part of the trolley crane control system, monitoring and control of wind turbine generators, monitoring and control of distributed energy production system, greenhouse automation and cattlehouse automation. The application development will be done with the companies who participate to the GENSEN project and operate in the pre-mentioned areas. It is obvious that the system requirements of each of these applications differ from each other, but the common part of the requirements can be found by comparing them to each other. By doing so we can find the level up to which the common technology platform can be developed. This level can be increased by following the maximal requirements, but it must be done in such a way that it does not cause useless additional costs or useless burden in the applications with lower requirements. The definitions of the application requirements offer a lot of detailed information for the application platform development, such as network performance requirements in terms of lifetime, reliability and data throughput; sensor node size, cost and performance requirements; the requirements of the physical environment; the requirements of the compatibility with the rest of the automation system etc.

**Application Platform**

We are developing such application platform, which allows a fast production of applications as illustrated in Figure 2. In practice it means that the main objective is to develop a generic wireless sensor network architecture, which scales up for different numbers of sensor nodes, enables advanced networking and data processing and is easy to equip with different types of existing commercial sensors depending on the particular application needs. Advanced networking properties such as multi-hop support, automatic network configuration, reconfiguration and advanced data processing are available in the application platform such that they do not need to be tailored separately for different applications. The purpose of this architecture is to enable flexible configuration of sensor networks to be applied in various applications with dramatically less tailoring compared to the state of the art solutions. The core of the concept is a flexible protocol stack developed especially for the needs of time-critical applications, such as wireless automation. The design of the generic wireless sensor network architecture requires research and development in the following main tracks:

1) **Technology platform development**
   - Integration and definition of generic, compatible hardware components (nodes, sensors, radios)
   - Software and driver libraries for the hardware (including protocol stacks)
• Configuration tool to integrate the application design

2) Protocol development
• A flexible communication and networking protocol stack for the generic sensor network platform
• Implementing the useful and energy-efficient parts of the time-synchronized mesh protocol used for example in WirelessHART
• Better support for asynchronous sampling, data compression, data fusion and control algorithms as required in the technical requirements of the application platforms

3) System validation and testing
• Development of five applications
• Interaction between application platform and technology platform development

4) Evaluation of the commercialization capabilities
• Marketing opportunities of the developed technology platform and application platforms
• Technical requirements set by the commercialization process

The important requirements we set for the applied sensor platforms are open software and IP-based networking with IPv4 and IPv6 protocol support. We apply Sensinode Microseries and such versions of Sensinode Nanoseries which still provide open software (Sensinode 2009). In addition to further development of Sensinode Nanoseries we will also consider such platforms, which support Contiki operating system (Contiki 2009). Contiki is an open source, highly portable, multi-tasking operating system for memory-efficient networked embedded systems and wireless sensor networks. A typical Contiki configuration can have 2 kilobytes of RAM and 40 kilobytes of ROM memory. Contiki provides IP communication, both for IPv4 and IPv6 protocols. Contiki is developed by a group of developers from industry and academia led by Adam Dunkels from the Swedish Institute of Computer Science (SICS). The Contiki team consists of sixteen developers from SICS, SAP AG, Cisco, Atmel, NewAE and TU Münich. Currently Telos is offering support for Contiki operating system, and Sensinode is going to announce its own Contiki support during the year 2009.

The work which will be done in the context of sensor platforms consists of the software implementation of the developed algorithms and the software and hardware integration. We will use commercially available sensors in our application platform. The types of the applied sensors depend on the particular application needs. We may also develop our own hardware solutions compatible with the pre-mentioned sensor platforms to make them more useful in variety of applications. The main reasons for the need of adding certain hardware components is that many of the sensor network hardware platforms have only single microcontroller that handles both the sampling, data processing and communication protocol related tasks. The system is energy efficient, but the architecture imposes constraints on the utilized applications and protocols. For instance, the node cannot sample data and transmit at the same time. More flexible platform could be achieved by using two microcontrollers where one is dedicated to sampling and data processing and the other is dedicated to running the communication protocols. In addition, the solution based on two microcontrollers could be made very flexible in terms of used radio system. An API could be defined for the application to access the communication module. Same application system (sensors and processing) could be utilized
with various radios depending on the communication needs (required radio range, level of interference etc.). We simplify the classification presented by Hill et al. (Hill 2004) to respond better to the needs of wireless automation by proposing a sensor network architecture consisting of two kinds of nodes: 1) energy efficient small capacity nodes that could be utilized as simple sensors or relays, and 2) high capacity flexible nodes that could serve as soft sensors, cluster heads or gateways.

