Integrating Robots and WSN: communications and interfacing aspects

Christos P. Antonopoulos, Christos Panagiotou, Konstantinos Antonopoulos, Alexandros Spournias ,Nikolaos Voros,
Embedded System Design and Application Lab
Computer and Informatics Engineering Department
Technological Educational Institution of Western Greece, Greece
{cantonopoulos; chpanag;voros}@teiwest.gr,
{kostasantonop;alexspou1}@cied.teiwest.gr, voros@teiwest.gr

Fynn Schwiegelshohn, Philipp Wehner, Michael Huebner
Ruhr-University, Bochum, Germany
{fynn.schwiegelshohn;philipp.wehner;michael.huebner}@rub.de

Diana Göhringer
Technische Universitaet Dresden, Germany
diana.goehringer@tu-dresden.de

Raquel Ventura, Alberto Fernández
Sensing & Control Systems S.L., Spain.
{raquel.ventura; alberto.fernandez}@sensingcontrol.com

Georgios Stavrinos
Institute of Informatics and Telecommunications, NCSR "Demokritos", Greece
gstavrinos@iit.demokritos.gr

Evangelinos Mariatos
AVN Innovative Technology Solutions, Cyprus
emariatos@avntechgroup.com

Corresponding author’s email: *cantonopoulos@teiwest.gr*

Abstract

In the context of RADIO project, seamless integration of Robotic platform, heterogenous home automation technologies and diverse back infrastructures comprised an integral challenge to the success of the overall endeavor. This observation, gave the involved partners the chance

to explore difference technological and design approaches in order to yield the optimum implementation. At the same time, it highlighted the need to follow a structured, multifaceted approach in order to tackle efficiently all requirements. Based on this process, the main objective of this chapter is to present the most important lessons learned as well as emphasize on the key aspects where enhancements and novel contributions were achieved in RADIO.

Therefore, in the first section, after the conceptual communication architecture defined in RADIO is presented, critical communication requirements of the RADIO platform are identified focusing on local area ultra-low power communication technologies deployed in the home premises as well as efficient cloud infrastructure utilized. Then the second section highlights the actual state-of-the-art communication technologies explored, deployed and integrated in the context of RADIO platform emphasizing on bridging and coexistence critical challenges encountered and addressed. Focusing on the integration problem the RADIO Gateway entity represented one of the most important cornerstones for the overall RADIO platform performance effectively comprising a highly complex compound entity, adequately analyzed in Section 3. Then in Section 4 important research activities based on BLE communication technology are presented, enabling, on one hand, accurate indoor localization and, on the other hand, extended coverage area capabilities through multi-hop communication. Finally, Section 5 presents all development and integration effort devoted by RADIO partners to the backend infrastructure. Although, many times, not highlighted adequately, back-end infrastructure is a critical part of the overall architecture since it effectively comprises the bases for key services such as, data storage, data processing, data representation, network management, network configuration etc.

1. Conceptual RADIO Communication Architecture

1.1. End-to-end Conceptual Communication Architecture

The operating environments targeted by RADIO are domestic homes of elderly people. These homes, generally, do not have sufficient technological infrastructure to provide ad-hoc ambient assisted living services. In order to guarantee that the impact of the RADIO system is not limited by requiring specific communication infrastructure to be prior deployed at the end-users' homes, the infrastructure connecting the RADIO components represents a substantial part of our architecture design. Specifically, the RADIO communication architecture includes: The wide-area communication between each RADIO Home and remote components, such as storage and processing facilities at the hospital or notification functionalities at care-givers' devices. The local, mostly (or even exclusively) wireless communication of the components deployed within each RADIO Home.

The RADIO Home environment itself comprises subgroups (Figure 1), each fulfilling a different task and designed in response to different requirements:

- Basic smart home: Off-the-shelf smart home devices
- Extended smart home: Advanced devices, integrating sensing and low-energy processing, as well as the RADIO Home server
- Mobile platform: Robotic platform, integrating sensors and limited computation functionality
- User interface devices

In order to provide long term support and reliability, the *basic smart home* sensor and actuator devices are off-the-shelf components such as Z-Wave products.

The *extended smart home subgroup* is based on Bluetooth to connect devices that carry out sensing and local processing in order to recognize daily activities and routines. This creates the requirement for a gateway that interfaces between Z-Wave and Bluetooth, in order to allow RADIO to combine the robustness offered by commercial Z-Wave devices with the flexibility to develop new services offered by Bluetooth.

Finally, the mobile platform establishes its own internal network in order to integrate its various sensing and processing elements, but also needs to connect to the overall RADIO Home. The platform is outfitted with two interfaces, the Bluetooth Low Energy (BLE) and the WiFi interface. BLE connectivity is provided to support direct access to devices of the extended smart home and WiFi connectivity for data transfer requirements.

This analysis gives us the architecture in Figure 1 where the Z-Wave network, the Bluetooth network and the WiFi network interface at a *central gateway*. The gateway also serves as the central data processing point, since it is connected to all available devices in the smart home environment. Finally, the gateway interfaces with remote components of the RADIO ecosystem over the Internet.

Figure 1. Conceptual architecture of RADIO communication channels

1.2. Requirements at WSN Level

Wireless sensor networks, represent one of the fastest growing ICT sector, although posing stringent requirements, especially in the case where incorporated into an intelligent and assisted living environment, such as the ecosystem of the RADIO project [**1**]. The first important point aiming towards wide spread of a wireless sensors network, constitutes its design, so as to offer a useful and practical system for home environment monitoring. Also, especially in the case where installation takes place in to user's private residence, it should offer high level protection of personal data, as well as minimal obtrusiveness or discomfort to the user [**2**].

So, in this section, we analyze the requirements posed on wireless sensor network technologies, aiming to meet the objective of RADIO project. Such requirements take into account the end-users requirements in conjunction with the restrictions imposed by the existing technology of wireless sensor networks.

1.2.1. Communication technology support

The ability to efficiently, reliably and securely transmit data wirelessly exploiting prominent communication technologies and protocols, comprises the most fundamental requirement for wireless sensor network. This ability is related the variety of protocols used in RADIO and with different characteristics and requirements that each sensor may pose as well, thus drastically affecting complexity and energy consumption [**2**].

1.2.2. Low Power Consumption

Based on the assumption of scares energy availability, it is important to manage the node, both at software, as well as at the hardware level, to minimize energy consumption and to increase energy autonomy. To meet this requirement, MAC layer protocols play an important role allowing the wireless interface to enter low power mode during periods of idleness.

Another important factor drastically attributing to the overall WSN power consumption is the processing units of each sensor. Thus, in this aspect, wireless sensor manufacturers now offer several solutions, such as the well-known TI MSP430 and / or the ARM Cortex M family [**1**] yielding ultra-low power current demands and multiple low-power operational modes.

1.2.3. Data Throughput Capabilities

An important role in designing the wireless network pertains to the amount of data that can be efficiently transmitted during a specific time interval, which for the needs of RADIO project is low, so that it can easily be adapted to existing solutions. Towards such objectives multiple prominent ultra-low power communication technologies are explored (e.g. Z-Wave, IEEE 802.15.4, BLE etc.) in order to optimally meet the data throughput requirements of the RADIO application scenarios.

1.2.4. Delay - Jitter requirements

Another requirement to be met, has to do with the delay variance of the receiving data packets from sensors particularly considering data streams. In this case, packets are sent in a continuous stream evenly spaced. Due to network congestion, improper queuing, or configuration errors, this steady stream can become lumpy, or the delay between each packet can vary. Respective technologies used must make sure that predefined time deadline is not violated.

1.2.5. Easy and Rapid Connection Support

On connection's supporting side between two nodes, there are two techniques, which are the connection-based and the connectionless, with respectively advantages and disadvantages.

Connectionless communication, on the one hand uses low complex protocol such as IEEE 802.15.4, as there is no need for additional mechanisms to establish and/or terminate a connection. Also, such a communication makes it easier to create multi-hop and mesh networks. On the other hand, they do not offer inherent support to QoS, robust data transfer, load balancing and in general traffic control, something that is required to offer in nowadays WSNs, as their applications increase the criticality, complexity and communication requirements.

On the other side, considering connection-based linking of two nodes, there is usually a scenario, where a node is used as a master and the other node as a slave. This technique provides data traffic load anticipation, Quality of Service, which allows to prepare an effective packet transmission timing. This approach is traditionally followed by Bluetooth technology. Each approach offers specific advantages and disadvantages and correct selection is critical in order to meet end-user demands.

1.2.6. Security Support

As the popularity and expansion of WSNs are increasing, the support of data security becomes mandatory, especially when applied into demanding areas such as health care and welfare of users, as it relates directly to the management and distribution of sensitive personal data. In RADIO project, three services are used to support security.

The first service is data privacy, which ensures that data can be useful only on nodes that can understand that. The second service, refers to the authentication of node, which ensures that it is authorized, to receive and/or send data. The last third service, provides authorization as it determines the level of access that can have a particular user to certain data and / or functions and is inextricably linked to the authentication service.

1.3. Requirements at back-end Level

RADIO poses several requirements for its backend platform as they are introduced by the multifaceted nature of the services that RADIO offers. IoT technologies lead to a rapid increase of the data sources and respectively to a massive growth of the volume of modalities that demand data storage and processing. This is the context that the Big Data problem describes. Particularly, the IoT and Big Data challenges are identified by the 3 Vs [**3**]. **3Vs** stand for **1) Volume, 2) Velocity and 3) Variety**. **Volume**, as the term implies, refers to the excessive amount of data that an IoT platform needs to handle. **Velocity** describes the high speed of the data flow, change and processing. Finally, **Variety** defines all kind of diversity that is present in data, data models, query languages and data sources.

In a more fine – grained view, these 3 main classes of challenges, that are also present in the RADIO infrastructure back–end level, can be broken-down in several other challenges that give more insights towards a modern IoT platform [**4**]:

Scalability: Scalability is not applied only in terms of the number of sensors and actuators connected to the system, or the networks that interconnect them. Rather scalability concerns the amount of data associated with the system and the data rate and the amount of processing power required. With the number of data sources integrated to an IoT system always increasing, the need for a scalable IoT Backend that scales efficiently is a major requirement that drives the architecture design of the backend.

