Architecting Evolving Internet of Things Application

Swaytha Sasidharan

International Doctorate School in Information and Communication Technologies
University of Trento
June 24,2019

Supervisors:
Prof. Fabrizio Granelli
Raffaele Giafreda

Abstract

The Internet of Things paradigm has witnessed an unprecedented growth pervading every sector of the societal fabric from homes, cars, health, industry, and business. Given the technological advances witnessed in all the enabling technologies of IoT, it is now possible to connect an increasing number of devices to the internet. The data value chain has slowly risen in prominence. Moving from the vertical solutions to a horizontal architecture has resulted in the development of services which is the culmination of multiple sources of data. The complexity of handling the growing number of connected devices has resulted in active research of architectures and platforms which enable the service providers and data providers to navigate through the maze of technologies. We also look at the data generated by the real world, which is dynamic and non-stationary in nature. The always connected virtual representations of the devices facilitates applications to proactively perceive, comprehend and adapt to the real world situations. Paving the way to integrate learning algorithms, this thesis presents a modular architecture with elements to detect, respond and adapt to the changing data. Given the scope of IoT in different applications, we explore the implementation challenges both the advantages and limitations in two different domains. It includes (i) Smart Asset Management Framework: To provide real time localization of movable medical objects in a hospital. Additionally the movement patterns of the objects in studied and modeled to facilitate predictions. This helps to improve the energy savings of the localization technology. It helps the hospital authorities to understand the usage of the objects for efficient resource planning. (ii)Transitioning to a Industrial4.0 application: To facilitate in the digital transformation of a solar cell research center. With the similar concepts of virtualization (digital twins) and real world knowledge generation, the digital factory vision is conceptualized and implemented in phases.The supporting work including prototypes for smart home environment control, people activity detection and presence detection using bluetooth beacons leading to the development of the architectural components is presented. The implementation details along with results and observations in each of the sections is presented.
Keywords: Internet of Things,IoT Architecture, Real World Knowledge, Industry4.0

Acknowledgements

The thesis is the culmination of a long and steep learning journey. I would like to thank the University of Trento,Italy for providing me with the opportunity to conduct my PhD studies and providing the necessary guidance and opening up different perspectives to research methodologies. The course structure provided me the opportunity to look beyond the defined work structure and have an open mind to looking at problems from different perspectives. I would like to thank Create-Net,Trento,Italy who provided me with the opportunity to carry out an industrial PhD. Moving beyond a pure theoretical research and creating opportunities to implement the cutting edge technologies to contribute to the betterment of people and life is a lesson that I take forth in my professional career. Next I would like to thank my advisor, Mr.Raffaele Giaffreda, who has been a constant source of motivation throughout my research period. It was under his tutelage, through discussions and insights encouraged me to continuously evolve and improve on the work. My advisor, Prof. Fabrizio Granelli, for having supported my thesis and providing me the opportunity to present the work. The Project iCore and the discussions with all the partners provided me with a very global view of the advantages and challenges working in collaborative environments. On the technical front, it laid the foundation for an aptitude to continuously learn new softwares and skills. I would like to thank International Solar Energy Research Center(ISC), Konstanz,Germany for having provided me the chance to work in an challenging field as the Industry4.0. I thank Mr.Rudolf Harney for his guidance charting through a new domain. Additionally I thank Bosch IoT Lab and University of St.Gallen, Switzerland for the internship and the lessons learnt in addressing different applications in the smart home and smart city domains. Throughout the PhD journey I witnessed several personal and professional upheavals which has all defined and molded me as an individual. Finally I thank my husband,Lejo who is my source of strength and encouragement and my son, who brings in the sunshine everyday. My parents and in-laws whose wishes and blessings has brought me to this pedestal. More family than friends, Tinku and Sajna for their support personally and professionally. A very supportive friends circle and family who I will forever be grateful for their continued well wishes.

Swaytha Sasidharan

Contents

List of Figures

List of Tables

List of Abbreviations

IoT	**I**nternet **o**f **T**hings
VO	**V**irtual **O**bject
CVO	**C**omposite **V**irtual **O**bject
SL	**S**ervice **L**evel
SR	**S**ervice **R**equest
ST	**S**ervice **T**emplates
RWK	**R**eal **W**orld **K**nowledge
REST	**R**epresentational **S**tate **T**ransfer
MQTT	**M**essaging **Q**ueing **T**elemetry **T**ransport
BPMN	**S**ervice **L**evel
XML	**S**ervice **L**evel
HMM	**H**idden **M**arkov **M**odel
CRF	**C**onditional **R**andom **F**ields
SECSGEM	**S**emi **E**quipment **C**ommunications **S**tandard **G**eneric **E**quipment **M**odel
OPCUA	**O**PC **U**nified **A**rchitecture

Chapter 1

Introduction

1.1 Internet of Things

With the increasing prevalence of digital and electronic devices, the technological advances interconnecting these devices has been unprecedented. This exponential upscaling of devices leads us to the Internet of Things (IoT). The term IoT was first coined by Ashton demonstrating an application based on Radio Frequency Identification (RFID) tags [3]. IoT, represented in Figure 1.1, is the technological infrastructure that links the physical world objects to the internet [36], consisting of a network of physical objects that can sense, communicate and interact with their environment. The advances in Wireless Sensor Networks (WSN) and the emergence of low power communication protocols, made the end-end connectivity with devices a reality. The connected devices has been established due to the advancements in lowpower network protocols such as Zigbee, Bluetooth, IEEE 802.15.4 or low-power WiFi and 6LoWPAN [27] [30]. It moves on to create ecosystems with various services and applications in diverse domains such as smart homes and cities, health and industries. Since the introduction to the Gartners Hype-Cycle of emerging Technologies in 2011, IoT has been continually featured in the consequent years with the 2014 chart showing it reaching the peak of inflated expectations.

FIGURE 1.1: Internet of Things [28]

With the increase in the number of connected devices, the interactions between humans and devices also increase, leading to complex environments. This has led to a dramatic increase in the number of applications in different domains and the need for a unifying platform to mange the various devices, technologies, standards and methodologies. This continuous evolution of IoT has resulted in the creation of an extensively large information pool. IoT creates a virtual representation of the real world and the applications are required to proactively perceive, comprehend and adapt to the changing real world scenarios especially in environments that are non-stationary in nature [51]. With the ever increasing scope of applications in diverse domains, it is very important to study and validate the real-time implementations of the IoT architectures. This diverse application domains makes it difficult to create one architecture which fits all applications. The reasons are differences in operational, functional, domain-specific, hardware and software requirements. The aim is then to have a modular architecture which can be adapted to different applications with minimum efforts. Rather than reinventing the wheel, re-use and adapt should be strategy. The architectures should take into account the following factors:

- Networking and Connectivity: Given the number of devices, the need to hide the technological heterogeneity in connecting the devices is predominant. With the differences in communication protocols, standards and several sections of devices having propriety protocols,a communication layer to handle this has to be addressed in the design phase. This leads to the interoperability, as with the number of different vendors, it is essential to adopt a minimum open standards to facilitate a seamless integration. We are now transitioning to the phase of the definition of such common standards.In addition the timing requirements of the various applications will have to be factored in, as some applications require a near real-time response, whereas others require historical data to make inferences.

- Computational Capacity and Storage: The billion devices generates a large amount of data. It is important to define the rules and regulations governing the storage and deletion of data. Database requirements, data models, storage and retrieval mechanisms will be considered. In addition, with the cloud solutions getting more popular, many of the data and computations are now moved to the cloud, which makes it possible for the service builders to design solutions which require higher computational power.

- Security: This remains an open challenge to the IoT and applications. With devices now ubiquitous and seamlessly integrated into the very fabric of our living it is important to address the security and privacy concerns of all the parties involved in the service provided. Right from the data provider, to the service developer to the end-user.

The realization of the IoT architectures poses several challenges including design, technical, data governance and social factors. Exploiting the available contextual information, deriving meaningful knowledge and reasoning continually about its consequences defines an intelligent adaptive system. An active area of research includes designing adaptive autonomic computational models for evolving environments that addresses both, the growing complexities brought about by the increasing number of environmental objects and their inter-relationships, and the evolving environmental parameters which should be reflected in the computational models. Changes are a part of the evolution process of nature and computational models reflecting real world scenarios should be able to evolve with the changes and time otherwise they tend to decay over time and often become unused services. The state of the art shows a growing interest in this field, and also provides solutions for specific problem domains. With the IoT becoming more prevalent, more intelligent applications are developed by exploiting the knowledge that is built by observing and understanding the real-world parameters. Hence, addressing this problem of change detection and adaption, and positioning it within an IoT framework moves towards developing systems with a higher degree of intelligence and awareness. Given the large number of applications that are being developed, the need for the framework to be highly adaptable is important. .

1.2 Motivation

The motivation behind the work was guided by the EU FP7 project, iCore, which was in the process of defining a cognitive IoT architecture to used to various application domains. The IoT architectures, at the time of the start of this thesis were limited in scope and number. Architectures focused on building vertical applications. The shift towards a horizontal architecture which brings together the vertical silos of information into one information portal was gaining prominence. Also,the IoT architectures aimed at providing the necessary components of applications based on the analytics performed on the data coming from the real world. This involved the integration of learning models and machine learning algorithms. As the real world evolves, changes are observed in the contextual parameters of the objects and its associated situations. These changes should be reflected in the models and failure to adapt to the changes will demerit the functionality of the application, as it progressively degrades over time.
In this thesis, we present an adaptive architectural framework with an adaptive framework implemented in applications from different domains. It addresses the challenges in the integration of technologies required to realize the architecture and the differences in each application domain are studied. The unifying factor in all the use-cases is the need for a IoT architecture and the requirement for adaptation. The first step therefore was the architectural design and the definition of the building blocks to realize the adaptive functionality that re-configures to changes. Validation scenarios in different application domains are defined for testing the adaptive architecture: a hospital test-bed, wherein a model to predict medical objects' movement patterns is designed (Predictive Modelling), in smart home application scenarios and in human behaviour prediction. Another important chapter in the thesis is the application of the developed framework in an Industry 4.0 application. Here, with the transition to a digital solar lab, use-cases are explored which will evolve with the changing real time information. Given that this PhD was conducted in collaboration with industrial projects, the goal of the thesis, therefore focuses also on the implementation aspects of the IoT applications and the design considerations that were deliberated. In addition, we have analyzed the challenges faced due to the convergence of various technologies, inter-dependencies and highlights the need for a collaborative task force in order to realize the potential to its maximum.