To manage the configurations of the sensor network hardware, software, protocols and algorithms in a particular application, we will develop a configuration tool, which will contain all the sensor drivers, communication protocols and application layer algorithms as well as configuration options of the nodes in libraries. With the tool, the appropriate hardware/software configuration is defined easily with the specifications of the communication duty cycles and mechanisms. The configuration tool will be integrated to the PiccSIM simulator (Kohtamäki 2009), which is a complete co-simulation and co-design environment for wireless networked control systems and wireless automation.

We are targeting to offer an application platform for such automation systems that may include tens or hundreds of measurement points. The flexibility the wireless communication provides must be preserved such that one can easily add, remove or re-locate sensor nodes in the network based on demand on the fly. Thus, after the initial deployment the network must automatically configure itself. The network configuration includes routing, localization, time-synchronization and the assigning of frequencies and network addresses. Clustering can also be included to the configuring operations, if hierarchical network architecture is applied (Virrankoski 2005). Once the initial configuration is completed, the network must maintain that information and also adapt to the possible changes in the network. During the network operation, some nodes may start to malfunction, some may run out of power, some may be removed and some new nodes may be added. There may also exist network level malfunctioning. Hence, network monitoring and diagnostics, together with self-healing mechanisms are important to ensure reliable operation and to identify the sources of malfunctioning. The network solution we develop applies IEEE 802.15.4 communication protocol. The system must support the co-existence of IEEE 802.15.4 and IEEE 802.11 because of the widely spread wireless local area networks, which apply the latter protocol.

The beacon mode of the IEEE 802.15.4 MAC supports network synchronization and slotted CSMA with guaranteed time slots which enable some devices to do contention free transmissions. In real-time applications, synchronized collision free transmissions are needed to minimize transmission delays and jitter. However, the design objective of the beacon mode has been power saving by introducing inactive periods for sleeping. The beacon mode of IEEE 802.15.4 has been found to be very complex and have low throughput, see e.g. (Ko 2006). Hence, we believe that there is a need for MAC protocol that is designed especially for real-time applications.

The TSMP protocol has been designed with the above restrictions in mind. The key properties of the protocol are: 1) time synchronized communication, 2) frequency hopping, 3) automatic node joining and network formation, 4) fully redundant mesh routing, 5) secure message transfer, hence fulfilling all the requirements posed above. The downsides of the protocol are the need for the coordinator node that continuously assigns the time and frequency slots and takes care of node joining etc. Besides the duty cycles achieved, for example, with WirelessHART are too slow for our usage. We will adjust the protocols applied in our application platform such that it better suits the applications we consider and such that the achievable duty cycle will be enhanced. We will define and propose a “TSMP-Lite Protocol” and implement it to the
application platform and the configuration tool with simultaneous support for 6LoWPAN (IP-based networking in WSNs).

Applied network topologies can include sub-networks consisting of very simple, small-size, short range, low-cost and non-standard sensor nodes. These kinds of sub-networks are connected to the rest of the system via bridges with the interface including the standard platforms.

So far many wireless automation applications are still focusing on data collection. In these cases the whole wireless communication acts as a cable replacement; all data is collected to the centralized sink, control decisions are made there and then control commands are sent back to the actuators. There are many reasons why this is not an efficient way to operate. First, sensor nodes have scarce resources that should not be wasted to the continuous transmission of such a data, which is not necessarily relevant. Second, the useless redundancy of the data is already a burden in many systems. Even though the sink has remarkably higher computation and memory resources compared to any of the nodes in the network, useless amounts of redundancy can still cause longer delays and a remarkable waste of resources in the system. Third, bigger amounts of data will increase the risk of packet losses and other data transmission errors in the network. These errors will then increase the network latency and they may even lead to serious network malfunctioning.