Heterogeneous modalities: Modern IoT systems depend on the analysis of vast quantities of data. In order to extract patterns and meaningful information from high volumes of raw and heterogenous data (sensor readings, video footage, etc.) the need to efficiently support highly demanding processing tasks is required by the IoT backend platform. Such requirements can be addressed in pair with the need for scalability, through the adoption of microservices approach. Processing tasks can be identified as isolated microservices able to operate in a sandbox environment and scale efficiently when required.

Cloud computing: IoT backend systems involve the use of cloud computing platforms. Cloud computing platforms increased the availability of storage and processing resources and through the microservices approach achieve even more efficient utilization of these resources. Cloud computing and microservices are the two technologies that lead to the development of flexible and scalable cloud services and the efficient integration with IoT systems. RADIO backend comprises by two distinct IoT platforms, thus being highly adaptable and able to deal with new requirements, firmware or system updates and offer new capabilities over time.

Real time: IoT systems often operates in application domains where real time is required. Streams of data are continually transferred to the backend system which must respond in predefined time windows. The real time requirements concern, processing of the data stream, decision making based on the events produced and final reaction. In most time critical application domains, where real time is required, the result of violating the real time constraints could lead in catastrophic results for assets, infrastructures or even human life.

Highly distributed: IoT systems can span large geographical areas. In that sense, data can be stored at the edge of the network or stored centrally in the backend platform. While processing can take place at the edge of the network, either in the IoT gateways or even in sensors and actuators the most critical and intensive tasks takes place centrally in backend infrastructure and offered as cloud services. Despite the high level of distribution, IoT Backend platforms are a fundamental component of an IoT ecosystem that are responsible at least for persisting the state of the system and harmonizing raw data and extracted information. RADIO backend platform is the center of the RADIO system that guarantees the data storage, processing and service delivery for all involved end-users.

Security and Privacy: The mass adoption of IoT devices in various sectors of social and economic life that vary from personal residences to industry and health premises demand security and privacy concerns to be addressed convincingly. Regardless the application domain and its functional requirements, security and privacy of IoT systems is a requirement that is always present and thus, given solutions must scale and evolve with the systems. Security and privacy in RADIO is addressed through encrypted communication through adequate APIs, strong authentication and authorization mechanisms and encrypted data storage.

Compliance and Standardization: The wide variety of data sources and technologies have already lead to a fragmentation that make the need for standardization in IoT a challenge of major importance. Moreover, since these systems, as already noted, are applied in demanding environments must comply with rules and regulations that provide the appropriate confidence to the stakeholders of the IoT systems. Thus, RADIO APIs comply with adequate standard communication models.

Integration Capabilities: IoT evolution was driven by the need for delivering communication of physical devices (things) that are uniquely identified in a common network (Internet) and interact with other devices, services and end users. Therefore, the requirement for seamless integration for devices applied in various domains (home appliances, wearables, automation and control systems, online services, etc.) is highlighted as the cornerstone for IoT Platforms. In that respect, the RADIO backend platform communicates with the rest of the RADIO ecosystem (sensors, actuators, robot, gateways) through a homogeneous way offering both HTTP as well as MQTT messaging protocol support.

2. Integrated Wireless Communication Technologies and Bridging Considerations

2.1. Introduction to Z-Wave and Interfaces

Z-Wave is one of the communication protocols used within connected world to exchange information between smart products. A communication protocol defines how signals are sent from one device to another. Z-Wave operates at the low frequency 918/960 MHz band, meaning interference is minimal. Z-Wave uses AES-128 symmetric encryption for security. There are a wide range of devices compatible with Z-Wave, around 2,400 supported devices and 700 members associated. Some smart home brands and devices that support Z-Wave include: Samsung SmartThings, Wink hub, Honeywell thermostats, Hogar Milo, Somfy, LG SmartThiq.

Z-Wave devices use a proprietary protocol from Sigma Designs, http://www.sigmadesigns.com. Respective devices can communicate directly, but in the most common case, they form a wireless network being managed by a node called controller. The controller is responsible to manage the Z-Wave network and Z-Wave devices, while APIs are provided to the controller application in order to create simple or complex systems through the combination of the different device functionalities that forms the network. In the case of Sensing & Control's smart home system, the system provides functions for security, comfort, home automation and energy management.

Figure 2 shows the chosen hardware device that brings Z-Wave connectivity in order to communicate with Z-Wave products. The controller (gateway) and smart home application use the Z-Wave module called RaZberry for the Raspberry Pi.

Figure 2. zWave module (called RaZberry) for interfacing the gateways to the Z-Wave network

In the smart home, the Z-Wave gateway is the only point for communication between Z-Wave devices and the corresponding remote elements of the RADIO ecosystem, i.e. the respective IoT Platform.

In the first integrated prototype, the clients of the Z-Wave Smart Home devices needed to access devices through the RESTful API provided by the cloud-based IoT Platform. In the last prototype, the RADIO Home Controller was capable of communicating smart home devices with external services without using the cloud-based IoT platform. A fully local RESTful API was implemented that allows client functions to access data structures in Z-Wave and execute per devices and handling network management functions. This allows client functions to be integrated within the smart home application by running in the local controller.

Client functions integrated within the smart home application running in the Z-Wave gateway can result in high computational cost depending on the complexity of the client functions and the gateway's limited hardware resources (CPU and RAM memory). Therefore, this architecture requires a priori evaluation in order not to reduce the performance of the smart home functions already implemented into the gateway.

The local RESTful API implements the whole control logic of the Z-Wave network. The two main functions are:

- Management of the network. This includes adding and removing devices and managing the routing of the network. In the Z-Wave terminology all these functions are called 'function classes'. They are functions offered by the controller itself.
- Execution of commands through 'command classes' that allows the control of Z-Wave device functions. Command classes offer the variables and the commands according to the abilities of the respective devices. Command classes consist of two types of commands: commands for users (most of them are "GET" and "SET") and commands for configuration.

Most of the Z-Wave commands can be controlled by the local RESTful API but only a small subset is accessible for external applications for security reasons.

The basic Z-Wave commands accessible for external applications are:

Z-Wave command	**Type of Class**
sendData	FUNCTION_CLASS
AddNodetoNetwork	FUNCTION_CLASS
RemoveNodeFromNetwork	FUNCTION_CLASS
setValue	FUNCTION_CLASS
SetNodeLocation	FUNCTION_CLASS
SetNodeName	FUNCTION_CLASS
toggleActuatorSensor	COMMAND_CLASS
toggleDimmableSensor	COMMAND_CLASS
setThermostatSetPoint	COMMAND_CLASS
setThermostatMode	COMMAND_CLASS

setThermostatFanMode	COMMAND_CLASS

2.2. Introduction to BLE and Interfaces

Bluetooth is a wireless radio specification designed to replace cables as the medium for data and voice signals between electronic devices. The specification is defined by the Bluetooth Special Interest Group (SIG) which is made up of over 1000 electronics manufacturers. Intended primarily for mobile devices, Bluetooth's design places a high priority on small size, low power consumption and low costs. The Bluetooth specification seeks to simplify communication between electronic devices by automating the connection process.

Bluetooth radios operate in the unlicensed 2.4GHz Industrial, Scientific, and Medical application (ISM) frequency range. Since this frequency is already widely used by other devices, to avoid interference from these devices, Bluetooth uses a technology called spread spectrum frequency hopping. Spread spectrum frequency hopping changes the transmission frequency up to 1600 times per second across 79 different frequencies. As a result, interference on any one of those frequencies will only last a fraction of a second. This, coupled with the limited range of Bluetooth radio transmitters, results in a robust signal that is highly tolerant of interfering devices sharing the same frequency.

Bluetooth based solutions' performance can vary significantly depending both on the version of the protocol supported and even more on the specific implementation's characteristics. Therefore, concerning data rates solutions covering a wide range from 300Kbps up to 1.5Mbps can be found. Indicative examples of relative solutions include Shimmer [**5**] and MoviSens [**6**] platforms. The former is utilized in the Roving Networks based Bluetooth modules [**7**].

The latest version of the standard, Bluetooth Low Energy (BLE), represents a different technology from Classic Bluetooth (and in fact incompatible technology) being promoted by the Bluetooth Special Interest Group (SIG) and benefitting of the hugely successful Classic Bluetooth it shows significant dynamics compared to analogous technologies being incorporated. Furthermore, it offers high degree of flexibility both concerning implementation approaches and communication approaches supporting different ways for nodes to communicate through different data structure profiles to best fit the application requirements. Both these aspects were critical for the RADIO objectives highlighting relative solutions as good candidates for RADIO

purposes. The main fundamental step was to actually evaluate respective solutions and realistically verify that the performance offered is adequate for the goals of the project.
The increased adoption of the BLE technologies in the IoT domain has enabled researchers, companies and the Bluetooth Special Interest Group (SIG) to explore the feasibility of mesh networking over BLE. Currently, enhancements focused on mesh networking are on the roadmap of all the BLE related vendors. BLE mesh support is expected to provide new capabilities and increase the IoT functionalities.
Dominant players in the market of low power wireless SoCs and BLE particularly, have already released their first efforts towards mesh enabled BLE networks. Since the standardization of a mesh mechanism for BLE has not been finalized yet, each company is working on its own version of mesh networking over Bluetooth [**8**], [**9**], [**10**], [**11**].

2.3. Introduction to ZigBEE and Interfaces

ZigBee is an open standard wireless protocol, build on top of IEEE 802.15.4, developed by ZigBee Alliance. ZigBee is particularly targeted at low-power, low-cost and low data rate wireless sensors and control networks. Aimed at interoperability, it emphasizes on low complexity implementation and can support connectivity of up to 65,000 nodes. ZigBee operates at three different frequency bands including 868MHz, 915MHz and 2.4GHz. Using the most popular frequency band, at 2.4GHz, ZigBee nodes can communicate within a distance of up to 100 meters, with a maximum throughput of 250Kbps.