1.2.1 Thesis Contribution

This thesis tackles the following: (i) A common unifying IoT architecture through a study of existing architectures and their advantages with focus on mechanisms to to learn and adapt to the changing real world parameters. (ii)Definition and implementation of use-case test scenarios, where the adaptive IoT framework, will be deployed and tested. Implementation scenarios that have been defined include:

- Smart Asset Tracking Application: Deployed in a hospital, the application tracks medical objects and also focuses on learning the movement patterns of the medical objects. The advantages of the platform, to generate more services such as maintenance records, tracing misplaced objects and optimizing the energy of the localization technology is highlighted

- Supporting the development of modules for the final architecture, component level prototypes and simulations were performed.
  - Smart home applications such as room environment control. This provided the basic communication modules to sensor platforms, virtualization and storage functionalities in addition to dealing with the service level user interactions. The application was tested in the office premises and showcased in various events.
  - Modelling human activity pattern based on open data set using hidden markov models. This was conducted as an initial prototype so as to reuse the module as part of the architecture to study movement patterns of the medical objects.
  - An internship study period was defined to study the possibility of including indoor localization technologies to the architecture. Presence detection algorithm was developed using bluetooth beacons and a corresponding android application. Here also, the components include the possibility to virtualize objects and integrate services in the architectural level.
- In an Industry4.0 setting, the platform is used as the foundation to start the digitalization of a solar cell research laboratory. The contributions include the design, implementation and the validation phases and is part of the ongoing work.

Some of the related challenges include: (i) Formal representation of objects, virtual objects and knowledge along with its associated changing contextual properties in order to be compatible with the existing representations adopted in the IoT architecture. (ii) The modular architecture implementation across different domains still imposes challenges due to the difference in the underlying requirements, technologies, communication standards, data requirements and interaction with the end-users. (ii) During design phase it is not possible to anticipate all changes, so it is important to set the boundaries of expected changes that will be tackled. The boundaries also extend to define the time-window lengths to observe changes. (iii) To include user feedback in the adaptive model to make the system proactively responsive to changes is a challenge. The above mentioned aspects will be touched upon briefly while addressing the main problem statement within the scope of the implementation as enablers that will be defined within a constrained boundary.

1.3 Thesis Structure

The thesis is structured to reflect the various applications developed during the course of the study. The chapters are sub-divided as follows:

- Chapter 2: Adaptive Cognitive Architectural Framework. It explains the framework developed with details on the adaptive and self-learning aspect of the architecture. It includes a sub-chapter detailing the implementation of a use-case 'Asset Tracking in Hospitals'
- Chapter 3: Related Work on Application Implementations focuses on the related work done in the lead up to the realization of the architecture and sub-projects

done as part of the overall work. It incldues protypes of a smart home room control application, human activity modelling and indoor localization module.

- Chapter 4: Industry 4.0 and IoT convergence - introduces the Industry 4.0 concepts and the convergence of IoT and industry. The adaptation of the developed framework for an industrial application in a solar research lab is presented.

- Chapter 5:Conclusion and future directions. It presents the summary of the work done in the thesis from the design, implementation and the lessons learned. It presents the need for the inter-disciplinary collaboration, challenges faced and how it is essential in building of ecosystem of the next generation of the smart applications and industries

Chapter 2

Adaptive IoT Cognitive Architectural Framework

2.1 Introduction

Connecting the growing number of heterogeneous devices with different underlying technologies requires a suitable architecture. Such architectures (or IoT platforms) are an important part of the IoT ecosystem. The important features of a platform include as device management, connectivity manager, service management, data analytics, visualizations and security. It should have all the underpinnings to make the application development process quicker managing all the interactions between the devices and the applications. It is also very crucial for the platform to be scalable and open to customization since the applications can span many application domains. The project iCore aimed at developing a cognitive IoT framework for the creation and management of smart services by leveraging on virtualisation of real-world objects and accessing the resources. The important aspect of the platform that was developed and implemented as part of this thesis is the adaptive loop to account for the changing user situations based on both the information coming from the real world and the system level information.

The proposed architecture is three tiered linking the devices or the physical objects to end-user services. The first layer is the Virtual Object(VO) Level which presents the semantic abstraction of the real-world object(e.g. sensors, actuators). The second layer is the Composite Virtual Object(CVO) Level which provides a aggregation of VO level features. The third layer is the Service Level (SL), maps availability of underlying CVO/VO features to the needs of end-users and associated IoT applications. The interaction with an iCore system is initiated through a Service Request generated for the purpose of activating data streams from IoT objects and continuously processing these to support an end-user or ICT application with a set of processes monitoring a situation and producing alerts when particular conditions are met.

The deliberation for the various aspects of the architecture was carried out with the different partners. In order to demonstrate the cross-functional approach of the architecture, several use-cases were defined and implemented. Our research focussed on the smart hospital application to tackle the tracking of medical objects in a hospital, to continuously locate and assess status and maintenance needs of medical equipment in a large unit of a hospital in Trento and route operators to these objects in a situation-aware way. This trial was meant to show the value iCore can bring to SMEs in reducing time-to-market for deploying solutions for the management of spatio-temporal IoT generated events in a variety of application domaiWe have applied the above mentioned architecture to the application scenario. The following sections delves in detail about the use-case and the implementation aspects.

2.2 Use Case Description

Modern hospitals are complex logistic systems composed of a number of departments and rooms containing medical equipment and non-medical devices that are used by the people including medical staff, patients and visitors. The portable medical devices such as defibrillators, ultrasound systems are transported from one department to another and are often misplaced. Location of the medical devices and their usage statistics, i.e. if the device is in the authorized department/misplaced and frequency of usage of the device represents useful information. Data analysis generates inferred information that can eventually lead to improved services. Current systems are not equipped to collect and analyze such information. Most of the asset management systems today are managed by the medical personnel. Solutions that can provide a better asset management is highly beneficial in such time-sensitive domains, where overhead in looking locating devices can be utilized to provide better care. In addition asset management together with associated applications provides significant time and energy savings e.g. tracking portable medical equipment are often misplaced or unavailable.

In this work, we present a solution for the asset management in a hospital leveraging the cognitive IoT platform to empower the objects and the application to observe and react to the situations happening in the real world. The emphasis here is on the implementation of the horizontal IoT platform which can be extended for further integration of services, e.g. tracking of medical devices, schedule maintenance of the medical equipment, etc. The cognitive IoT framework described earlier is adapted for this application scenario. While this work provides a generic framework, this paper looks at the technical design choices for the generic architecture and is adapted to the requirements of the hospital domain. In particular, the proposed solution ensures the integration of various technologies related to the IoT, therefore extending standalone tracking applications such as the Radio-Frequency Identification (RFID) technologies [24] which are widely available. For example, the asset management described in this work is featured by the translation of user requirements to locate medical devices in real time, modelling the movement patterns and ensuring energy efficiency of the system. Besides, we highlight the development process, requirements of the cloud interactions between the various enabling technologies, advantages of a cognitive IoT in this scenario, challenges faced, and the easy integration of various software and hardware technologies that demonstrate the technological heterogeneity of the platform.

Another significant contribution of this work lies in the application of open software and real hardware, as well as the platform deployment and testing in the premises of Santa Chiara hospital in Trento, Italy.

2.3 Related Work

In introduction we have identified the IoT as an emerging and promising technology for smart-x applications including the medical ones. At the moment, there is a lack of real deployments of IoT based test-beds and trials [21]. Those deployed, typically exploit IoT paradigm as a 'global connectivity platform' by connecting and interfacing the objects to the Internet and do not employ cloud opportunities and cognitive technologies. Consequently, more close cooperation among the state-of-the-art technologies and real world is expected by the IoT community [49]. It should be noted that there is a big gap between the IoT research where a number of smart and intelligent approaches have been presented so far [55] and industrial, real deployments which sometimes rely on straightforward solutions for the sake of reliability and robustness. This discussion is conducted in more details in [60].

In this respect, the state-of-the-art work related to our contribution can be divided into two parts: (i) real IoT-based deployments and (ii) relevant IoT frameworks with cognitive capabilities.

IoT deployments. The research in the applied and practical IoT is still fragmented and lacks real deployments. Although there is a number of available experimental IoT testbeds [21], they are aimed at research and investigation of particular scenarios, protocols, firmware and tend to be more 'do-it-yourself' ones [8] rather than be really integrated in our everyday life. For example, in [39] the authors try to address this problem by deploying the user-centric IoT-based *SmartCampus* testbed consisting of heterogeneous IoT devices in a real life office environment. The authors demonstrate that the realized deployment can serve for a number of use cases, e.g. energy efficiency in buildings and monitoring of human movements in the office environment. The study conducted within this research activity is aimed at the improvement of the building infrastructure, reduction energy costs and understanding the human behaviour.

IoT deployments in transportation domain are presented in [17][47]. In [17] the development and integration of a retractable bollard management application is presented. The goal of this real life IoT deployment is to restrict the access of non authorized vehicles to the city center of Bologna, Italy. This deployment adopts M2M technology to connect the objects [50], i.e. cars, with virtual city services and perform simple logical operations, e.g. if a car ID is in a data base than let it enter the city center. Work presented in [47] performs more advanced inference procedures. The deployment is realized by installing inductive loops on the crosses of the city of Enschede, the Netherlands. This technology enables counting and classification of detected vehicles. Together with the IoT cognitive framework, the sensors data are enriched with a context aware information which helps to infer about, e.g. the traffic jams.

An integrated system for regional environmental monitoring and management is deployed in Xingiang, China [16]. The proposed approach uses not only specific sensors, but includes geoinformatics, online services and data bases containing past records on the environmental status. Cloud computing and intelligent algorithms, e.g. decision making, assist in data processing and making inference procedures out of available data.