We apply spatial correlations and blind compression methods to eliminate the useless redundancy from the data as near the measurement points as possible. We will pay attention to both node level and cluster level data compression. If there is more than one level of hierarchies in the network, the hierarchical structure can also be exploited to create different abstraction levels of the data.

We will also test position assisted data compression: By using the location information, every physical position or area generates only one data flow independent on the actual number of sensor nodes in the area. Once this method is applied, the network can be addressed by positions instead of individual node addressing.

One way to reduce the amount of transmitted data is adaptive and asynchronous sampling. Instead of sampling in a constant frequency, each node can adapt its own sampling rate based on the measured data. In the condition monitoring, for example, there is no need for high sample rate or data transmission, if everything works as expected. If the measured data starts to indicate exceptional behavior, the node should automatically report about the change and also increase its sample rate to be able to figure out more precisely what is actually happening. The amount of node and network resources sacrificed to trace down the malfunctioning must depend on the seriousness of the observed phenomenon.

Usually more than one type of sensor data is required in advanced control systems. The sensor nodes we are applying enable simultaneous use of several types of sensors. Thus, we need data fusion algorithms to combine the multi-sensor data to be able to perform the correct control operations. Data fusion can be done in a partially distributed (or locally centralized) manner in the network cluster heads, or it can be done in a fully centralized manner in the network sink.

We consider data fusion methods typically applied in control engineering, such as Kalman Filter, particle filters, fuzzy computation and Hidden Markov Models. If these methods are applied in a centralized manner, one challenge is the real-time requirement. The data fusion computation should not cause so long delays to the network operation, that it would seriously violate the system real-time requirement. In locally centralized computation one challenge is to find such
light-weight implementation of pre-mentioned methods that it can be executed with a limited memory and computation resources.

**Product Platform**

Usually the slowest and most expensive part in a product development is the period during which the application platform will be developed up to such a level, which allows a fast production of different types of applications. Once the platform is completed, the application development is much faster and cheaper. Developed application platform can be used with a wide spectrum of automation and monitoring applications such that in each particular case only minor modifications are needed to produce the particular application.

Final step is the development of product platform. Developed application platform itself as well as each of the five validation applications we produce during the project can be commercialized. Some key requirements of the product platform, such as market opportunities, costs, system usability, standardization and possibilities to offer technical support for the application must be taken into account in the application platform development. On the other hand, mandatory requirements rising from the technology platform as well as practical issues observed during the application platform development will guide the development of the product platform.

**Conclusions and Future Work**

In this paper we made a survey to the field of wireless automation and presented a way to combine technology platform, application platform and product platform development such that it can be done in parallel. Moreover, the platform development processes interact with each other in beneficial way as illustrated in Figure 3.
As a part of the GENSEN-project on 2009-2010, we will evaluate the validity of the presented combined platform development. We will also clarify the interactions between technology platform, application platform and product platform development. Some marketing research in the discussed application field will be performed to be able to specify the steps to the developed application platform commercialization.

References


Contiki website (2009), http://www.sics.se/contiki/

Crossbow website (2009), http://www.xbow.com


Imote2 datasheet (2009),


Sensinode website (2009), http://www.sensinode.com


WirelessHART; HART Communication Foundation website (2009), http://www.hartcomm.org

6LoWPAN website (2009), http://www.6lowpan.org

Biographical Notes

Mr. Reino Virrankoski (Lecturer of Telecommunications, M.Sc.) received his M.Sc. in Mathematics from the University of Helsinki on 2000 and he is a Ph.D.-student in Control Engineering at Helsinki University of Technology. On 2000-2007 he was working as a Graduate Researcher at Helsinki University of Technology. During that time he also worked as a Visiting Assistant Researcher at Yale University on 2004-2005. Currently he holds a faculty position as a Lecturer of Telecommunications at the University of Vaasa. Mr. Virrankoski’s main research interests are communication and control in telecommunication systems and in wireless sensor networks, wireless automation, localization and controlled mobility and wireless networks in defense and security.

Mr. Simo Keskinen (Senior Researcher, Licentiate of Economics) is currently working as Senior Researcher in the Department of Production at University of Vaasa. His research focuses on development of infrastructure networks, technology management, production and product development. He has been working in industrial R&D tasks in Salora Oy (later a part of Nokia) and in teaching tasks in the polytechnics of Oulu and Vaasa.