A ZigBee node that performs all the tasks defined by ZigBee standard, is called a Full-Function Device (FFD). In contrast to an FFD, a Reduce-Function Device (RFD) is a ZigBee node with limited tasks, which can only connect to an FFD. With respect to these functionalities, ZigBee devices can be classified in three types, namely Coordinator, Router and End Devices. A ZigBee Coordinator is a Full-Function Device, and a ZigBee network can contain only one. Its main responsibilities are the initial setup of the ZigBee network as well as the overall management of it. Main functions of the ZigBee Coordinator include address allocation, granting permission to nodes to join or leave the network and transfer application data. Due to the criticality of this node on a ZigBee network, it must always be powered on. A ZigBee Router is another Full-Function Device. A ZigBee network can contain none, one or more routers, depending on the size and the topology of the network. Its main functionality is to expand a ZigBee network. More specifically, it performs all the functions of the ZigBee Coordinator except network establishment. Constant power source must also be provided for a ZigBee Router.

A ZigBee End Device is a Reduce-Function Device, which located at the edges of a network. Its main functionality is to transmit and receive data. In order to conserve resources, ZigBee End Devices turn off their radio when they are idle and this where ZigBee as wireless communication technology emphasizes on power conservation.

2.4. Bridging WSN and WiFi Interfaces in RADIO

The smart home is comprised of several devices with different communication protocols. Figure 3 shows all available devices within the smart home environment and their respective interconnections. The RADIO Robot is presented in more detail in Chapter 6 of this volume, but what is relevant to the discussion here is that the robot's software is integrated using the Robot OS (ROS) middleware, with WiFi used to physically connect ROS nodes executing on the robot with ROS nodes executing on an off-board Raspberry Pi. This off-board unit also integrates the ROS WiFi network with the ZWave network and the BLE network as follows:

- One module is simultaneously a node of the ROS middleware and a node of the MQTT middleware used by the BLE network, and copies messages between the two.
- One module is simultaneously a node of the ROS middleware and a client of the REST API to the ZWave network provided by the ZWave Gateway, and copies messages between the two.

It should be noted that there is no direct bridge between the two WSN networks. Such a bridge is not needed, as both sensor networks only need to communicate with the Main Controller (see Section 3.1 below), which is main information broker of the overall system.

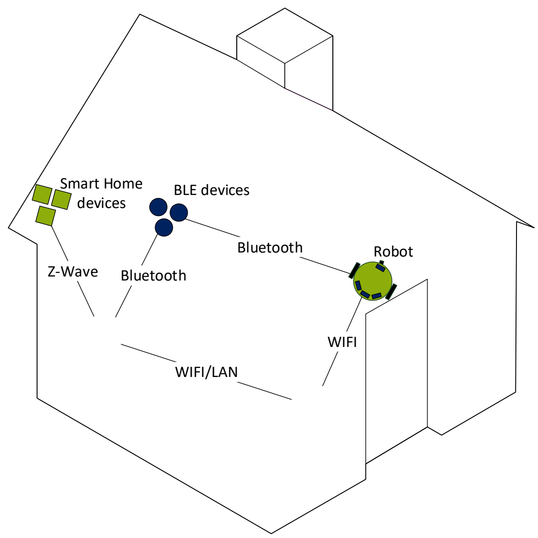

Figure 3: Device interconnection within the smart home environment

Since the robot should act autonomously and not be controlled remotely, it needs to be interconnected to the smart home environment. The robot is outfitted with two processing platforms: an Intel NUC and an Avnet PicoZed. The NUC is responsible for controlling the base platforms sensors and actuators. Therefore, it is directly connected to the TurtleBot2 base platform via USB. The Avnet PicoZed serves as accelerator platform to reduce the computation load of the Intel NUC. Additional devices that are placed on the robot are an Asus Xtion Pro camera and a Hokuyo laser scanner. These two devices can either be connected to the Intel NUC or to the Avnet PicoZed. The Intel NUC supports two wireless communication interfaces, the Bluetooth Low energy (BLE) and the WIFI interface. BLE connectivity is used for localization tasks performed by the robot, while the WIFI interface is required to connect the smart home environment to the IoT platform.

Initial measurements have shown that the usage the BLE and WIFI transceiver simultaenously results in degraded RSSI performance of the BLE transceiver. This is because the localization task requires accurate readings of the received signal strength indication values of the respective BLE devices. Because both interfaces are placed on the same network chip, the WIFI interface interferes with the BLE interface and vice versa. Additionally, both signals use the ISM band around 2.4GHz so that spectral overlapping of the signals can occur. This can results in strong signal interference. Exemplary measurements are shown in Figure 4 and Figure 5.

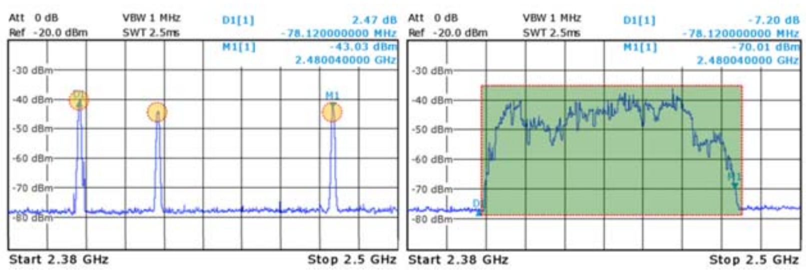

Figure 4 Spectrum BLE (left) and 2.4 GHz WIFI (right)

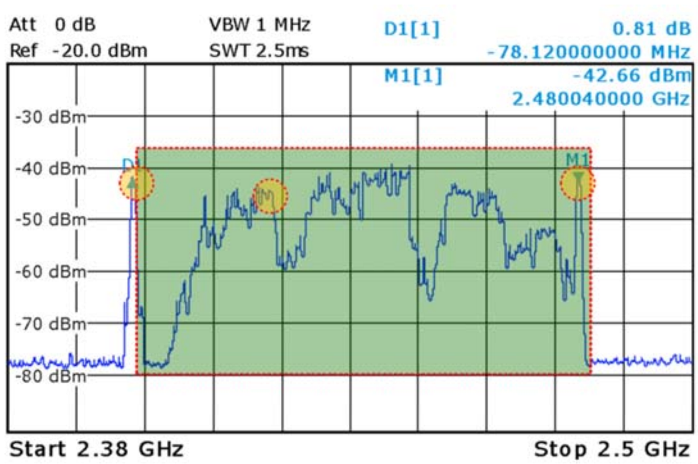

Figure 5 Spectrum BLE and WIFI

The Figures show the signal strength in dBm over the frequency in GHz. The left spectrum of Figure 4 depicts the spectrum of the BLE broadcaster. The three peaks show the frequency band of an advertising channel of the spectrum. They are highlighted in yellow. The mean value of the singal strength equals to -41.3 dBm. In contrast, the right spectrum of Figure 4 shows the spectrum over 2.4 GHz 802.11 b/g/n wireless LAN transceiver. Its region of influence starts at 2.395 GHz and ends at 2.484 GHz and is hihglighted in green. The mean signal strength equals to approximately -44 dBm. The combination of both signals is shown in Figure 5. Based on the measurements shown in the spectrums in Figure 4, an overlapping of the signals can be seen in the frequency band. Especially the advertising channel 38 is located in the WIFI spectrum. In order to reduce this interference, an external WIFI USB interface is used and connected to the Intel NUC.

3. RADIO Gateway Component

3.1. Main Controller

The RADIO Main Controller is the main orchestrator of the behaviours of the RADIO Home and the main keeper of the information collected and analysed by the various RADIO Home systems. Its functionalities include:

- System orchestration
- Bridging between the different sub-systems
- Storing and serving ADL recognition results

The Main Controller is (physically) partially distributed between the home computer and the robot computer, via a multi-master architecture: the core ROS process *(ROS Master)* executes on both the robot's on-board computer and on an off-board Raspberry Pi and messages are copying between the two integrate the two subsystems. This adds integration complexity compared to having ROS nodes (possibly remotely) connect to a single ROS Master, but it resolves the deadlock that:

- Many RADIO Home functionalities do not depend on the robot. If a single ROS Master executes on the robot, the Main Controller would be unable to operate with the robot turned off or having run out of battery.
- If a single ROS Master executes off the robot, the bandwidth-hungry communication channels between the sensors and the perception modules would have to use the Wifi.

An important module of the Main Controller if the *node manager* that orchestrates the overall system, including reacting to user initiatives through the user device and initiating automated actions, except for home automation directly handled by the S&C suite. Orchestration is implemented by sending control messages and by starting and stopping ROS nodes. The node manager also monitors ROS node execution to re-start nodes that have crashed. The node manager is distributed between the on-board and the off-board computer and is implemented as two ROS nodes. The main node is the one executing on the off-board computer, and it relegates to the on-board node the distribution of control messages for the system components that execute on the robot. Only the main node is required for the operation of the overall RADIO Home, so that functionalities not related to the robot remain active even if the robot is off-line or turned off.

3.2. Z-Wave GW

The Z-Wave gateway is responsible to manage the Z-Wave network and Z-Wave devices, while providing APIs to the controller application in order to create simple or complex systems through the combination of the different device functionalities that forms the network. The Z-Wave gateway initiates control commands and sends out the commands to devices and IoT platform. The controller has the capability to include/exclude nodes in the network and therefore always have the latest network topology. The controller device has a full routing table and is therefore able to communicate with all nodes in the Z-Wave network. The routing table is where the controller keeps the information from the nodes about the network topology. The routing table is built by the controller based on information received from all the nodes in the network, at installation time, about each nodes range.

In the case of Sensing & Control's smart home system, the system provides functions for security, comfort, home automation and energy management. The Z-Wave gateway provides three important functions:

1) Management of Z-WAVE network and devices
2) Basic pre-processing capabilities (mainly energy calculations)

3) Local information repository in order to deal with temporal internet cut offs (so historical information is stored and sent when internet connectivity is resumed).