The ultimate goal of this work is to integrate the IoT deployment in a real environment as it is done in [39][47][16]. Also, we use smart IoT solutions to ensure intelligent service provisioning instead of exploiting IoT as a communication medium [17].

IoT cognitive frameworks. A recent work on cognitive IoT [59] highlights gaps in existing facilities and identifies how cognition helps to bridge real world with high level intelligence. Semantic technologies and virtualization techniques are the major ingredients for bringing real world objects into the virtual world of IoT and, specifically, to address the problem of heterogeneity of the objects. The authors of [1] apply virtual sensor abstraction to hide the complexity in large-scale interconnected sensor networks. Semantic technologies are used in [55] to describe any real world object as well as to enrich it with context-aware information useful for further inference.

2.4 Deployment Scenario

The deployment has been executed at the "Santa Chiara" Hospital of Trento (Italy) within the Neonatology unit of the Paediatric Department.

The indirect "end-users" are the newborn babies who receive care and attention of the medical personnel. It is therefore important to relieve the medical staff from having to dedicate their valuable time to non-core activities. There are around 240 medical devices in the department utilized to provide high-quality medical service. In order to guarantee good healthcare services, efficient time management can be achieved with a well-managed environment,which can track, monitor and maintain all the available technologies and devices that are present inside the department.

To enable doctors and nurses to concentrate mostly on their core duties, an asset management solution is presented.The devices are IoT enabled and thereby become more proactive and generate notifications and statistics. The medical devices in the hospital are tagged with sensors.Figure **??** shows images of the sensors tagged to the medical devices in the hospital. For the first implementation, 40 medical devices were tagged with positioning sensors to be tracked. The medical devices are categorized into the following types: portable ultrasound, volumetric pumps, syringe pumps, ventilators and incubators. While traditional positioning systems can be used in this scenario, a system which provides more than just location updates is required given we are tackling a multi functional problem solution.

2.5 Smart Asset Management Architectural Framework

The architectural framework of the Smart Asset Management Application (SAM) comprises of the core framework and related components in addition to the enablers required to fulfil the requirements. Figure 2.1 presents the overall framework of

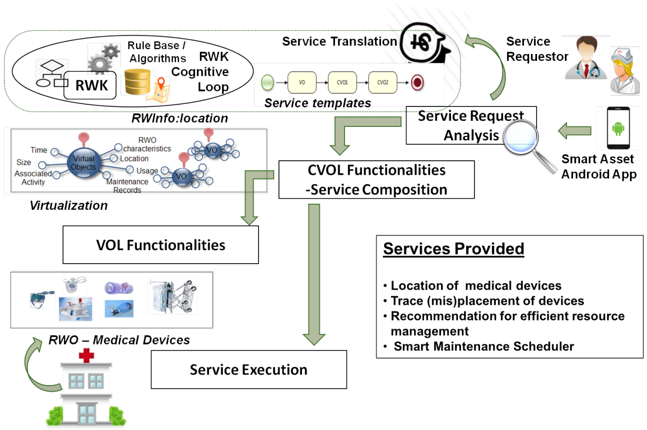

FIGURE 2.1: Overall Smart Asset Management Framework Functionalities based on IoT

SAM, each of which is detailed below. The service requesters represent the medical personnel and end-users of the application. They issue service requests and an android application handles the interactions with the service requester. This request is then passed on to the core of the SAM framework, which processes the request and determines all the required components needed to fulfil the request. The execution of the selected components provides the desired service. The services provided by SAM include:

- Locate Objects: Provides location of medical devices
- Trace/Monitor Objects: Monitors a device and triggers events every time the device leaves a pre-determined geo-perimeter. An alert is sent if the device is not returned to its original perimeter after a pre-determined time limit.
- Maintenance of Objects: Provides the status of maintenance of devices viz. due for maintenance, out of maintenance and functional. It can also send alerts to the maintenance personnel when a device is due for service.
- Power Plan Recommendations: This feature provides recommendations to the indoor positioning system to ensure an energy efficient system.

The framework is implemented in a Java environment and the key contributing technologies include real-time positioning system, geo-information system based functionalities, REST and MQTT communication protocols, database technologies, rule based engines, machine learning, BPMN based workflow engines and an android application.The paper brings out the ease of integration of technologies to the IoT platform, thus moving from vertical solutions to an encompassing solution.

2.5.1 SAM Core Framework

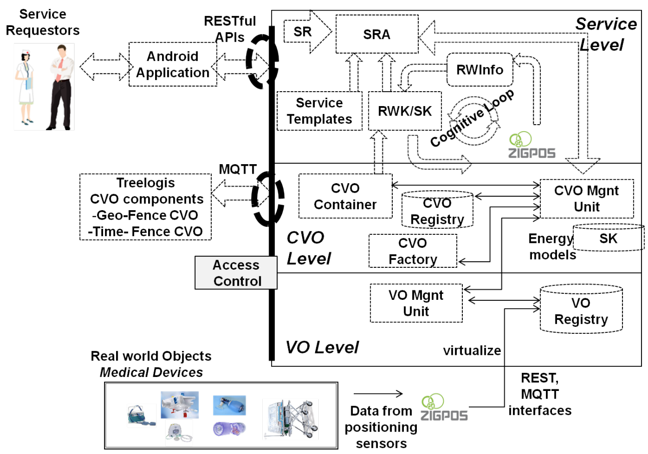

FIGURE 2.2: Core Smart Asset Management Framework

The core framework has three levels of enablers as depicted in Figure 2.2: (i) Service level(SL): Mechanisms to translate service requirements, build and exploit real world knowledge to improve rendered services (ii) Composite Virtual Object (CVO): Cognitive mash-up of VOs facilitating improved execution of services requested by the users (iii)Virtual Object (VO): It represents the virtual semantically enriched representations of various real world objects.

Service Level Components

The Service level performs the functionality of receiving the service request from the end-user and translating it into the required VOs and CVOs. The sequence of operations performed at the service level are:

Service Request Translation The service request (SR) represents the requirement of the user for a particular service. Examples include *"Locate objectX"*, where *objectX* represents any one of the medical devices that are tagged to be tracked. The first step is to translate this SR to extract the functional tokens by means of Natural Language Processing Module. The implementation is based on the OpenNLP library [19], a machine learning based toolkit. The OpenNLP supports tasks such as tokenization, sentence segmentation, part-of-speech tagging, named entity extraction, chunking, parsing, and co-reference resolution. In the implementation, the functionality is currently restricted to the extraction of "functional" tokens such locate, trace, and maintenance. However, the integration of the whole library makes it possible to utilize the other functionalities of NLP as required. Hence this is a reasonably modular and standalone component that can be improved as per requirements.

Service Request Analysis The functional tokens obtained from the SR translation are the basis on which the Service Templates(ST) are chosen. The ST consists of the list of VOs and CVOs that are required to fulfil the SR. STs are implemented as workflow based models using the open source business integration platform, Drools [11]. Drools supports the JSR-94 standard for the business rule engine and enterprise framework for the construction, maintenance, and enforcement of business policies in an organization, application, or service. Business process facilitates defining requirements by describing the steps that need to be executed to achieve that goal and the order, using a flow chart. This greatly improves the visibility and agility of execution, resulting in higher-level and domain-specific representations that can be understood by business users and is easier to monitor.Figure 2.3 shows a graphical representation of a sample service template. The two circles represents the start and end of the process. The underlying model is an XML file. Each block represents an execution sequence which can be defined in a rules file. The rules are simple *if..then* conditions. The Rete algorithm, which is a pattern matching algorithm for implementing production rule systems, is used to determine the rules execution sequence.

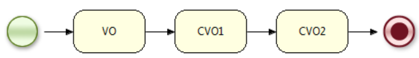

FIGURE 2.3: Sample Service Template based on Drools

On the basis of the ST and after a cross-reference with the real world knowledge model, which is elaborated in the next section, the service execution request(SER) is formulated. The SER is passed on to the CVO Management Unit to perform further actions. The format of the SER in the implementation is a *hashmap of parameters* containing all the information of the required VOs and CVOs.

Real World Knowledge The 'real world' defines the scope of the application domain and knowledge represents all the available information and rules/behaviour that characterizes the defined scope. Any information that comes from the real world is defined as the RWInfo. It plays a vital role in influencing the functioning and execution of the service level requirements. In order to leverage on the abundant data coming from the real world a RWK Cognitive adaptive cycle is designed. Cognitive processing and functionality refers to the existence of a software intelligent agent that encompasses the important dimensions of perception (level 1), comprehension (level 2), adaptation (level 3) and learning from feedback, in order to create awareness of a situation (e.g. VO or CVO situation) [13]. Effective situation awareness is a critical element for decision making support in a wide range of scenarios. Applying situation awareness can greatly increase operational effectiveness, by improving the quality and timeliness of decision making. A cognitive processing cycle that can be used to create an awareness of a VO or CVO situation consists of three principle phases as illustrated in the 2.4. Situations that exist within a specific time and space frame are ultimately represented in terms of a knowledge framework. The knowledge base can then act as a mechanism for initiating relevant decision based outcomes (i.e. actuated responses), triggered by system or user queries.

