

Web Services to Improve Interoperability of Home Healthcare Devices

Guido Moritz¹, Elmar Zeeb¹, Frank Golatowski², Dirk Timmermann¹, Regina Stoll³

¹Institute of Applied Microelectronics and Computer Engineering, University of Rostock, 18055 Rostock, Germany
{guido.moritz, elmar.zeeb, dirk.timmemann}@uni-rostock.de

²Center for Life Science and Automation, 18119 Rostock, Germany,
frank.golatowski@celisca.de

³Institute of Preventive Medicine, University of Rostock, 18055 Rostock, Germany
regina.stoll@uni-rostock.de

Abstract—Home healthcare can be divided into monitoring and assistance scenarios. In both scenarios, miscellaneous devices are applied for dedicated sensor or actuator functionalities. The requirements and assumptions for the different nodes split the devices in three different classes with particular characteristics concerning energy consumption, power supply, memory, computing power, and bandwidth. Nevertheless, the number of nodes and complexity of setups require new concepts like Service-oriented Architectures to solve arising problems. In this paper, the Devices Profile for Web Services is suggested as comprehensive middleware for all three device classes. Hence, Web services technologies, which are already applied in the internet and in automation industry, can avoid interoperability problems of sensors and actors in the domain of home healthcare also.

Keywords: *Web Services for Devices, DPWS, lwIP, 6LoWPAN*

I. INTRODUCTION

In the domain of distributed systems, the increasing numbers of devices are an issue that has to be solved. Service-Oriented Architectures (SOA) and their related concepts have been shown applicable and have been proven in different projects [7, 8]. The individual functionalities of a particular device can be encapsulated in services. These services allow a much higher abstraction of communication and information flow between the devices. Such a device centric SOA (see SODA [4]) can provide an abstract middleware for the inter device communication. They are characterized by independency of operating systems, programming languages, and communication channels. A single physical device can host several services, which can be accessed by different logical and physical interfaces. Moreover, complex services can appear as a single service where further calls to subservices are hidden from the service user. In the domain home healthcare, the increasing number of heterogeneous devices claims numerous efforts to improve the interoperability. In section II, this paper presents assumptions of device classes within the domain of home healthcare. Section III outlines a motivating approach, to realize device centric SOAs in the domain of e-Health, before section IV briefly concludes the results.

II. ASSUMPTIONS

Proprietary solutions of devices and communication concepts from different vendors result in isolated applications and non interoperable devices. The devices architecture for home monitoring of patients has special characteristics, which will be presented in this section. Furthermore, this section will divide the required devices in three different classes.

Two classes of scenarios of interacting devices exist, where the interoperability of miscellaneous devices has to be concerned. The first are monitoring scenarios of healing processes. The vital parameters of the patient are observed and stored in databases on devices at home or transferred through a home gateway to external servers. These servers are accessed by health professionals, who can adapt the use and dosage of medicine or propose further therapies. The second class of scenarios are assistance scenarios. These scenarios can be formed independent or in addition to monitoring scenarios. Assistance systems consist of movement sensors, sensors for monitoring the environmental conditions like temperature and gas anomalies, and actors to support the patient in daily living. Hence, sensors and actors have to cooperate in contrast to monitoring scenarios, where only sensors are applied. Such assistance technologies allow a self-determined living for the patient in their familiar home environment and can help to avoid emergency situations. Therefore, interacting devices in the domain of home healthcare can be divided in three different classes.

A small number of devices are forming the Wireless Body Area Network (WBAN). Vital parameters of the patient which cannot be measured ambient are recorded by these devices. These sensors are deeply embedded devices in most cases. Deeply embedded device are small microcontrollers with only few kB of RAM and memory and are forming the first class of devices. Deeply embedded devices are essential in monitoring scenarios. The sensor nodes of a WBAN cannot be provided with heavy and space consuming batteries due to requirements in size and weight for the sensor. Hence, energy constraints are the most significant requirements for this first class of devices. Concerning energy consumption, the most expensive operations are sending and receiving of messages. Therefore, the sensor node is set into a sleep mode as often as possible.