In addition, the gateway has both computational resources and physical interfaces that can be used to connect third-party transceivers to enable connectivity via other communication protocols, such as WiFi or Bluetooth. The controller is composed by (i) a logical division of Java code named as Hardware Abstraction Layer (HAL) that bridges the gap between different hardware architectures and software by providing a uniform interface to the system peripherals, and (ii) a set of Java programs that provide different features, including a rules engine for setting 'if this, then that' style rules for how things interact (Figure 6) and a scheduler for enabling to set up the time that an action will be executed (Figure 7). For instance, you can trigger an action at a specific time (an hour of the day) or during a specific period (day, night or other intervals of time), and on given days. The user can also schedule an action to repeat itself several times.

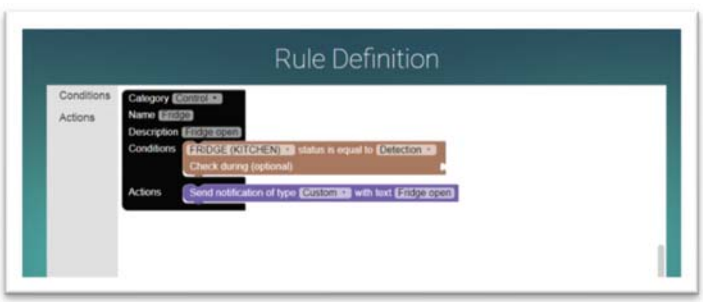

Figure 6. Setting a rule to receive a customized notification

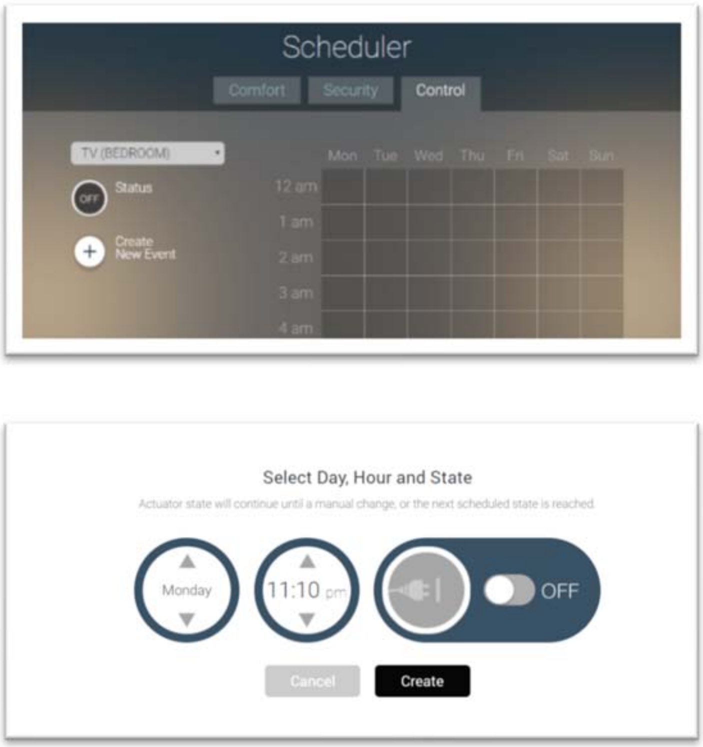

Figure 7. Scheduling an action on the TV device

Additional functions may also be integrated to run on the gateway if their resource consumption (or computational cost) is equal or less a predefined acceptable level, i.e. without compromising the overall performance of the RADIO solution. The growing number of IoT devices and dependency between them require a faster and smarter approach than the traditional one (gathering data and sending them through networks to the cloud to be processed). Fog computing architecture pushes the intelligence and processing power to the local area network level through the gateway. For RADIO, the gateway was delivered running on Raspberry Pi and Linux-enabled.

3.3. BLE GW

In this section the presented gateway, shown in Figure 8 aggregating data transmitted over Bluetooth Low Energy (BLE) wireless technology, but not limited only to BLE, in the context of the RADIO.

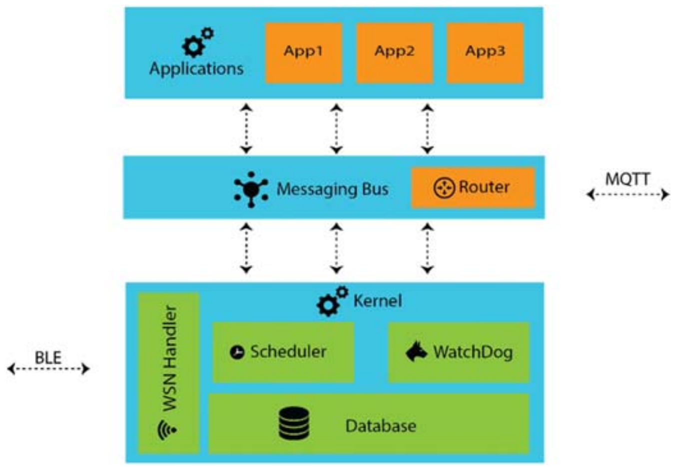

Figure 8. BLE Gateway High Level Architecture

The high-level design, of the Gateway architecture consists from three main components, namely the Kernel, Applications and the Messaging Bus. At the lower layer of the architecture lies the Kernel of the Gateway, where the core modules are deployed. On the contrary, at the top of the architecture, user-defined applications are commissioned, typically requiring a considerably higher degree of flexibility. Finally, in order to assure efficient interconnection between the two aforementioned components, a dedicated intermediate layer, the Messaging Bus, is introduced providing intra Gateway communication capabilities and functionalities.

BLE Gateway Kernel

The WSN Handler, shown in Figure 9, is the main aggregation point for operations associated with the Wireless Sensor Networks, and more specifically with BLE networks. Due to the fact that the communication between devices, in a BLE network, can be either connection-oriented or connection-less, the WSN Handler divided in three submodules, namely Advertisements Publisher, Advertisements Scanner and GATT Handler. The Advertisements Publisher and Scanner are handling the connection-less communication, where GATT Handler handle the connection-oriented communication.

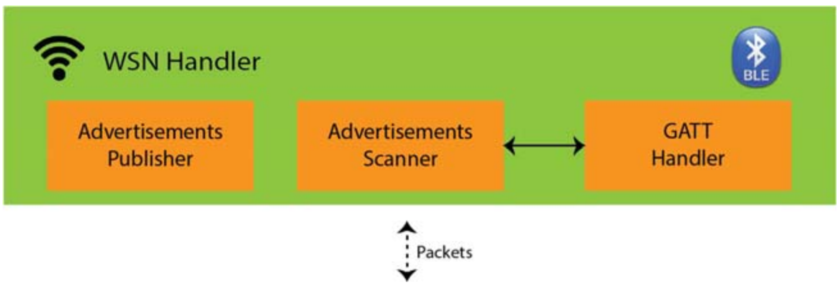

Figure 9. BLE Gateway WSN Handler

The main entry point for the BLE advertisement packets is the Advertisements Scanner, where every advertisement packet received on the BLE interface, is being considered. Next, according to the advertisement type, exported by the BLE PDU, the received packet forwarded to the appropriate module or discarded. There are many types of advertisement packets according to the Bluetooth Specification [**12**]**.** In the case of RADIO platform, Advertisements Scanner identifies solely the two following types of advertisements:

1. ADV_NONCONN_IND, Non-connectable undirected advertising. Used by devices that want to broadcast and don't want to be connected or scannable. This is the only option for a device that is only a transmitter.
2. ADV_DIRECT_IND, Connectable directed advertising. Directed advertising is used when a device needs to connect to another device.

When an *ADV_NONCONN_IND* advertisement packet received, the Advertisement Scanner forwards it to all interested parties. In contrast, when an *ADV_DIRECT_IND* advertisement packet is received, this packet is forwarded to the GATT Handler. The GATT Handler is responsible to handle all the lifecycle of a BLE Device, such as, Connection initialization, Data Collection and Connection Termination. Finally, the Advertisement Publisher is responsible to broadcast messages originated from the Gateway to the BLE network.

Database module, as the name implies, provides data persistence capabilities in the gateway. The Scheduler module, is a dynamic scheduling mechanism exposing functionalities for other modules and applications to execute simple or long running tasks.

Watchdog component, is the main monitoring tool of the Gateway. It monitors all hardware and software components of the Gateway and periodically sends respective data and system metrics to the Backend infrastructure. In this way administrators can effectively maintain the overall status of the deployed gateways.

Messaging Bus

The Messaging Bus, is the main aggregation point for all incoming and outgoing messages related to the Gateway. Internally, it offers local MQTT connections among the Gateway modules and applications. Externally, it maintains MQTT connections to the respective IoT Back-end infrastructure, in order for the Gateway to communicate with IP networks. Due to the existence of two different messaging systems, the internal Gateway messaging system (Modules, applications communication), and the external messaging system (Back-end IoT infrastructure communication), the need for a module was raised to bridge these two systems. To tackle this need, the Router plugin is added to the design, inside the Messaging Bus. The plugin performs topic-based routing by intercepting messages from topics, analyzing them and finally forwarding them, to the appropriate, external or internal MQTT topics, without modifying the message content.

User-Defined Applications

At the top level of the architecture, the Applications component, is responsible to dynamically deploy and manage all user-defined applications that run on the gateway. Applications resides outside the Gateway Kernel, and the communication is done mainly using the MQTT protocol.

4. Novel Services Based on BLE Capabilities

4.1. Introduction to BLE Multiloop Communication capabilities

The RADIO project identified the popularity that BLE attracts and foresees the benefits that modern IoT and Ambient Assisted Living (AAL) application domains will gain from the upcoming BLE mesh networking support. Therefore, RADIO designs a mesh mechanism that will be integrated in distributed sensor/actuator devices scattered across the RADIO AAL environment.