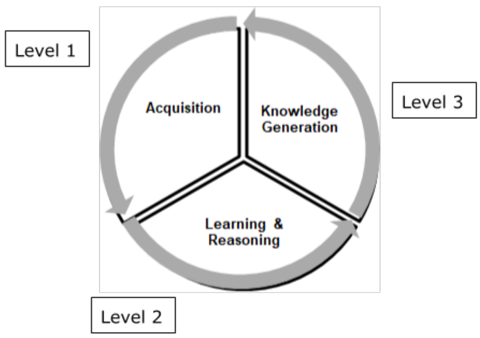

FIGURE 2.4: Adaptive Cognitive Cycle to handle knowledge generation

In 2.4, an adaptive cognitive processing cycle consists of basic functionality blocks that can meet the needs of the application and generally encompasses the following functions:

i. Process and enrich data based on application, user or real world object behaviour – Level 1 - Perception ii. Acquire data/Knowledge Query –Level 1 - Perception iii. Apply/Derive a case based reasoning – Level 2- Learning iv. Search for prior knowledge of scenarios for solutions – Level 2- Learning v. Apply learning methodologies to derive solutions – Level 2- Learning vi. Generate/Update knowledge base – Level 3 - Adaption

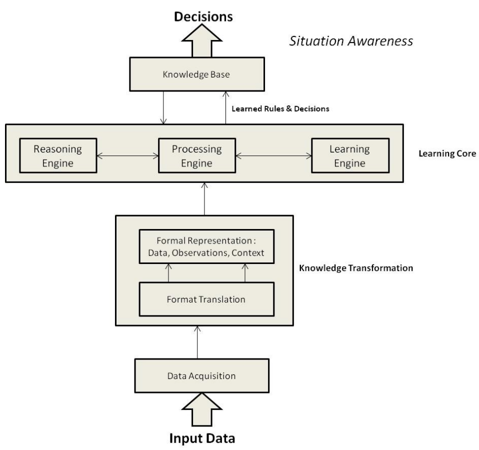

FIGURE 2.5: Adaptive Cognitive Engine

The functions listed above, which form part of the overall cognitive processing cycle, can be formally represented in terms of a cognitive engine, used to derive the eventual knowledge base as shown in 2.5.

Data Acquisition represents the first stage of the cognitive processing cycle. The data is obtained from various heterogeneous objects and hence the necessity arises to translate the data to a specific format, which is handled by the format translation block. The formal representation block translates the data to more meaningful data based on semantic models. The data is represented in a structured format that facilitates abstract definition of objects, their characteristics and derivation of relationships between the relevant elements. Based on the service level requirements, the data enrichment is also done to include additional contextual information such as, data access rights and data certainty measures for building confidence in abstract definition of objects. Cognition within the processing lifecycle can be broadly subdivided into the reasoning and learning from experience modules. The cognitive engine can then make decisions based on both the modules or from using just a single module. Thus the intelligence of the cognitive engine sits in making decisions supported by the reasoning and machine learning models. Machine learning algorithms are designed based on the application requirements. There are various machine learning algorithms that could be applied namely Neural Networks, Support Vector Machine, Fuzzy Logic, Genetic Algorithms, Hidden Markov Models and Statistical Pattern Recognition Techniques. While the list of available methods is huge, a case by case analysis is being done to identify the optimum learning methodologies that will be best suited for IoT architecture. The main parameters that are being considered in the algorithm selection process are scalability, the computational complexity and the time to learn and execute decisions. The algorithms will be optimized for and against system performance related metrics. A separate reasoning engine is designed to derive relationships from the semantic representation of the data. The reasoning engine makes decisions based on various factors namely rules, mathematical formulations, logic, and probability reasoning based techniques. The choice of the reasoning logic is dependent on the use case/application.

The process of transforming the RWInfo to knowledge is defined as the knowledge building process realized by invoking the cognitive loop mechanism. The cognitive loop employs statistical and machine learning techniques to model a targeted behaviour of the real world based on the RWInfo to eventually generate new or updated knowledge, which helps to optimize a performance criterion. Here,the RWInfo is the location information of medical devices over time. These observations are logged and used to build prediction models.

The prediction model provides the estimated locations of objects even when the positioning system is unavailable. Another aspect is the discovery of the hidden correlations and associations in the data. Given the large volumes of data available, it requires an intelligent model to extract these relationships. While the initial RWK model is designed by the domain experts, the RWK cognitive loop continuously observes, learns and adapts the models to changes in the environment. RWK models could be exploited to predict, classify, cluster and make inferences from data. The cognitive loop handles both space and time complexities with a desired level of accuracy.

Facilitating the real world knowledge building process, there exists one RWK representative model capturing all the parameters and the relationships with the various components in the environment under consideration. This is tailored to reflect the domain of interest. This facilitates an enriched data model capturing all the characteristics of the real world including the extracted knowledge, thereby becoming the precursor to making decision. It is implemented as a MySQL database as represented in Figure 2.6. It shows the classes of hospital, medical devices, rooms and personnel. The hospital has an id and description. It also depicts the relationships e.g. a one-to-many relationship between the room and medical devices i.e. a room can contain multiple medical devices but each device can be only in one room at a time. It also can include the personnel information with the access rights.

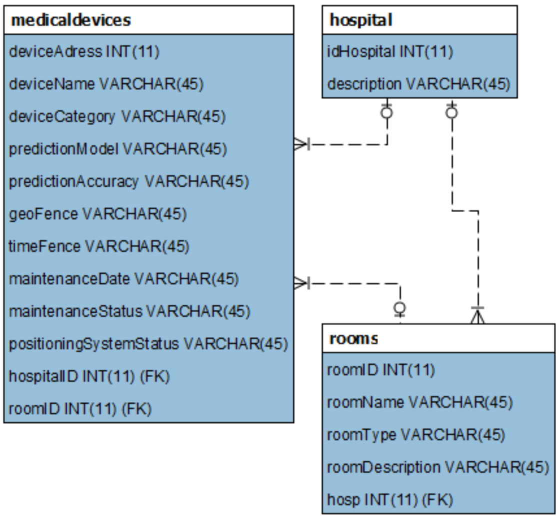

FIGURE 2.6: Capturing relationships of medical devices in MySQL database

RWK Prediction Model RWK Prediction Model: In order to facilitate the knowledge building process to realize the prediction models, relevant data is required. Given that the hospitals are highly-sensitive areas where only the final deployment will be done. Hence a test-bed with the scenario mimicking the original scenario was set-up for the initial data collection phase. Data analysis and modelling is performed and eventually the model based on the simulated data will be tuned to data from the actual scenario in the final deployment. The knowledge building process is depicted inFigure 2.7. The modelling process includes a data collection, data pre-processing

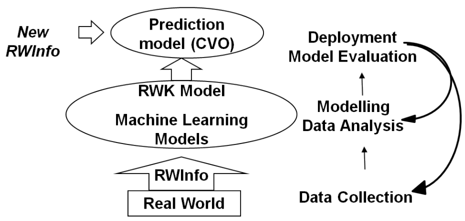

FIGURE 2.7: Modelling process generating knowledge

and a model development phase. Two prediction models namely Auto Regression (AR) and Markov Model (MM) are considered. Here a brief description of the AR model is provided. The goal of prediction mechanism is to accurately predict the next location of the object so that the sensor on the device being tagged can be put to sleep mode to save energy. An autocorrelation based transversal filter can be described as follows:

$$u(t) = u(t-i)(i) \tag{2.1}$$

where *u(t)* is the predicted time-series value at time instance t are the previous p values of time-series; p is the order of the autocorrelation model.
Once the filter coefficients are determined, the current value of the series can be estimated using the past values. To find the filter coefficients, a set of training data

is required from the time series. Suppose, *Tp* represents training data *(u(1), u(2), ..., u(Tp))* available from the series. Then, the filter coefficients can be found by solving the following linear equations:

$$u_\alpha = u \tag{2.2}$$

$$\alpha = [\alpha(1), \alpha(2), ...\alpha(p)]^T \tag{2.3}$$

$$u = [u(p+1), u(p+2), ...u(T_p)]^T \tag{2.4}$$

is an over determined system of linear equations, as the number of equations are more than the number of variables p. As U is not a square matrix, a unique solution is not possible, and an approximate solution can be found using least-square method. The least-square solution of the above equation can be found using the following equation

$$\alpha = (U^T U)^{-1} U^(T) u \tag{2.5}$$

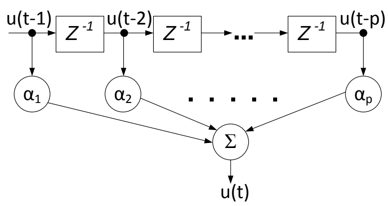

FIGURE 2.8: AutoRegressive Model

In a nutshell, as shown in Figure 2.8 autoregressive (AR) process models the conditional mean of *u(t)* as a function of past observations, *u(t-1), u(t-2)…u(t-p)*. An AR process that depends on p past observations is called an AR model of degree p, denoted by AR(p). To test the AR model, we train the AR model with the training dataset comprising of the object location and obtain the parameters of AR. The predicted location of the object is obtained using the model for different time granularities.

Composite Virtual Object Level Components

The functionalities of the CVO level are sub-divided into the following modules:

CVO Management Unit The CVO Management Unit performs numerous functionalities. It receives the SER from the service level. Based on this SER, the CVO initiates a discovery process that to lookup for the CVOs as specified in the SER. The CVOs reside as templates that are available in a local database. These templates that can be instantiated and executed.

CVOs and CVO Container In the implementation, the following CVOs are implemented: Locate CVO, Trace CVO, Recommend CVO, and Predict CVO. The CVOs can also represent external services that can be called and controlled from the CVO Management Unit. The Locate and Trace CVO represent two such instances. More details of these CVOs are found below. The CVO registry holds information of which CVOs are available to satisfy the Service Request. A situation observer is defined, which in this case is the discriminator, which helps the platform select automatically the most suitable CVO to fulfil the request based on context. The CVOs are executed in the CVO container. It is a workflow based execution engine. The implementation is based on Drools [11]. It largely consists of rules that define the execution flow, with features of complex event processing. The CVO container is also linked to the MySQL database to store the outcome of the executions, which can then be utilized by the subsequent service requests as knowledge to provide better services. The CVO templates are executed as defined by the execution flow in the service templates, enhanced with the information from the RWK. The rule based format of the Drools engine also facilitates easy integration to calling other services.

- Locate CVO: This CVO is responsible for returning the location of a particular medical object. On invocation, it calls the real time positioning system to determine the location of the object. The real world objects in the IoT platform are represented by the medical devices that are tagged with the sensors that are responsible for determining the position of the device. The indoor positioning solution used in this implementation is provided by ZIGPOS. ZIGPOS eeRTLS is a real-time location and tracking system that operates over wireless sensors and actuator networks providing the possibility to estimate accurate position of objects. The communication is based on the global standard IEEE 802.15.4, 2.4 GHz frequency band enabling interoperability with existing wireless solutions. We note here that the localization mechanism is out of scope of this paper, [61] provides more details regarding the solution. The architectural set-up of the positioning system comprises of the following components and is interconnected as shown inFigure 2.9.
  - Mobile Tags: Tags are small wireless transceivers; people and physical objects can be equipped with these tags for tracking purposes. These are battery powered tags that are placed on the medical devices to be tracked.
  - Anchor nodes: Anchor nodes are responsible for triangulation of the position of the mobile tags. The position of the anchor nodes are usually fixed.
  - Gateway: The Gateway is the interface between the localization network and the IT- Infrastructure. At least one Gateway is required for the whole network.
  - Localization server: Position evaluation, visualization and tagging of mobile tags is done on the PC.