TABLE I. OVERVIEW OF DEVICE CLASSES

| | Deeply embedded devices | Resource constrained devices | Resource rich devices |
|----------------------------------|-------------------------|------------------------------|-----------------------|
| Energy constraints | +++ | + | -- |
| Power supply | Low capacity battery | Battery or power outlet | Power outlet |
| Computing power | --- | - | +++ |
| Data transmission rate and range | -- | - | +++ |
| Data transfer | Wireless | Wired and wireless | Wired and wireless |
| Memory constraints | +++ | + | --- |
| Size and weight requirements | +++ | + | -- |

Permanent polling of parameters and continuous transmitting of not necessary values must be avoided. The second class of devices are sensors and actors distributed in the home environment of the patient. They can be provided with the required batteries for an applicable operational life span or are connected to mains power system. This second class still has limited resources, due to avoid high costs and the minimized size to embed the devices in the environment. Resource rich devices are required in monitoring and assistance scenarios as well. These resource rich devices are forming the third class of devices. Data records have to be stored in comprehensive data bases and formats, wherefore a huge amount of storage space is necessary. Furthermore, computation intensive analysis of dedicated data records can be realized on these devices. By the storage and analysis of personal health records, these devices can also act as gateways, to assure the communication with and synchronization of data bases hosted on external servers. An overview of the three different device classes and their characteristics is given in TABLE I.

Not in all setups, a continuous monitoring of dedicated parameters is necessary. Event driven communication between devices with small payload has to be taken into account also. Generic interfaces to provide universal interoperability between the devices is more important. Most of the sensors within the home environment remain in sleep mode. They only have to report their measured values if they are invoked directly. Two examples: If the motion sensors are recording unusual behavior of the patient, the body area network sensors are invoked, to report the vital parameters. In another scenario, the vital parameter observing sensors have maximum levels for the recorded values. Recorded values within these limits do not have to be considered. Communication and reporting is only necessary in case of exceeding the limits. Thus, event driven communication can help to handle resources more energy efficient.

Three different classes of devices have been presented for home healthcare scenarios. From resource rich devices with constant energy supply, to embedded devices with limited resources, down to deeply embedded devices with highest constraints for computing power, memory, energy, bandwidth, and transmission range. Existing device architectures and concepts are based on gateways and proxies. Challenges

concerning the high constraints of embedded and deeply embedded devices are solved by transferring almost all functionalities on resource rich devices. These devices act as proxies and/or gateways for the underlying sensors and actors. Functionalities and services are mapped on these devices and requests to and from the sensors are converted from proprietary protocols on one side of the proxy/gateway to proprietary protocols on the other side. Hence, these concepts are vulnerable because of their centralized architecture. Furthermore, not necessary transmission might occur, if event driven communication cannot be provided by the protocols. Additionally, significant efforts have to be done, to convert the different protocols into compliant forms. Gateway and proxy concepts require the payload – the measured values and the designated commands from and to the sensors and actors – to be translated to fit to the according further protocols and data representation formats. Hence, deployment of new sensors and actors requires new converters to be implemented. The deployment of special profiles, like the Health Devices Profile for Bluetooth [3], improve the situation few only. An interaction between existing IP based networks and the sensors is still not possible.

Our vision is a comprehensive middleware to integrate all three classes of devices in existing and in future networks. This middleware can be applied on resource rich devices and wireless deeply embedded sensor nodes as well. This would allow a much higher abstraction of devices and the communication. For the interaction independent of the device class, no protocol conversion has to be done. Between wired and wireless devices, only routers have to be applied for seamless connection of the miscellaneous networks. Combined with standardized data representation formats like units and scales, such a comprehensive middleware can overcome the difficulties concerning the missing interoperability.

III. LIGHTWEIGHT DEVICE CENTRIC SOA

In the domain of Wireless Sensor Networks (WSNW), energy, computing power, and memory aware protocols already have been developed and tested. Sensor nodes of WSNWs have to fulfill the same requirements as deeply embedded devices forming a WBAN. The application of IP, TCP, UDP, and further protocols has been proven in miscellaneous other domains and are considered as the basis for a comprehensive middleware for the dedicated domains and for home healthcare as well. In [5], Jardek et al. have implemented nanoIP and miniaturized and resource optimized versions of TCP, UDP, and HTTP and have tested the performance. Jardek et al. have also shown that the used protocols can result in high throughput with low latencies and without significant performance losses. However, the presented architecture and protocols are still designing isolated applications with local addressing concepts, where a useful communication to external networks is still possible by dedicated gateways only. Caused by this drawback, compliant IP, TCP, UDP, and HTTP protocols have to be deployed in favor.