Network Formation

After a, thorough study of the existing approaches, the RADIO BLE mesh networking mechanism focused on the principles of BLE connectionless communications through advertising and scanning.
Advertising is the act of broadcasting data and it aims in device discovery and data publishing. The advertising mechanism involves 2 possible types of data packets that can be transmitted. The mandatory packet is the advertising packet while a node can optionally send a Scan Response packet. These two types of packets are structured mainly by the advertiser address and 31 bytes of payload as presented in Figure 10. During the normal operation, the BLE advertiser constantly broadcasts the advertising packets within an advertising interval bounded by a minimum and maximum value. These intervals typically may range from 20ms to 10.24s.

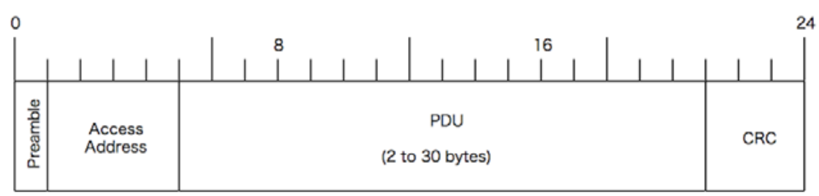

Figure 10. Advertisement/Scan Response packets' structure

During the network formation phase, the RADIO mesh mechanism utilizes the advertisement packets along with the broadcasting technique in order to achieve the network formation and route discovery. During the initialization of the network, each node broadcasts an advertising packet, which is the route discovery packet (route request - RREQ) in the context of the RADIO mesh. The RREQ messages are forwarded by each adjacent node until a pair of nodes exchange RREQs. Then a RREP message which contains the repliers address is sent backwards. During the reception of the RREPs, the respective neighbors table of each node is built. An instance of the neighbors table construction is presented in Figure 11.

This flooding approach replicates every message at every relay node and is expected to deliver near optimal delivery probability. Furthermore, it doesn't require any knowledge about the network during the design time. The increased resource consumption can be tolerated since flooding occurs once during the network initialization. Furthermore, the deployment scale of the RADIO mesh network is not expected to expand more than a number of nodes for a small to medium network size. Scalability is accomplished since the route discovery can be performed on a new node's entrance.

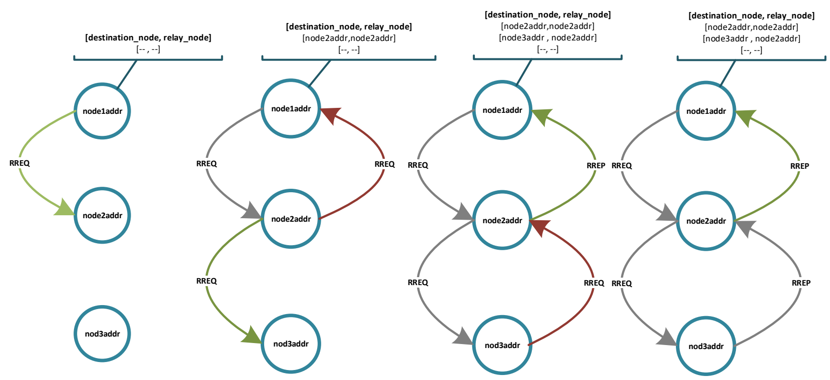

Figure 11. Network discovery mechanism

Message Forwarding

After the network discovery is completed by every node of the network every node has knowledge about its surrounding neighbors. The nodes now are ready to start forwarding their data messages. The approach followed in every version of BLE mesh released by the industry so far is on the connectionless mode of BLE and advertisements are used as data carriers.

The first approach of the RADIO mesh forwards data messages through rebroadcasting. Rebroadcasting works by flooding all messages to all nodes in the network through broadcasts. Nodes are in scanning mode and when an advertisement is received, the receiver rebroadcasts it to its neighbors (Figure 11). The process is repeated on every node and it is completed when every node receives the respective message. To avoid the broadcast storms through the continuously rebroadcasting of the same message, a versioning mechanism is implemented and runs on every receiver node. Upon the generation of the message on the source node, a version indicator is paired with the respective value and propagated along the network. Each relay node that receives the message stores it in a local data table. On every reception of a message from the same source and sensor, the relay node checks the version of the packet and propagates it to its neighbors if the received message version differs from the local. The message broadcasting process is completed when every node in the network received the message.

While broadcasting may increase the packet delivery ratio due to route redundancy it has a major drawback which highly affects the network performance under various network traffic loads and node density. Bluetooth Smart uses 40 RF channels in the ISM band (2.4GHz). These RF channels have center frequencies 2402 + k*2MHz where k ranges from 0 to 39. Advertising utilizes three of them specifically channel 37 (2402MHz), 38 (2426MHz) and 39 (2480MHz). Due to broadcasting, the network is flooded with redundant transmissions. Thus, the poor utilization of frequency bands may degrade network performance and overall power consumption.

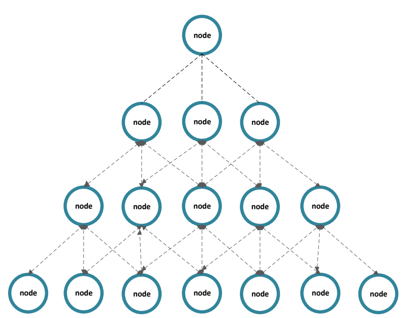

Figure 12. Message Forwarding with broadcasting

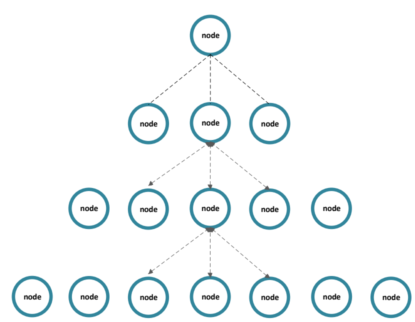

Figure 13. Message Forwarding with selective broadcasting

During the development and evaluation phase of the RADIO mesh a selective broadcasting policy was implemented. As described earlier,

during the network phase, the nodes discover their neighbors and they store this knowledge locally. This knowledge is used to build the respective forwarding mechanism depending on the destination of their transmissions. These routing tables are used by the RADIO mesh to perform the selective broadcasting mechanism. The 29 bytes of payload are used to encapsulate the address of the destination node, the sensor model (described by the data type and data value) and the message version (described later). The application domain where RADIO mesh is deployed handles simple numeric sensor values that do not demand large packet payloads to carry them through the network. Therefore, the 29 bytes of the BLE advertisement packet is considered adequate for similar applications.
During the operation of the RADIO mesh with the selective broadcast enabled, every parent node that receives a message by the child node parses the payload of the packet and checks the packet's destination node. If the destination node is registered in its routing table, the message is rebroadcasted to its neighbors. In case the relay node doesn't retrieve a route to the destination node it discards the packet from its queue. Figure 12 and give a visual representation of the traffic generated by the broadcasting and selective broadcasting transmissions. The figures show how the traffic load differentiates among the two approaches. Particularly we observe that selective broadcast relaxes the traffic load significantly, while at the same time retains a degree of route redundancy that benefits the packet delivery ratio without abusing the network resources.

4.2. Indoor Localization Based on BLE Beacons

The following subsections outline a set of functionalities that must be efficiently executed on the robot platform focusing on the navigation functionality [18].

One fundamental problem in robotics is the simultaneous localization and mapping (SLAM), also known as Concurrent Mapping and Localization (CML) [**13**]. It arises when neither a map of the surrounding, nor the actual position of the robot is known. Several algorithms such as the SLAM with Extended Kalman Filters (EKF) and the SLAM with Particle Filters address this issue. The robot simultaneously creates a map of the environment and localizes itself relatively to this map. Within the RADIO project, a SLAM algorithm is implemented enabling self-localization estimation of the robot within the smart home. Based on the results of the SLAM algorithm, the robot is aware of its own position and can further use this information to navigate autonomously through the environment. This objective creates new challenges for man-machine interaction, as the robot should never become an obstacle for the human. Additionally, the robot is required to autonomously start the mapping process in “out-of-the-box” mode, meaning that it should detect by itself that its surroundings are unknown. When this detection is completed, the robot should start the mapping process. When autonomously mapping the surroundings, the robot will follow a contour (e.g. a wall) and chart the unknown environment with the data of the laser scanner and of its odometric sensors. Figure 15 shows the results of the mapping process. Figure 14 shows the robots mapping process.

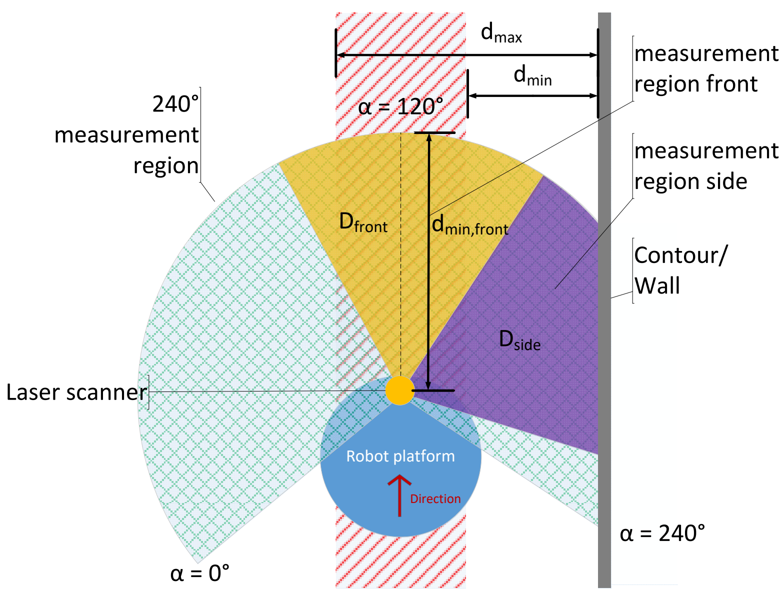

Figure 14 Schematic process of following the contours.