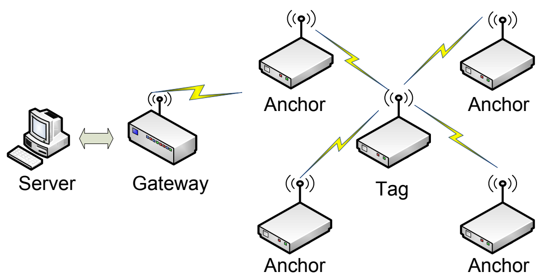

FIGURE 2.9: Indoor Positioning System Components

The information from all the anchors is consolidated on a server and used to determine the absolute location of the object. The information from the positioning system resides in the cloud server and can be accessed via REST interfaces and MQTT interfaces. This compatibility with standard communication technologies for IoT ensures that different hardware platforms can be easily integrated, provided it adheres to the defined API. The information that is retrieved at the client side includes list of devices registered in the server, position of the devices and other system level information.

- Trace CVO: The Trace CVO invokes the tracing features i.e. geo and time-fencing features provided by the external software component, Treelogis[20]. It is a geo-information based system (GIS) which can provide tracking solutions. It receives input from the environment and uses geo-spatial and temporal methods to provide features such as time-fencing and geo-fencing. Geo-fence refers to defining geo perimeters within which an object remains. When the object enters/exits the perimeter an event is generated. Similarly for the time-fence, it functions like a timer firing event. Time-fences ensure monitoring of situations when an object has not returned to its original place/ tracking maintenance records of the devices. The solution provided is generic, which implies that it is not tied to a particular positioning technology. APIs are provided for accessing these services which currently resides in the cloud. MQTT based communication is the design choice, as an pub-sub event based model fits the requirements of the features very well. Figure 2.10 represents an example map with the geo-fencing boundaries depicted. Each object represented as dots is defined to remain within a particular geo-fence. Every time it leaves the boundary, an event is triggered.

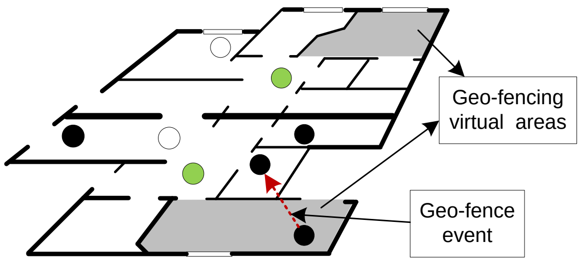

FIGURE 2.10: Treelogis Software triggering geo-fence events

System Knowledge Model It is representative of the system related information that enables the virtualization of the objects. The system information includes the power modes, battery levels of the ZIGPOS devices along with the energy model of the deployed infrastructure. The energy model contains the energy consumption data in the various power modes. The positioning system is based on a time-synchronized communication and distance estimation approach. The energy consumption also depends on other parameters such as the required level of accuracy, the number of anchors used for triangulation, etc. Based on the energy model, rules have been extracted in order to provide the necessary knowledge to allow the recommendation CVO to provide recommendations for an energy efficient power plan. Thus RWK and SK together facilitate the efficient use of resources in addition to fulfilling the requirements of the SR and providing a reliable service.

Virtual Object Level Components

The VOs are represented by the medical devices which are tagged with the positioning sensors. In addition to which there are the anchor nodes, which are deployed along the perimeter of the test environment, and are responsible for triangulating the position of the devices. VOs are registered on the system at the time of deployment.It contains sensor information i.e. location information of objects, name of device, unique address, description, custom name, time-stamp, movement information and accuracy of readings. The semantic enrichment of VOs i.e. adding additional contextual details is also performed e.g. adding geo/time fence perimeters which are required in order to define boundaries where the entry/exit events will be triggered. The VO Management Unit is responsible for managing the VOs and has access to the VO registry. It acts as the communication interface between the VO and CVO levels. It continuously monitors the performance and availability of VOs. It interacts with the tracking and positioning systems to populate and update the VO registry with device information. Updates associated with a VO is handled via the MQTT communication protocol. MQTT is a publish/subscribe, extremely simple and lightweight messaging protocol, designed for constrained devices and low-bandwidth, high-latency or unreliable networks.A public MQTT broker implementation based on Mosquitto MQTT [18] broker is configured on the server. The communication between positioning system is via REST, where information such as the list of devices can be retrieved by RESTful commands. Also the system information e.g. power modes can be retrieved via REST. In addition to which, the information from the positioning system i.e. location updates are provided through MQTT. The initial deployment thus exposes the interfaces/topics which allows for the search and discovery of the VOs. All the information of the VOs is stored in a local VO registry which facilitates a quick discovery process. The MQTT interfaces are structured so as to receive all the location updates if the "wildcard" is utilized on the topic or one can listen for the updates of only a particular VO. For example, "/assetPosition/+" provides the location updates of all the devices whereas "/assetPosition/assetId" provides updates only the specific device.

2.5.2 Security

The hospital scenario represents a highly protected environment and thereby the data coming from such an environment should incorporate privacy policies. The privacy of the data has to be ensured. In order to address this issue, a security toolkit [40] has been incorporated into the framework. The toolkit is specifically designed for the hospital domain particularly focussing on the access policies and restrictions. The toolkit is implemented in Java and is compatible with defining access policies over the MQTT interfaces. Figure 2.11 shows a screenshot of the example security policies. It shows the various configurable parameters:identities, roles and context that can be defined.

2.5.3 Mobile Application

An android application has been developed to facilitate easy interaction between the user and the services offered. The android application has a front-end graphical user interface which interacts with the back-end to retrieve the service outcomes. The back-end interacts with the core server to push the service requests and retrieves the response. The Jersey libraries have been used to implement this functionality. For example, the outcome for a locate service request is the location of the selected object

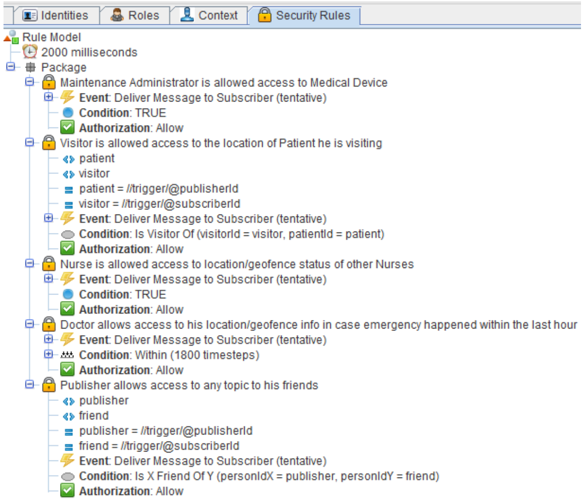

FIGURE 2.11: Security Toolkit in the SAM framework

and this is displayed on a map with the floorplan of the hospital. Additionally, the application has a SQLite implementation to store the necessary data on-board the phone.

2.6 Discussion and Results

The technical validation of the functional parameters of the smart asset management has been carried out. The tests focused mainly on verifying the functioning of each component and the interaction with other components. The test cases of interest are summarized below.

Inter Communication The communication between the components showed in Figure 2.12 was tested. APIs have been developed that follow the REST and MQTT communication standards. Given that all the servers are up and running, the interactions took place reliably. While the Internet is universally available these days, there are still issues with connectivity and also in an environment like hospitals where the possibility to add routers to every room might not be feasible due to health risks. One of the problems faced was due to the fact that all the three servers were located remotely and hence the application relies heavily on the internet to perform all its functionalities. Future enhancement in this direction is to have local/cached decision making capabilities as a fall back while the communication fades. The communication works reliably and accurately.

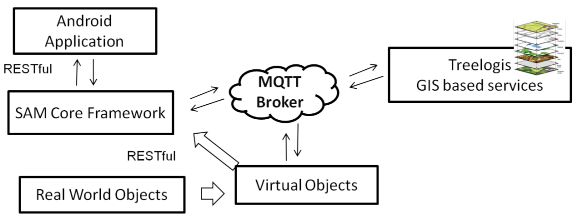

FIGURE 2.12: Communication between the interfaces

RWK Prediction Model For building the RWK models,in the data collection phase the movement patterns from the positioning was collected. Before the work carried out here, a simulation of the object movement and the usage patterns were simulated and the observation were analyzed in a simulation framework described in Appendix A Data samples were collected every 30 seconds and logged with timestamp and location information. In the data processing step, aggregation of two consecutive samples is done in order to eliminate transitions and missing information in the dataset is eliminated by appending the previous location if found. B.2, shows the accuracy of our prediction model. The first row indicates when one entire day data is used to train and obtain model parameters the accuracy of prediction is 83%. The remaining rows shows data that is split every 6 hours and the corresponding prediction accuracies. Splitting data for AR model can increase prediction accuracy only when the object is actively moving in the data split duration. For example, the accuracy for data 12hr to 17hr is 87% whereas 18hr to 23hr is only 5% this is because the object was actively moving during the 12hr to 17hr and more stagnant during the 18hr to 23hr. (As the data was collected in a office environment for the experiments, the objects are not moved after 18hr). These prediction models are validated periodically to compute its accuracy and performance.

Time	Correct Predicted Locations	Total Locations	Accuracy
Complete Data	4782	5760	83%
0hr to 5 hr	1058	1440	73%
6hr	1220	1440	85%
12hr to 17hr	1252	1440	87%
18hr to 23hr	82	1435	5%

TABLE 2.1: Tabulation of AR Modelling Test

Estimated Energy Consumption Based on the system knowledge comprising the energy model of the positioning technology, the energy consumption has been estimated. Considering the network of the positioning devices, we present here the energy consumption for a mobile tag. A detailed description of the energy model is out of scope of this paper. The important parameters include the beacon interval, accuracy settings and static time-slots. The accuracy settings determines the number of anchors, which can vary from a minimum of four to six, that are required to derive the position of the mobile tag. Beacons are used for ranging and also for time-synchronization. Based on the assumption of the above presented prediction model, it is possible to estimate the location of the object. Also given that with >*80%*, the location is known, during these times, the mobile tags as well as the anchors are put in a sleep mode, and wakes up only for synchronization.