A. Web Services in Home Healthcare

Service-oriented Architectures are often mentioned in the same breath with Web Services (WS) [1], which are specified

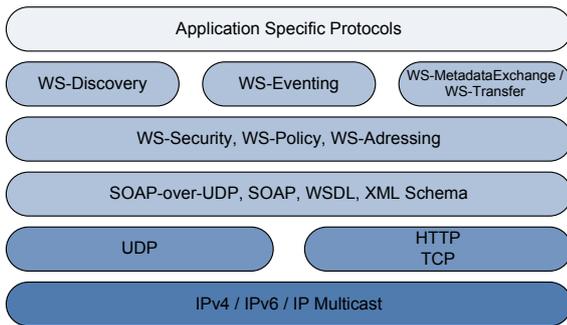


Figure 1. DPWS Protocol Stack

by the W3C. Certainly, the WS-* standards often need too many resources to be applied on devices with limitations in computing power, memory, and energy. Therefore, a group led by Microsoft has specified the Devices Profile for Web Services (DPWS) [2] that is standardized by OASIS now under the abbreviation WS-DD - Web Services Discovery and Web Services Devices Profile - [20]. This specification combines a set of Web services standards and adds several supplements, to enable Web services functionalities on resource constrained devices. The DPWS stack is depicted in Figure 1. The deployment of DPWS assumes the existence of IP on one hand and the capability to handle XML on the other hand. DPWS concerns all assumptions to provide the required interoperability of former presented dedicated devices and device classes in home healthcare. The specification includes mechanisms for automatic finding of devices and services and generic description of interfaces. Additionally, eventing concepts are included. A main difference to other protocols is the optional central service registry. Every device can announce itself and search for dedicated services itself. This decentralized architecture is less vulnerable and more scaleable. But some more efforts have to be done, to adopt DPWS for home healthcare scenarios and to provide the missing middleware for all classes of devices.

For embedded devices with limit resources, often no underlying operating system can provide network functionalities like IP, TCP and UDP. Dunkels et. al have developed lwIP, a standard compliant TCP/IP stacks for 8 Bit controller architectures [9, 10]. The major focuses are minimal footprint and reduced computing power usage on the controller, without losing standard conformance. lwIP also fulfills non mandatory features of TCP/IP and is designed to run with and without an operating system. DPWS makes use of WS-Discovery for discovery of devices, which is based on IP Multicast. Multicast applications use the connectionless and unreliable User Datagram Protocol (UDP) in order to achieve multicast communications. lwIP can fulfill these requirements and should be applied as network stack for a lightweight DPWS implementation.

In [18, 19], we presented the table driven approach for Web services on deeply embedded devices. The recommended table driven approach embeds all possible associated messages of possible scenarios for the devices at compile time. Only minor adoptions of the messages have to be done during runtime. Additionally, the incoming messages are parsed with an elementary string compare, which is also available and possible

on a deeply embedded device. Nevertheless, the table driven approach is only applicable in static scenarios with dedicated functionalities and knowledge of devices architecture at compile time. In opposite to the table driven approach, the focus of this paper is to implement full DPWS compliant functionalities on resource constrained embedded devices. Our research group has implemented the WS4D-gSOAP toolkit [12]. This is a software solution which includes a DPWS stack and software tools for creation of own Web services based on DPWS. This toolkit extends the gSOAP Web services toolkit [11] with an implementation of the DPWS specification. As an extension, we have now ported the existing WS4D-gSOAP toolkit to lwIP. Thereby, the WS4D-gSOAP toolkit gets independent from the underlying operating system and can be applied on embedded and resource constrained devices with or without operating system. Hence, it is required to adapt the network hardware driver only, to port the toolkit to new platforms. The footprint of the executable, compared to standard WS4D-gSOAP for Linux including standard BSD sockets and network stack, is increased from 354kB to 435kB. The additional 81kB are acceptable for resource constrained embedded devices, but will be reduced in further development.