The robot platform will periodically measure the distances *d_i* between itself and the contour. The angle α represents the measurement angle corresponding to the measured distance. Based on α, the distances *d_i* are assigned to the region D_{front}, D_{side}, or are ignored. Figure 14 depicts D_{front} as a yellow and D_{side} as a purple region. If all $d_i \in D_{side}$ $\mathrm{d_i} \in \mathrm{D_{side}}$ lie in the interval $[d_{min}, d_{max}]$ $[\mathrm{d_{min}}, \mathrm{d_{max}}]$ and all $\mathrm{d_i} \in \mathrm{D_{front}}$ are larger than $\mathrm{d_{max}}$, the robot performs a linear motion until the next measurement is initiated. The interval serves as hysteresis function to avoid the oscillation around the boundary limits d_{max} and d_{min}. If some $\mathrm{d_i} \in \mathrm{D_{side}}$ do not lie in the interval, the two cases $\mathrm{d_i} \leq \mathrm{d_{min}}$ and $d_i \geq d_{min}$ $\mathrm{d_i} \geq \mathrm{d_{max}}$ have to be checked. In the case of $\mathrm{d_i} \leq \mathrm{d_{min}}$ the robot has to turn left and in the case of $\mathrm{d_i} \geq \mathrm{d_{max}}$ the robot has to turn right. This action is performed by changing the angular velocity ω. Additionally, if at least one $\mathrm{d_i} \in \mathrm{D_{front}}$ is smaller than $\mathrm{d_{min,front}}$, the robot performs a rotation of ω>0 in order to avoid a frontal collision. This case generally occurs in corners of the room. After a finite amount of time, based on the size of the room, the robot created a map and is now able to localize itself within the now known environment.

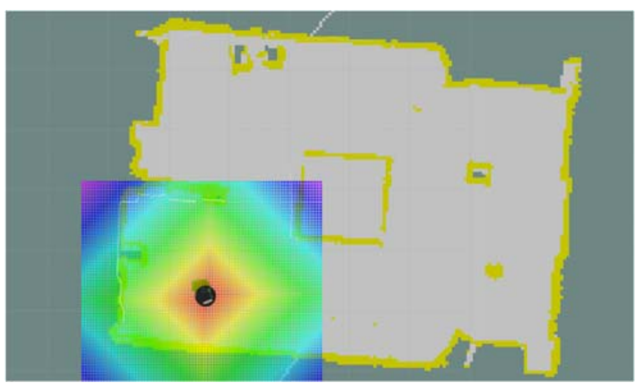

Figure 15 Mapped exemplary room and estimated position of the robot.

Localization is very important for the cognition task since it provides the most relevant data regarding current distances to obstacles for safe navigation through the environment. The localization task depends on the data from the perception task. For localization within the SLAM algorithm, a particle filter can be chosen since it easily incorporates information from different sensor types, thus enabling sensor data fusion.

With additional sensors, the robot is also able to localize other objects. This can be accomplished with the Bluetooth devices that are positioned within the house. Thus, the robot can also help the end-user to search for an item which is outfitted with a BLE transmitter. Because Bluetooth beacons only broadcast their ID, the received signal strength indication (RSSI) is used in order to extract position information out of the beacons signal. The RSSI value can then be converted into a distance measurement. With one beacon, a distance measurement can only define a region of interest in which the receiver is currently located. The BLE receiver can extract a distance to the BLE beacons by solving the free space loss equation. In general the free space loss of a signal is described by

$$F = \frac{P_{Rx}}{P_{Tx}} = \left(\frac{\lambda}{4 \cdot \pi \cdot r} \right)^2 \text{, with } \lambda = \frac{c}{f}. \tag{1}$$

P_{Rx} indicates the signal power received by the receiver and P_{Tx} indicates the signal power sent by the sender. *r* represents the distance between sender and receiver and *f* is the signals frequency which is 2.4 GHz for Bluetooth. According to equation (1), the received signal strength is reduced by $\sim \frac{1}{r^2}$ with increasing distance from the sender. However, indoor environments vary greatly in terms of floor plan and equipment, thus influencing the signal strength through signal reflection, refraction, or interference with other signals. In order to incorporate these factors into the distance measurement, a logarithmic distance loss model is used. This model adds the variable γ which describes the signal loss in dependence on the surroundings [**14**].

$$\frac{P_{Rx}}{dBm} = \frac{P_0}{dBm} - 10 \cdot \gamma \cdot \log_{10}\left(\frac{r}{r_0} \right) \text{, with } \gamma \in R \tag{2}$$

Here, P_0 corresponds to the received signal strength to the corresponding distance r_0. The value for γ is determined through several test measurements with known distances. However, when using equation (2) for distance calculation, the resulting distances are very noisy and inaccurate. Therefore, converting the RSSI values into distances requires exhaustive measurements in the respective indoor environment because the absorption of the signal varies greatly depending on the surroundings.

Thus, the relation between RSSI values and the corresponding distance for our specific environment is effectively approximated. Because the relation of RSSI to distance is dependent on the surroundings, we performed measurements in a hallway which is 2m wide and 50m long and in a room with $4m \times 5m$ floor space. Figure 16 shows the relation of RSSI and distance and the corresponding floorplan of the hallway.

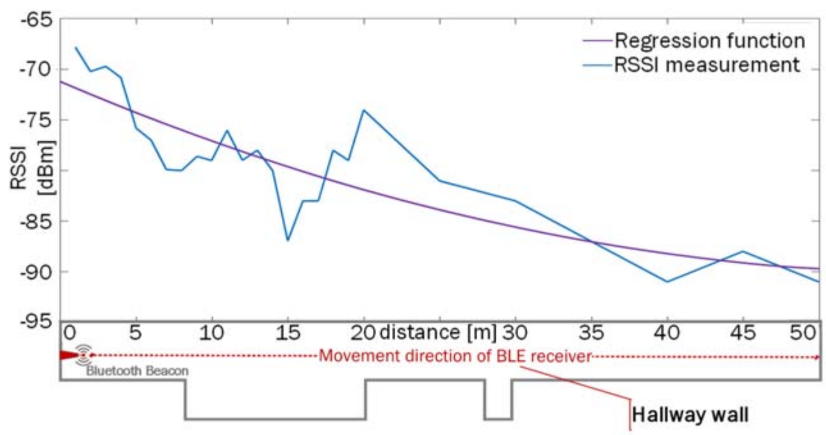

Figure 16 RSSI to distance relation of in the hallway

A beacon was positioned at the start of the hallway in 1.5m height; see Figure 16 the red icon on the left. Starting from the beacons position, signal strength measurements are conducted in 1-meter intervals until the end of the hallway is reached. At every measurement position, 40 individual measurements are acquired, and the mean value is calculated. All measured mean RSSI values are shown by the blue line in Figure 16.

Between 8m and 20m, a larger width of the hallway can be seen. In this region, the RSSI values do not correlate well to their respective distance. We can see a decrease of the received signal power from -79dBm to -87dBm. This occurrence leads to the assumption that in this region more reflections and refractions influence the signal strength. In the region after 20m distance, the signal power increases again to -73dBm. As can be seen, no direct relation between the RSSI values and the corresponding distance can be established. The RSSI values do not even decrease monotonously with increasing distance. Therefore, we determine a regression function which fits the measurements in a satisfactory manner. The regression function is depicted in equation (3)

$$r(x_{rssi}) = 0.07472 \cdot x_{rssi}^2 + 10.106 \cdot x_{rssi} + 345.21 \cdot \quad (3)$$

The measurements in the $4m \times 5m$ room show a different behavior of the signal strength in relation to the respective distance. Figure 17 shows the RSSI to distance relation of the $4m \times 5m$ room.

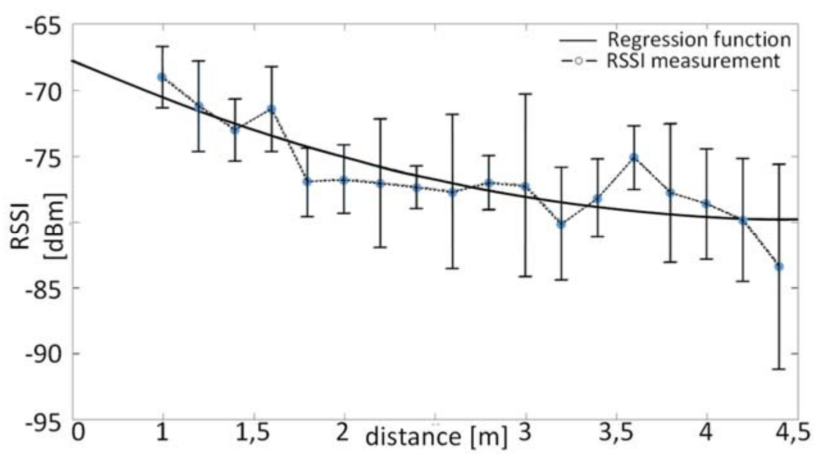

Figure 17: RSSI to distance relation of the 4m x 5m room

For each position, 200 measurements were performed. The deviation of an individual measurement can be up to 11% from the mean value over all 200 measurements. The standard deviation of the respective measurement point is depicted in Figure 17 as bar plot. The regression function which fits the measurements best is shown in equation (4).

$$r(x_{rssi}) = 0.0010332 \cdot x_{rssi}^3 0.24022 \cdot x_{rssi}^2 + 18.32 \cdot x_{rssi} + 460.89 \tag{4}$$

With no further information, the distance *r* describes a sphere in 3D space, see equation (5).

$$(x - x_m)^2 + (y - y_m)^2 + (z - z_m)^2 = r^2, \tag{5}$$

with $(x_m, y_m z_m)$ being the BLE receivers position. This 3D problem can be reduced to a 2D problem, by assuming that the height z_B of the beacons is known. Thus, the sphere can be reduced to a circle. The circle has the center at $(x_m, y_m z_B)$ and the resulting radius can be calculated as

$$r_C = \sqrt{r^2 - (z_m - z_B)^2}. \tag{6}$$

Thus we can describe every possible position of a BLE beacon through

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x_m + r_C \cos\varphi \\ y_m + r_C \cos\varphi \end{bmatrix} \text{ with } 0 \le \varphi \le 2\pi. \tag{7}$$

Through thresholding of the beacons measured distance, simple annotation of regions within an indoor environment is possible. This approach is shown in Figure 18.