Energy Consumption of 1 mobile tag during high accuracy ranging mode $\Rightarrow 38.81e^{-4}mWh$
Energy Consumption of 1 mobile tag during non-active mode $\Rightarrow 0.5e^{-4}mWh$
The energy savings per mobile node is greater than 80%
While the numbers look promising, these do not represent the actual energy savings as the actual process requires the nodes to wake up regularly for synchronization. And also to perform other system specific functionalities. These energy saving computations are therefore estimations and future work will look deeply at validating the

energy savings in the deployments and on the advantages of switching the power modes in such scenarios.

2.7 Conclusion

In this section we have presented the smart asset management framework deployed in a hospital. In particular, we have demonstrated how the framework could be useful in the hospital scenario relying on the IoT paradigm. It describes the technical choices made for the implementation and highlighting the need for a strong platform based architecture to handle the seamless integration of the various technologies. An important aspect of the work was to understand the process in deploying an asset management solution specifically in hospital scenarios. Some of the hospital records, to this day, such as the usage of devices and other statistics are manually recorded. Hence it is not easy to provide custom automated solutions. Hospitals are places where privacy of data should be protected. Hence any solution targeting such environments should have a strong security and privacy policies integrated. The framework modular architecture reveals the point on its flexibility: (i) virtualization technique helps to tackle the problem of heterogeneity of real world objects, (ii) cognitive aspects ensure its application not only for the assets localization, but for another services, e.g. assets tracking and maintenance, as well as to ensure the inference procedures.

The learning engine helps to resolve the problem of outdated models in the framework due to the human centric nature of hospital application. The engine can observe the realm and adapt the framework to new environment 'on the air'. The experimental results present high accuracy of prediction model and high potential towards the energy savings of mobile nodes. These achievements along with the positive feedback from the medical personnel reported on the ease of use of deployed IoT system demonstrate its high potential and practical importance in the hospital environment.

Since the considered application is a human centric one, there are challenges associated with the outdated models. To address this problem we have used a learning engine which can observe the realm and adapt the framework to new environment 'on the air'. The development and analysis of the IoT framework for the asset management reinforce the point that the IoT is a multi-disciplinary field which requires the integration of various technologies and therefore the need for a modular architecture. While the first implementation uses the minimal set of components, the need for a wrapper to hide the technological heterogeneity is highlighted. For example, given that another positioning system is to be used for enhancing the current version such as WiFi or Bluetooth based positioning systems, a wrapper can facilitate easy migration. In addition, the need to be compliant to standards was observed. IoT applications are largely human centric and there are challenges that a model can easily be outdated. While solutions that can be re-configured by the user is the first solution to tackle this, such an approach is intrusive and becomes difficult to adapt to. Instead a learning engine that can discreetly observe and adapt to changes preferences represents the next generation applications. Our implementation of the real world knowledge model takes a step towards that direction. The requirement for a platform when targeting such large scale deployments is also a key lesson learnt. The platform that was used in the implementation was generic but extended and adapted to suit application requirements.

Chapter 3

Related Work for architecture modules prototypes

3.1 Introduction

This section presents the related work developed as simulations and prototypes of the components of the architecture. The chapter has been divided into the following sections:

3.1.1 Smart Home Application: VO,CVO, RWK Module Protoypes

A prototype implementation of the cognitive architecture described in Chapter2 has been published in a conference [44]. The application scenarios in the prototype includes a (i) HVAC(Heating, Ventilation, Air-conditioning) (ii) Medical Status Monitoring. The complete architectural stack was implemented with minimum functionalities. At the SL, the service requests are received from the service requester through a GUI. It is translated based on the service identifiers and passed to the CVO level. At the CVO level, a look up of registries to check for existing CVOs to fulfill the requirements is done. If not available, a new CVO is composed based on lookup from the VO registries. The RWO objects, which are the sensors and actuators have their virtual components enriched by contextual information and is registered onto the Xively platform[1]. Communication to the device is through MQTT protocol and to the iCore service level components through REST interfaces. A real world knowledge model captures all the real world information and stores it. A reasoning engine holds all the rules and constraints to triggers necessary outcomes. A learning engine stores the temperatures preferences of a user along with the timestamps. Using a multilayer perceptron algorithm(MLP),the user preferences was trained.A 95% prediction outcome was obtained for time vs temperatures. Based on the outcome, the reasoning engine has rules that controls the actuators to modify the temperature of the room.

3.1.2 Presence Detection:CVOL Protoypes

Presence Detection using bluetooth beacons: This work was done as part of the internship carried at Bosch IoT Labs, St.Gallen, Switzerland during the doctoral thesis tenure. The lab was working on different smart-home and smart-city related applications. One aspect was the use of learning algorithms to predict user behaviour in order to control the heating in homes. For this we explored the possibility of using Bluetooth to determine the location of a person in the homes. This was conceived with a view that in future with the possible burgeoning use of bluetooth enabled smart watches, we would be able to exploit data from the watched to locate a person.

[1] https://xively.com/

The IoT architectural framework was reused to reflect the real world objects. In this scenario, the people represented RWOs and they carry mobile phones which serve as the localization tag to record movement patterns across different rooms in the test area. The VOs provide the contextual information of the person ID, location and time-stamp. These are then stored in a cloud database. The localization algorithm developed based on the bluetooth ranging, can be incorporated as a CVO which provides the real-time location of the objects. At the application level, an Android application was used to interact with the service requester. The services provided include: Real-time localization, depicted as a map on the mobile phone. The algorithm was deployed and tested in the office premises of IoT lab and 3 residential houses. The results will then be considered to model RWK. More details of the work carried out is added in Appendix B.

3.1.3 Activity Recognition: RWK Module Protoypes

In the domain of Internet of Things, applications such as Smart Homes exploits such probabilistic models in order to learn the behavioral pattern of subjects and thereby provide better services tailored to fit the requirements of the user. Behaviour recognition of the subjects is considered to be an important aspect of the smart homes. Understanding user behaviour is a challenging problem as human behaviour cannot be captured into a single model. The first step in identifying user behaviour is to aggregate all the VOs of interest in the same spatial and temporal frame. For the first stage, we shall consider a scenario where all the VOs present are in the same spatial and temporal frame. The motive of this exercise is to observe user behaviour over a period of time and automate the repeated tasks thereby removing the need to do repetitive tasks. The user behaviour in this context is defined as the all the VOs triggered by the user during the observation period. All the spatial, temporal (time, duration) and causal characteristic of the activities are logged. For determining user behaviour two key concepts are defined: (i) Event – Defined as a VO trigger (ii) Action (situation) – Defined as a sequence of events or VO triggers. Like in all pattern recognition systems, the first step is to obtain data. So an event manager is designed which will monitor and log all the activities performed by the user. In order to establish a behavioural pattern it is important for the event or the action to be repeated over a period of time. The frequency of this repetition is also an important factor that helps to extract the patterns of interest. The time granularity is defined in two stages (i) Large time granularities (days/weeks) (ii) Small time granularity (hours/minutes/seconds). Context is defined as the information used to define the situation of an entity in an environment. The entity could be any object/person which is tracked and observed by the system. Contextual information refers to the additional details including location, physical attributes, inter-relationship with other objects, computational capabilities, etc., that are inherent properties related to the entity. Contextual awareness is the ability of the system to understand the context of the application and the ability to reason and in general be aware of its surroundings. Context awareness is the precursor to the situation awareness mechanism which is built over the contextual information gleaned and tied to real time situations.

Causal reasoning mechanisms are employed to deduce the causal relationship between the sequences of VOs activated and are consolidated to resolve and prevent conflicts and paves the way for generating knowledge. In order to understand user behaviour, the interactions are mostly dealt in the meta-data plane. The relevant information of the VO triggered that are required are (i) VO id (sensor type) and (ii) Time. The meta-data information is filtered with moving time windows that are incremented gradually to establish the period of repeatability and the frequency of events. The thresholds for defining the frequency and periodicity are vital as these hold the key to understanding the behaviour. Data mining techniques will be employed to identify the relevant events that form a pattern in the user's activities. In real world, a user's behaviour is subjected to a wide deviation and influenced by various factors which could due to changes in environment, emotional response of the user or due to some external factor which cannot be classified as part of the routine behaviour. So understanding the routine and eliminating the outliers is a very important task. Causal reasoning is employed to establish the factors that trigger the events and these are brought together while extracting the patterns. These reasoning techniques are proposed as a preprocessing step as it eliminates the need for the learning algorithms to handle inconsistencies in the data and also some of the popular behaviour recognizing algorithms does not handle or resolve the spatial temporal conflicts. The patterns are represented in Markov chains and built on hierarchically based on the temporal information. The probabilities and expectations are computed and actions that are established as part of the routine behaviour. The architecture should support user input in addition to the learning that unfold in the backdrop. The overall model could be classified into various levels of autonomic functioning viz. active or passive/partial or relying only on received preference. The generic cognitive cycle of perception, comprehension and adaptation will be employed to establish and validate the behavioural pattern. The elements of the iCore situational awareness architecture help to build and capture the user behaviour.