B. Web Services on Wireless Deeply Embedded Devices

With respect to power consumption, the IEEE 802.15.4 standard [13] for wireless personal area networks is gaining in importance. 802.15.4 meets all requirements to be applied on energy constrained sensor nodes and on deeply embedded devices. In accordance to the IPv6 specification, the IETF has established the 6LoWPAN Working group [14]. The focus of 6LoWPAN is to find possibilities, how to compress IPv6 Headers [15] to be sent on top of 802.15.4. Therefore, 6LoWPAN omits all unnecessary addressing and packet information, which are already embedded in the MAC and PHY layer of 802.15.4. To save energy and to improve the ratio of header to payload, the addressing scheme of IPv6 is adapted. Thus, 6LoWPAN differentiates between local messaging with other 802.15.4 devices and external messaging with IPv6 networks. For local messaging, full compressed headers and addresses are used, where header/payload ratio is even better than the encoding defined by the ZigBee Alliance [17]. For external messaging, default IPv6 headers and addresses can be used. 6LoWPAN establishes the basis for TCP and UDP data [16] transmissions for deeply embedded devices. The main advantage of 6LoWPAN over nanoIP [5] and other proprietary protocols like ZigBee, is the compliance to existing network addressing schemes. For the external transmission out of the sensor network, 6LoWPAN has to be used in favor. For the transmission between internal (6LoWPAN) and external (IPv6) networks, only a compliant router is necessary and no proxy or gateway concepts have to be used. This supports the assumptions made in section II. On top of IPv6 and 6LoWPAN, middleware like DPWS can be applied. An overview about such a protocol stack in comparison to e.g. ZigBee is illustrated in Figure 2. Dunkels et al. have implemented the 6LoWPAN specification in the open source operating system Contiki for resource constrained devices. Porting the WS4D-gSOAP toolkit to 6LoWPAN is an ongoing work. Thereby, a comprehensive middleware for deeply embedded devices, resource constrained devices, and devices with rich resources can be provided.

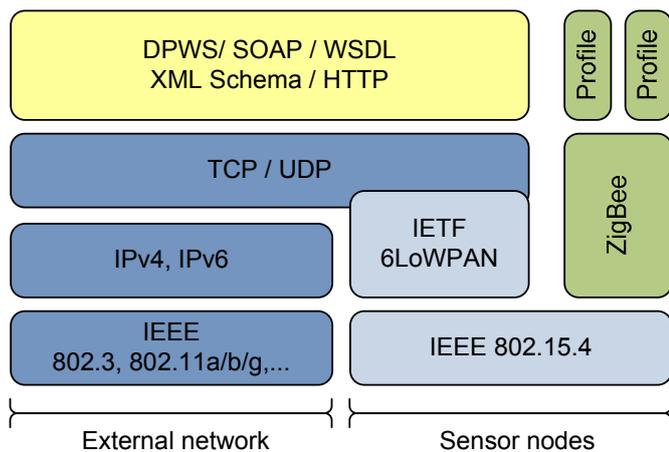


Figure 2. Comprehensive Middleware

IV. CONCLUSION

This paper presents, how to enable Web services functionalities in home healthcare scenarios to concern the lack of interoperability of devices. Therefore, healthcare scenarios are divided in two different classes: monitoring and assistance scenarios. Furthermore, all participating nodes and devices are divided in three classes with dedicated characteristics. Deeply embedded devices are assuming the highest requirements concerning energy consumption, computing power, memory, and bandwidth.

We presented, that it is possible to realize the Devices Profile for Web services on top of lwIP. This Open Source TCP/IP and UDP/IP implementation for resource constraint devices can be deployed on devices without underlying operating system. Additionally, we have described 6LoWPAN, which can provide IP addressing for deeply embedded devices on top of IEEE 802.15.4. Thereby, DPWS can act as the required comprehensive middleware and connect deeply embedded devices and devices with rich resources. The same technology allows a single sensor node to interact with other nodes or with complex Web services in external networks like the internet and allows a seamless integration in existing and future networks.