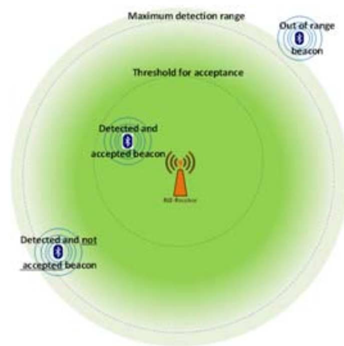

Figure 18: Thresholding of range determined through the RSSI values of BLE beacons

The main problem with this approach is that the RSSI values are very noisy. Because of the RSSI noisy properties, a single RSSI measurement does not yield accurate and therefore valuable information. Therefore, some manner of prefiltering has to be executed before the distance conversion. As a prefilter, we use a 10th order finite impulse response (FIR) filter. The FIR filter equation is shown in equation (8).

$$y_n = \frac{1}{a_0}\left\{\sum_{k=0}^{N} b_k \cdot x_{n-k}\right\} \tag{8}$$

a_0 and b_k describe the filter coefficients and were determined with the help of the Filter Design & Analysis Tool of Matlab. The filter coefficients are depicted in TABLE I. .

TABLE I. FIR FILTER COEFFICIENTS

Filter coefficients	Value	Filter coefficients	Value
a_0	1	b_5	0.3071
b_0	-0.0349	b_6	0.2562
b_1	-0.0370	b_7	0.1368
b_2	0.0209	b_8	0.02092
b_3	0.1368	b_9	-0.0370
b_4	0.2562	b_{10}	-0.0349

To show the increase in accuracy through the above filter, we compare the raw RSSI measurements with the filtered RSSI measurements. A beacon is positioned with 1.2m distance to the BLE receiver. Figure 19 shows the different results of raw RSSI and filtered RSSI measurements.

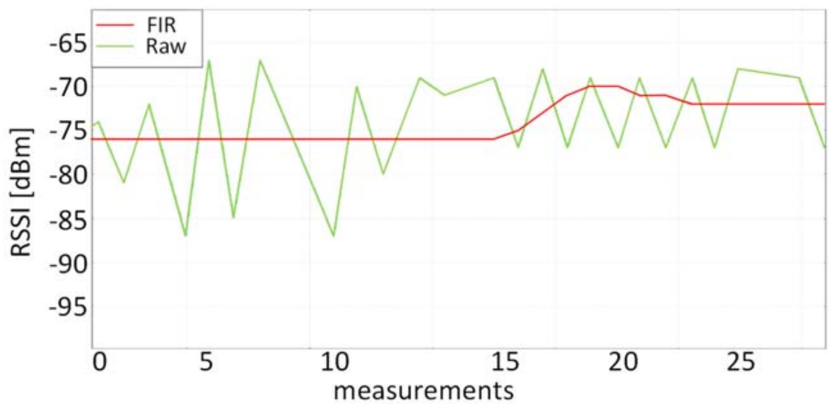

Figure 19 Comparison of raw RSSI data and FIR filtered RSSI data

It can be clearly seen that the filtered RSSI measurements are more stable than the raw RSSI measurements. To further evaluate the performance of the filtered RSSI measurement approach, we perform RSSI measurements in a radius of 1.2m distance of the BLE beacon, see Figure 20. 150 measurements were acquired on the circle in order to examine the behavior of the RSSI values when the transceiver has the same distance to the BLE beacon but a different orientation.

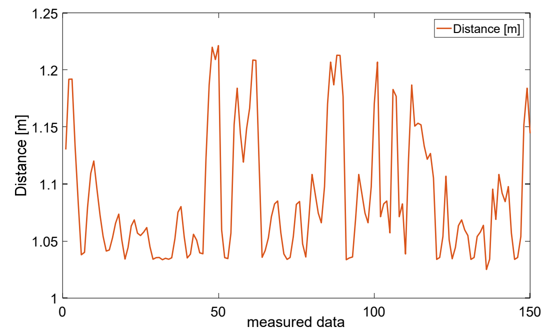

Figure 20: Calculated distance out of filtered RSSI measurements. 150 measurements were performed in a circle with 1.2m radius distance to the BLE beacon.

When calculating the distance out of the filtered RSSI values with equation (4), the mean value for the calculated distance is approximately 1.1m with a standard deviation of 0.054m. This results in a nominal error to the true distance of 10cm or 8.3%. Since the uncertainty of the calculated distances is in the range of single digit centimeters, accurate position estimation with several RSSI measurements can be performed.

For accurate position estimation through trilateration, three measurements at different positions have to be performed. Each position generates a circle with a certain radius which describes the measured distance to the BLE beacon. In the case of three measurement positions, 27 circle constellations exist. These 27 constellations contain redundant constellations which do not need to be analyzed separately. Then, these 27 constellations can be reduced to 10 different circle arrangements. We assume that no circle is contained within another circle except if several measurement errors occur. Therefore, we discard the circle arrangements where circles are included in other circles. Then, we receive four valid arrangements as seen in Figure 21. They are used to determine the region of interest (ROI) with the help of multi-lateration [**15**].

In the first arrangement, each circle has no intersection with the other circles. In this case, the center of the shortest connection for every circle pair is determined. These three calculated points are the triangle out of which the position estimation can be calculated for the BLE beacon. The region of interest is the circumcircle of the triangle with the radius being the uncertainty of the measurement.

In the second arrangement, exactly two circles intersect with each other. Then two intersection points exist. One of the intersection points has a larger distance to the third circle than the other intersection point. The intersection point with the larger distance is discarded and the resulting region of interest is a circle with the center point being the center of the shortest connection between the remaining intersection point and the third circle while the radius of the circle and thus the uncertainty is half the length of the connection.

In the third arrangement, one circle serves as connector to the two other circles. The two other circles do not share any intersection while the circle in the middle has two intersections with the respective circle. In order to determine the region of interest, all intersection points have to be calculated and the four distances between the intersection point pairs from the different circles have to be determined. The intersection point pair with the smallest distance is chosen and the center of this connection is the center point of the region of interest circle with the radius being half the length of the connection.

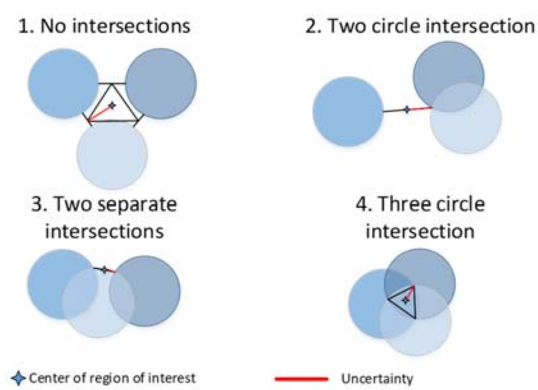

Figure 21: Four possible circle arrangements

In the fourth arrangement, all circles intersect with each other. The distance to the remaining circle is calculated for each intersection point. The intersection point with smallest distance to the remaining circle is chosen as final point for the resulting triangle. Just as in the first case, our region of interest is formed by the circumcircle of the resulting triangle.

If more than three measurements are performed, this approach is also valid. Then we consider four triples of circles (1, 2, 3), (1, 2, 4), (1, 3, 4), and (2, 3, 4) separately. The triple with the smallest measurement error determines then the final region of interest.

5. Backend Infrastructures

5.1. EnConrtol

The smart home architecture in RADIO is based on a white label B2B product by Sensing & Control Systems called *enControl™*. *enControl*™ provided an initial solution for home automation, and was enhanced and upgraded complementing other technologies and developments within RADIO in order to build the final RADIO solution. The main four functionalities[1] of *enControl™* are:

1) Comfort
   a. Climate monitoring
   b. Climate control
   c. Temperature, Humidity, CO2 (etc…) levels
2) Security
   a. Detection of door/window opening
   b. Detection of movement
   c. Detection of Smoke
   d. Detection of CO
   e. Detection of water basement
3) Energy Management
   a. Energy consumption
   b. Energy control (switch on/off electricity, water, gas, etc…)

[1] The list does not pretend to cover all possibilities, the Reader should understand that the smart home solution can integrate any ZWAVE standard product, thus enabling the functionality delivered by a particular product. For full list of product, please visit ZWAVE alliance web page.

4) Automation
    a. Switch on/off appliances
    b. Switch on/off lights
    c. Open/Close doors, curtains, shutters

Smart home back-end is divided into two main components, (i) home devices, and (ii) IoT Platform. The home devices component includes sensors, actuators and the home controller. The home controller is a product from Sensing & Control that complements the smart home functions delivered by the IoT platform. The IoT platform contains the core of the smart home solution. It provides an open REST API enabling the home controllers to exchange information bi-directionally, based on synchronous or asynchronous actions triggered by IoT and/or end users through *enControl™* interfaces. Also, the IoT platform acts as an information repository, storing both historical data about sensors (values and status) and actions triggered by users, enabling to know what action was executed, by who, and when. For example, “the TV set has been switched on by Maria on 11 July 2015, 20:30 CET”.

The IoT platform is able to connect, transport, process, analyse, and manage data from the sensors to the real world to high-level applications and vice versa. It is able to operate in both wireless/wired network environments and supports different communication protocols. As a function summary:

- **Collect data**

Collect data refers to the ability to retrieve and store information for further exploitation about all the parameters that are relevant to the system: messages, device status, commands, errors, exceptions.

- **Analyse data**

The IoT Platform has processes in charge of analysing data and taking decisions based on the quality/importance/integrity of this data.

- **Data Aggregation/Data Fusion**

Data aggregation refers to the ability to concatenate info from devices, for example when there is a mix of information between data from mash-up sensors.

Data fusion, is the process of integration of multiple data and knowledge representing the same real-world object into a consistent, accurate, and useful representation

- **Translate data**

This feature is related to the ability to transform raw data in bytes from the devices to high-level information.