There are several methods that exploit the spatial and temporal properties taking the contextual information of the scenario. Behaviour can be represented as a sequence of sensor events that are triggered in a sequence and are repeated over a period of time. Smart homes are equipped with various sensors whose trigger patterns reflect the behavior of the inhabitant. In order to understand the concept of changing real world data, an open data-set recording the movement of inhabitants in a house was considered. For the real world world knowledge generation, a key component is the requirement of machine learning modules. In this section, we implemented two different algorithms and made a comparison of the performances against the bench-marked data-sets. An online database [32] of sensor events for activity recognition is utilized. The concept of the models explored, along with the implementation details and results are provided in the sections below. The data set consists of sensor readings collected from varies sensors such as reed switches, pressure mats, mercury contacts, passive infrared and float sensors deployed around the house. These sensors were triggered when the inhabitants performed their daily activities. The dataset is annotated based on the input taken from a written diary and a Bluetooth headset used by the inhabitants to record their activities. 14 sensors and 10 activities have been considered for the study. For the implementation, the pre-processed data as mentioned in [32] is used for the experiment. The dataset is divided into 24 days. Each day is divided into time segments of ten minute intervals, i.e. 144 time segments. Each time segment has a corresponding activity. Only one activity is assumed to be taking place in a given time segment. The probabilistic models implemented include:

- Hidden Markov Models(HMM):
  The use of the Hidden Markov Models (HMMs) to model the user behaviour is described in [38]. Every monitored activity is segmented based on a moving Parsen window of varying lengths. The work emphasizes the advantages of

adding the spatial and temporal information of ADL that show better classification accuracy of the ADL. Another interesting aspect of the work is the implementation of separate HMMs for the activities augmented with the spatio-temporal properties. HMMs are statistical generative models of systems which satisfy the Markov Property condition i.e. the probability of getting into the next state depends only on the current state and not on the previous states. In HMMs the states are hidden but the observations, which are resultants of the state, are visible. It models the joint probability of the observations and states. Formally the HMM can be defined as a 5 tuple model:

$$\Omega = (S, Y, \eta, A, B) \tag{3.1}$$

S represents the hidden states, Y represents the sequence of observations A represents the transition probability, which is the probability from going from one state to another η represents prior probability, i.e. the probability at the beginning of the observations B represents the emission probability which is the probability that a particular observation is in a given state.

After the model is built, the Viterbi decoding algorithm is used to retrieve the states, given the model and a sequence of observations. The pseudo code for the decoding algorithm is shown below: Initialization: Setting the first time slice Recursion: Iterating over time slice $2 \leq t \leq T$ Termination: Which state has the highest probability in the last timeslice Sequence Backtracking: Lookup previous states from T to 1

- Conditional Random Fields(CRF): CRFs are probabilistic discriminative models and given an observation sequence models the conditional probability of the states. In linear chain CRF, as shown in Fig, the states are modeled as a linear chain, with a link between the consecutive chain elements. It is described by the formula:

$$p\,(y|x) = \frac{1}{Z(x)} exp \sum_t (exp(\sum_{k=1}^{K} \lambda_k f_k(y_t, y_{t-1}) + exp(\sum_{h=1}^{h} \mu_h f_h(x_t, y_t)) \tag{3.2}$$

  The parameter to be estimated *lamda* is done by maximization of the likelihood given the data. It can be performed by gradient descent algorithms or Quasi-Newton methods.

Implementation of the probabilistic models has been adapted based on the input formats, which is having an array of sensors which are active at a particular time segment. Based on this array, the calculations have been performed. As a first trial, the matlab toolbox for HMM was used but later had to be modified to suit the data formats. The complete implementation is done in Matlab. The pseudo-code of the HMM implementation is shown below:

- From the processed data, calculate the training data and the corresponding labels of sensors triggered in each time segment
- Count the number of transitions of states and the number of emissions
- Initialize a small value (pseudo transmission and pseudo emission) to avoid zero transition/ emission condition
- Normalize the transition and emission matrices

- Use the viterbi decoding algorithm for retrieving the states given an observation sequence and the trained HMM model

For the implementation of the CRF, the toolbox of Kevin Murphy has been used. The linear chain implementation in the toolbox has been adapted to accept the input in the required format. In the implementation, the scaled conjugate gradient method is chosen for the maximization of the likelihood function. Given that a benchmark for comparison was available, the implementation was tested against the original results. While the methodologies compared are the same, the data has been handled differently in the implementation aspects. Another difference is that the in [1], the quasi newton method was employed for maximization. Results and Discussion:
The following are the results obtained for the two algorithms for the dataset, using the leave one day out approach for testing. The mean accuracy, precision and recall obtained are comparable to the results obtained in [1].

	Accuracy%	**Precision%**	**Recall%**
HMM	83	57	64
HMM	83	45	62
CRF	94	42	76
CRF	96	72	70

TABLE 3.1: Performance Comparison of Inhabitants Activity Data: implementation vs bench-marked data

While it is observed that the CRF has a better accuracy than the HMM, the CRF is computationally intensive and takes a very long time to converge. This study has been useful in learning how to adapt the available toolboxes for graphical problems to real world datasets. These comparable results indicate the validity and the performance of the implementation. This implementation could thus be further extended and tested with other datasets. At the conclusion of this prototype development, we gained an understanding of how to model activity data. In real time, the challenges would be to obtain the ground truth to do a complete feedback learning loop.

Chapter 4

Convergence of Internet of Things and Industry4.0

4.1 Introduction

Industry 4.0 or the fourth industrial revolution refers to introduction of digitilization, automation and the self-learning capabilities into manufacturing. The evolution of the manufacturing sector occurred in the stages moving from mechanization to mass production and automation. We are now in the fourth industrial revolution which aims to build on the third revolution concepts with new innovations from the field of the Internet of Things, cyber-physical systems, robotics, nanotechnology, data analytics, artificial intelligence, 3D printing, autonomous vehicles, biotechnology, cognitive and quantum computing.

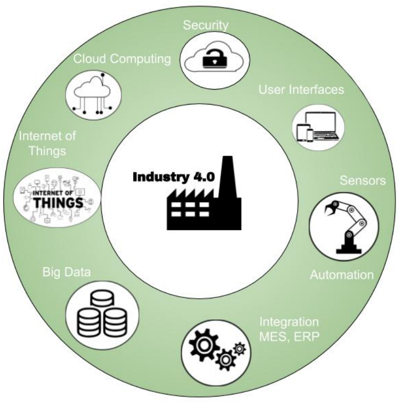

FIGURE 4.1: Industry4.0

In this section, we talk about the Industry4.0 adaptation in the field of photovoltaics and solar cell manufacturing, an overview shown in Figure 4.1. The concepts of Industry 4.0 are still in an early phase and are being slowly integrated by the companies. One important aspect refers to the interconnection and communication of the machines and personnel. The enablers include sensors, actuators, gateways and networking infrastructure. In addition there are IoT platforms, algorithms and data analysis components that support the enablers in analysis to improve the manufacturing process. This paves the way to model real world knowledge to gain insights into the condition, use, performance, and location of the factory ecosystem for improved decision making, predictive and adaptive maintenance, and advanced process control. Outfitting production units with sensors/actuators and incorporation of all machine and material related data leads to the possibility of defining a digital twin of the manufacturing equipment. In this work we extend the adaptive architecture to working towards the goal of realizing a smart self-learning solar cell production unit. The work is carried out in collaboration with the International Solar Energy Research Center, Konstanz, Germany.

4.2 Photovoltaics - a brief overview

Photovoltaics refers to the science of converting solar radiation to electricity using semiconductors exhibiting the photovoltaic effect. The value chain of solar chain can be described in three main steps: Assimilation and treatment of silicon to form silicon wafers, production of solar cells from the processed silicon wafers, production of solar modules with the solar cells. The value chain can thus be depicted in Figure 4.2.

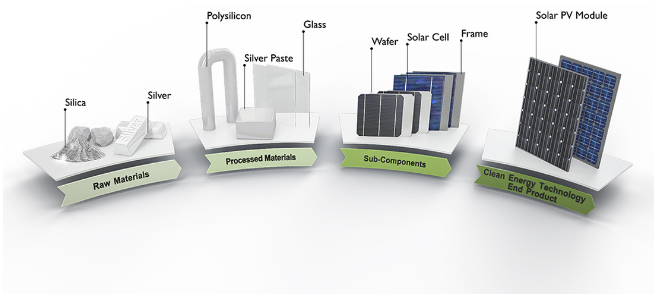

FIGURE 4.2: PV Value Chain: add source

Solar cell production involves a multiple sequences of process steps with complex interrelationships (Figure 4.3) and with automation increasingly used in the larger manufacturing facilities. As is the trend in the manufacturing, every sector is now moving to the integration of Industry 4.0 in order to improve the overall throughput and factory efficiency. The data relevant to this sector include production data, equipment utilization, material management, recipe management and maintenance related data.

4.3 Convergence of IoT frameworks and Industry4.0

In this section we explore concepts of industry 4.0 and extend the adaptive cognitive framework to fit the requirements of a manufacturing line. The study will focus on the following aspects:

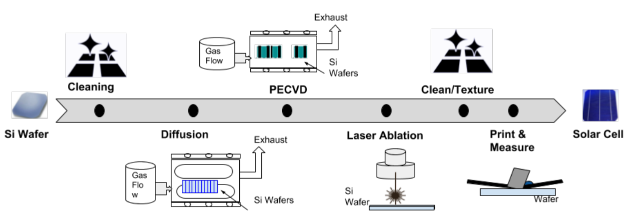

FIGURE 4.3: Solar Cell Production Steps

- Adaptation of the platform which refers to a software suite which will enable to bring all the data from the various equipment. There are various platforms available in the market, some beginning to focus on industries. These platforms are studied and a report on the strengths and differences in the platforms are presented

- Equipping the solar production equipment with sensors to monitor system performance by real time analysis of the process parameters based on the data collected. Eg. production facilities can leverage the data to implement predictive preventive maintenance to preempt failure, correlate events to improve the overall factory yield

- Definition of the virtual objects or digital twin in order to leverage on the virtual model concept for simulations and improved data analytics

- Real World Knowledge Modeling of the data, starting with identifying the most relevant data, modelling and application of the various statistical and machine learning based algorithms to leverage on the data. Here the possibilities to define applications with requirements for self-adaptation are defined.