REFERENCES

[1] World Wide Web Consortium, *Web Service Architecture Specification*, Technical report, <http://www.w3.org/TR/ws-arch/>, 2007.
 [2] Microsoft, Intel, Ricoh, Lexmark, *DPWS Specification*, Technical report, <http://specs.xmlsoap.org/ws/2006/02/devprof>, 2006.

[3] Bluetooth Special Interest Group (SIG), *Health Device Profile*, Technical report, 2008.
 [4] S. de Deugd, R. Carroll, K. E. Kelly, B. Millett, and J. Ricker, "SODA: Service-Oriented Device Architecture," *IEEE Pervasive Computing*, vol. 5, no. 3, pages 94-C3, 2006.
 [5] C. Jardak, E. Meshkova, J. Riihijarvi, K. Rerkrai, and P. Mahonen, "Implementation and Performance Evaluation of nanoIP Protocols: Simplified Versions of TCP, UDP, HTTP and SLP for Wireless Sensor Networks," *Wireless Communications and Networking Conference, 2008. WCNC 2008. IEEE*, vol., no., pp.2474-2479, March 31 2008-April 3 2008
 [6] S. Karnouskos, O. Baecker, L. Moreira, S. de Souza, and P. Spieß, "Integration of SOA-ready networked embedded devices in enterprise systems via a cross-layered web service infrastructure," *Emerging Technologies & Factory Automation (ETFA)*, Patras, Greece, pages 293-300, Sept. 2007.
 [7] H. Bohn, A. Bobek, and F. Golatowski, "SIRENA - Service Infrastructure for Realtime Embedded Networked Devices: A service oriented framework for different domains," *International Conference on Systems and International Conference on Mobile Communications and Learning Technologies (ICNICONSMCL'06)*, page 43, Washington, DC, USA, 2006.
 [8] LOMS: Local Mobile Services, <http://www.loms-itea.org>, 2008.
 [9] Adam Dunkels, "Full TCP/IP for 8-Bit Architectures," *International Conference On Mobile Systems, Applications And Services*, San Francisco, California, pages 85-98, 2003.
 [10] Adam Dunkels, *lwIP - A Lightweight TCP/IP stack*, Technical report, <http://savannah.nongnu.org/projects/lwip/>, 2008.
 [11] Robert A. van Engelen, Kyle A. Gallivany, "The gSOAP Toolkit for Web Services and Peer-To-Peer Computing Networks," *2nd IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGRID)*, Washington, DC, USA, page 128, May 2002.
 [12] University of Rostock. *DPWS-Stack WS4D*, <http://ws4d.org>, 2007.
 [13] IEEE Computer Society, *Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs)*, Technical report, 2006.
 [14] IETF Network Working Group, *IPv6 over Low power WPAN (6lowpan)*, Technical report, <http://tools.ietf.org/wg/6lowpan/>, 2008.
 [15] IETF Network Working Group, *Transmission of IPv6 Packets over IEEE 802.15.4 Networks*, Technical report RFC 4944, <http://tools.ietf.org/html/rfc4944>, 2008
 [16] IETF Network Working Group, *Compression Format for IPv6 Datagrams in 6LoWPAN Networks*, Technical report, <http://tools.ietf.org/html/draft-ietf-6lowpan-hc-03>, 2008
 [17] Kinney P. *ZigBee Technology: Wireless Control that Simply Works*, Technical report, 2008.
 [18] G. Moritz, S. Prüter, D. Timmermann, and F. Golatowski, "Real-Time Service-oriented Communication Protocols on Resource Constrained Devices," *International Multiconference on Computer Science and Information Technology (IMCSIT2008)*, Proceedings on, vol. 3, pp. 695 - 701, Wisla, Polen, 2008
 [19] G. Moritz, S. Prüter, D. Timmermann, and F. Golatowski, "Web services on deeply embedded devices with real-time processing," *Emerging Technologies and Factory Automation, 2008. ETFA 2008. IEEE International Conference on*, vol., no., pp.432-435, 15-18 Sept. 2008.
 [20] OASIS Web Services Discovery and Web Services Devices Profile (WS-DD) TC, <http://www.oasis-open.org/committees/ws-dd/>, 2008.