- **System monitor**

The IoT Platform is able to monitor the communication between devices and high level apps; and the overall status of the platform and device network. For instance, it is used to monitor key performance indicators like the number of messages sent per minute, number of exceptions/errors, number of devices connected to the platform, etc.

- **Transfer**

The platform provides mechanisms to publish subscribe information to the queues exposed by the SDK

- **Audit information**

In order to enable traceability of the functionality of all sub-systems, active logging is implemented within the software components. Log files are stored and available for analysis.

- **Secure components and communications**

Communication between components and devices are secured

The current solution of *enControl™* provides means of interaction with home devices by accessing dedicated functions of the open API of the IoT platform. This implies that other ICT solutions (either at home like other RADIO components or remotely) willing to interact with *enControl™* must have internet connection.

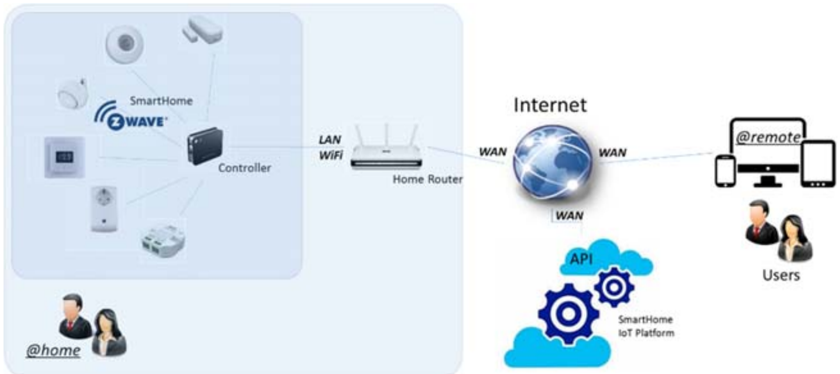

Figure 22 enControl™ Service Architecture

The API is divided into following main groups:

1) Authorisation and Authentication
    a. Functions used to grant access to different IoT resources and API functions
2) Devices
    a. Functions related to push and retrieve device status and data (including historical data)
3) Users
    a. Functions related to the management of users
4) Installations
    a. Functions related to the management of smart home installations
5) Monitor
    a. Functions related to the status monitoring of the IoT

The open REST API is used by user interfaces through web clients and smartphone apps in order to present to end users the information being acquired from home end devices, and the action than can be triggered to them, so it encapsulates the smart home functions offered.

It is important to notice that *enControl™* clients (smart phone apps and Web interfaces, 3rd party services, etc.) interact with the smart home through the API, making internet connections required. It should be noted that experience with deployment of the solution is that this introduces no perceptible delay in executing actions by Z-Wave devices.

5.2. ATLAS Presentation

In this section, a conceptual analysis of the ATLAS IoT platform as the technician's backend platform in the context of RADIO project is presented. Starting from the left-hand side of Figure 23, an efficient, flexible and extendible approach is targeted so that any kind of heterogeneous sensor or actuation modality is aggregated at the ATLAS Gateway. In order to support the heterogeneity posed by different wireless communication technologies, the MQTT-SN [**16**] protocol, has been selected, as a prominent communication protocol.

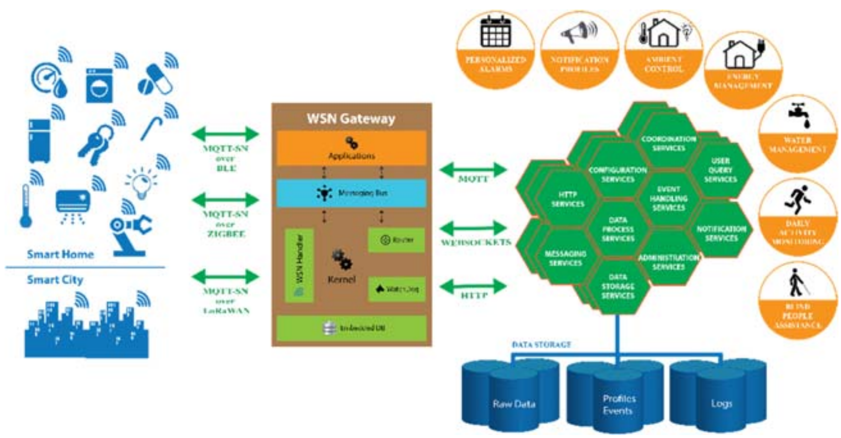

Figure 23. ATLAS IoT Platform.

MQTT-SN is a publish/subscribe messaging protocol that extends the well-established Message Queue Telemetry Transport (MQTT) [**17**] protocol in order to cope with the specific constraints of WSNs, such as resource-limited and battery-operated devices, low network bandwidth and high link failures. Another critical conceptual choice, designing such IoT platforms, concerns the way data, events or/and commands can be exchanged between the Gateway and the backend infrastructure. In that respect, ATLAS IoT Platform, besides the well-known HTTP protocol, expose additional communication interfaces, such as MQTT and WebSockets, in order to consume real time data from multiple sources, from both indoor and outdoor environments.

In the core of ATLAS platform, the deployed services are based on the micro-services architectural pattern, which allows to deploy small autonomous applications, that states on the single responsibility principle. Using this architectural pattern, ATLAS services can be deployed and scaled independently in the ATLAS Cloud Infrastructure, without affecting the overall performance of the platform.

Finally, given the growing needs for data-driven features and the complexity of modern IoT applications, it is important to offer services able to share and store data in an isolated and scalable way. In that respect, at the bottom level of the architecture, ATLAS IoT Platform deploys a heterogeneous data storage system, consisting from multiple different storage types, such as SQL, NoSQL, Key-Value stores etc., to address the diversity of data. On top of the deployed databases, ATLAS IoT Platform implements an abstraction layer that provides simplified data access using simple APIs to hide completely the complexity posed by database heterogeneity and low-level mechanisms for data manipulation.

6. Summary

This chapter aims to highlight multifaceted experience and knowledge gained in the context of RADIO project, from the efforts devoted to offer a reliable, efficient and versatile end-to-end communication infrastructure. Therefore, initially the conceptual architecture is clearly presented which, in conjunction with the application scenarios, effectively drove the identification of adequate requirements that the end solution should meet. As indicated, such requirements span over a wide range of challenging and diverse demands concerning both the Wireless Sensor Network domain but also the backend communication infrastructure.

Then, the main ultra-low power wireless communication technologies selected, supported and integrated by the RADIO communication platform are presented highlighting critical characteristics and features for the RADIO communication objectives. One of the main characteristics of the RADIO communication platform is efficient and seamless support of heterogenous communication protocols. Towards this goal in the context of the RADIO concept the Gateway device is of cornerstone importance since it comprises the entity that effectively supports all different communication paradigm and offers a unified way of data

transfer from and to the sensors. Therefore, critical aspects of this compound component are presented in detail in section 3.
However, in RADIO significant effort is devoted on offer extended and enhanced features and functionalities related to the adopted communication technologies. Therefore, in section 4, two such endeavors are indicated. On one hand, attempt to extend the BLE protocol to support multi-hop communication is presented while, on the other hand, indoor localization techniques based on BLE sensors are analyzed.
Last but not least, the backend communication infrastructure technologies adopted and exploited in RADIO project, also comprising critical cornerstones to meet the overall project's objectives, are presented in section 5.

7. References

[1] Giannakopoulos, T., Konstantopoulos, S., Siantikos, G., and Karkaletsis, V., "Design for a System of Multimodal Interconnected ADL Recognition Services". Components and Services for IoT Platforms: Paving the Way for IoT Standards, Springer, 2016. Reproduced as Chapter 2 of this volume.

[2] On Developing a Novel Versatile Framework for Heterogeneous Home Monitoring WSN networks, http://www.ingentaconnect.com/content/doaj/24117145/2015/00000001/00000001/art00005

[3] http://ec.europa.eu/information_society/newsroom/cf/dae/ document.cfm?action=display&doc_id=7681

[4] http://www.cloud-council.org/deliverables/CSCC-Cloud-Customer-Architecture-for-IoT.pdf

[5] http://www.shimmersensing.com/

[6] https://www.movisens.com/en/

[7] Revoking Networks "Advanced User Manual" Version 4.77 2/3/2011

[8] https://wiki.csr.com/wiki/CSRmesh

[9] https://github.com/NordicSemiconductor/nRF51-ble-bcast-mesh

[10] Levis, Philip, et al. The trickle algorithm. No. RFC 6206. 2011

[11] https://github.com/openthread/openthread

[12] Bluetooth SIG (Hrsg.): Specification of the Bluetooth System: Covered Core Package version:4.0, June 2010

[13] Thrun, S., Burgard, W., Fox, D. "Probabilistic Robotics", The MIT Press, ISBN 978-0-262-20162-9

[14] Grossmann, U., Schauch, M., Hakobyan, S. (2007) — The accuracy of algorithms for WLAN indoor positioning and the standardization of signal reception for different mobile devices, International Journal of Computing, Vol. 6, Issue 1, pp 103-109.

[15] Kuruoglu, G.S., Erol, M., Oktug S. (2009) — Localization in Wireless Sensor Networks with Range Measurement Errors, Fifth Advanced International Conference on Telecommunications, Istanbul.

[16] Andy Stan-ford-Clark and Hong Linh Truong – MQTT For Sensor Networks (MQTT-SN) Protocol Specification, Version 1.2, http://mqtt.org/new/wp-content/uploads/2009/06/MQTT-SN_spec_v1.2.pdf, November 2013

[17] MQTT Protocol Specification. Oct 29, 2014, http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/os/mqtt-v3.1.1-os.pdf

[18] Fynn Schwiegelshohn, Philipp Wehner, Florian Werner, Diana Göhringer, Michael Hübner, "Enabling In-door Object Localization through Bluetooth Beacons on the RADIO Robot Platform" In Proc. of the International Conference on Embedded Computer Systems: Architectures, Modeling and Simulation (SAMOS XV), pp. 1-6, Samos, Greece, 2016.