- Identify the key performance indicators to test the improvements/gain in a fab e.g. Yield, utilization, overall equipment effectiveness (OEE)

4.4 Architectural View of Industry 4.0 related applications

The following Figure 4.4 shows the requirements of the architecture for digitalization a solar cell research lab. The various layers include equipment layer, an additional communication layer is added to highlight the differences in the communication standards, the service layer includes the learning loops and the security layer. The architectural concepts of virtualization, object representation and the real world knowledge generation and analytics are retained. The platform will retrieve all requisite data and hide the technological heterogeneity among the various equipment through APIs (Application Programming Interfaces). At the real world knowledge level, the following potential use-case scenarios are identified:

- Quality assurance to ensure continued high performance modules: Inputs from various sources including efficiencies (IV characteristics) of cells and module, other influencing parameters such as used materials, process parameters (Shingle, laminator), environmental factors (temperature, humidity, pressure), EL images, HIPOT-Data, etc., will be used to analyse the criteria for the optimum module, assess performance changes with parameter differences. Statistical models including regression models, principal component analysis (PCA) amongst others to be used for performing such analysis.

- Root Cause Analysis: During the ramp-up phase, use multivariate models with the above mentioned parameters, to isolate the source of variance. The models will then provide a probabilistic output of the source of a poorly performing module (e.g. issues with Cells, stringer, connectivity, lamination, diode/electrical failures, optical degradation).

- Predictive Maintenance: Factors that lead to frequent maintenance issues: load, environmental factors, usage history to improve the MTBF (Mean time between Failures) estimates based on statistical models.

- Detect Outliers/Anomalies in equipment parameters: Anticipate errors in the equipment leading to early detection of failures by modelling (Artificial Neural Networks) the normal process behaviour and thereby having early warnings to abnormal patterns

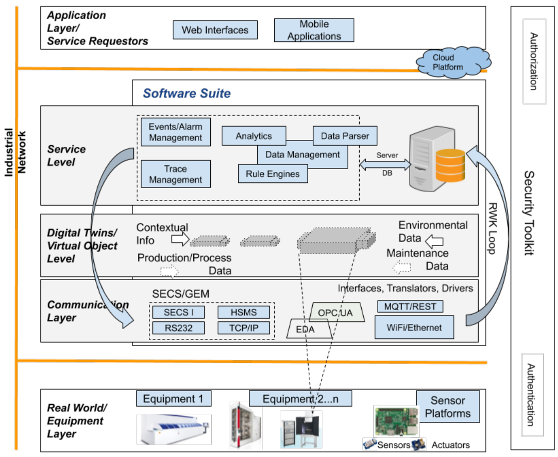

FIGURE 4.4: Positional Architecture for the industry4.0 application in solar production facility

The architectural layers are explained in the following section:

1. Real World Objects / Equipment Layer
From an industry point of view, the real world comprises of different production equipment and associated sensors. It would also include other objects such as personnel, material, etc. The equipment are vendor specific and each having its own communication protocol. There is a strong motivation to adopt standards to unify the access methodologies to the various equipment. The following presents the list of equipment chosen from the production line to be integrated to the architecture.

    - RENA InLine [1]: The Rena Inline wetbench has the following functionalities: (i) InTex system removes surface damage induced by the sawing process and improves the light trapping into the cell by texturing the surface of the wafer (ii) InOxSide ensures efficient junction isolation and PSG removal for high throughput solar cell production (iii) InWaClean removes slurry and sawing residues from separated wafers in a physical-chemical process The RENA production equipment is GEM compliant and therefore has the SECS/GEM interfaces to interact with a host software. We communicate through ethernet via the HSMS communication protocol of the SECS/GEM. The information from the equipment is also made available. The equipment publishes data through the MQTT protocol to a database hosted on a server. A bi-directional control to the equipment is currently not implemented.
    - Diffusion Furnace and PECVD– Centrotherm [2] The diffusion furnace is used for Diffusion, Annealing, and Wet and dry Oxidation processes which are carried out in sustained high temperatures and at moderate vacuum.

[1]http://www.rena.com/de/solutions/category/solar/
[2]http://www.centrotherm.world/technologien-loesungen/photovoltaik/produktionsequipment.html

The PECVD (Plasma Enhanced Chemical Vapor Deposition) is used for passivation and anti-reflective coating processes in the manufacturing of c-Si solar cells. The diffusion furnace of centrotherm are GEM compliant and thereby uses the SECSGEM protocol.

- IV – Halm[3] The Halm IV curve tracer measures the IV curves of PV cells or modules. IV curves are measured by connecting the PV specimen to a varying load with features of dark curves, advanced analysis, and reliability Voc and Isc measurements. It allows hysteresis measurements within one single flash, which allows measuring of high- efficiency cell technologies. Halm IV supports various protocols and provides interfaces for Digital IO, Profibus and Profinet for PLC and XML, SQL and SemiPV2 for MES.
- Sensors + Rasberry Pi In keeping with the Industry4.0 concept, the lab/factory premises will be equipped with sensors to provide more contextual and operational conditions data. We use the Raspberry Pi platform with temperature, pressure and humidity sensors. Rasberry Pi is a small credit card sized computer with a Linux operating system. The Rasberry Pi [4] communicates over Wi-Fi/Ethernet. The sensors connected to the Pi publish data via the MQTT protocol. MQTT stands for MQ Telemetry Transport. It is publish/ subscribe, simple and lightweight messaging protocol, designed for constrained devices and low-bandwidth, high-latency or unreliable networks. Currently, Raspberry Pi with temperature, humidity and pressure sensors are deployed in our labs. Bosch BME280 [5] with the integrated temperature, pressure and relative humidity sensor is used. Figure 4.5 shows a sample screen shot of the sensor data from a lab, showing the trends in the measured values.

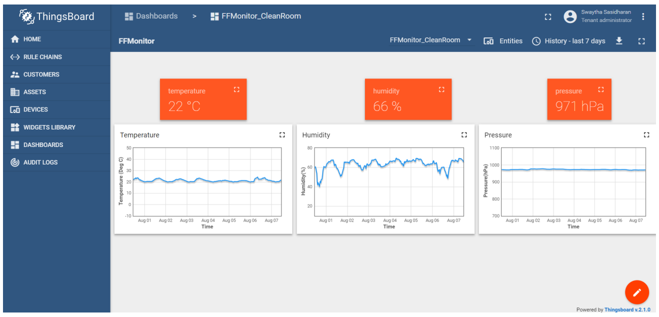

FIGURE 4.5: Sensor Data Measurements in the lab

2. Software Suite The software suite represents a software platform that will enable users and operators to manage, control and access the devices for relevant information.Integration of IoT/industrial platforms within the architecture will be crucial as it reduces the application development time and becomes easy to adapt to changing requirements. Existing IoT platforms/Industry4.o platforms was compared in order to leverage on the large resource pool already available and will build on improving and customizing it to our needs. There are a number of such platforms in the market with some being tailored to fit the Industry4.0 requirements. A lot of the industrial companies have started

[3]http://www.halm.de/
[4]https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/
[5]https://www.bosch-sensortec.com/bme280

developing their own platforms, which are tailored to fit specific requirements based on the industry. While many of them are open source for development and testing, as the system complexity and the communications and data volume increases, the platforms charge a fee for the same. In brief, we present a few of the platforms, but there are several market solutions with different merits. The choice of the platform depends on the cost, complexity, implementation scope, integration of modules and feature support.

(a) Predix : Predix [43]is the Industrial Internet Platform developed by General Electric (GE) Digital and its partners. It specifically targets industries and empowers organizations to develop, deploy, and operate industrial applications. It provides connectivity services, data management services, analytics framework, visualization services and security. The platform is available for initial test, development and deployment. As the usage and number of equipment interaction increases, the payment plans change as per the requirements.

(b) Kaa Platform : Kaa[31] is a multi-purpose middleware platform for the Internet of Things that allows building complete end-to-end IoT solutions, connected applications, and smart products. It is completely open source and has device, data managements, security and is built to made integrations and adaptations easy. It supports applications in various domains including Industry 4.0 applications such as industrial systems automation, predictive maintenance, and remote monitoring. It is hardware- and transport-agnostic and easily integrated with a broad variety of sensors, controllers, machines, and device gateways, enabling many-to-many interoperability between them. Through this interoperability and unified data sharing, separate parts of the production line become more responsive and capable to instantly react to different events or change their configuration settings accordingly.

(c) Eclipse IoT Eclipse IoT[12] provides the technology needed to build IoT Devices, Gateways, and Cloud Platforms. It is open source and built with various partners including Bosch, Redhat, IBM, Eurotech, Fortiss etc. There are various projects part of the Eclipse IoT working towards providing open source solutions such as - Kapua - a modular platform providing the services required to manage IoT gateways and smart edge devices - Hono - a uniform API for interacting with devices using arbitrary protocols, as well as an extensible framework to add other protocols - Mosquitto, Paho- an implementation of an MQTT broker - Kura - a general purpose middleware and application container for IoT gateway services - 4diac - an industrial-grade open source infrastructure for distributed industrial process measurement and control systems based on the IEC 61499 standard. 4DIAC is ideally suited for Industrie 4.0 and Industrial IoT applications in a manufacturing setting

(d) Thingsworx ThingWorx Manufacturing Apps[54] readily connect to Kepware and other OPC servers to provide instant connectivity to your various assets and deliver role-based real-time visibility into operations. Apps are built on the ThingWorx industrial IoT platform and can be extended using ThingWorx native Industrial Connectivity, Analytics and Studio capabilities. This allows for accelerated time-to-value and the ability to quickly kickstart your Manufacturing Journey.

(e) Thingsboard It [53] is an open-source IoT platform for data collection, processing, visualization, and device management. It enables device connectivity via industry standard IoT protocols - MQTT, CoAP and HTTP and supports both cloud and on-premises deployments. ThingsBoard combines scalability, fault-tolerance and performance so you will never lose your data.

Currently, the open source platform, Thingsboard[6] , is used for the implementation of the software suite. The software suite encompasses three layers:

(a) Communication layer
This layer handles all the communication exchange between the upper layers and the equipment. Here all the required libraries, drivers, protocols for communication to the various equipment will be implemented.It is a challenge to connect all the production equipment due to heterogeneous use of technologies adopted by each vendor. There is a strong push towards the creation and adoption of open standards in order to establish a common way to access the machines. The Reference Architecture Model for Industrie 4.0 (RAMI 4.0) [23] advocates the use of IEC standard 62541 OPC Unified Architecture (OPC UA) for the implemention of the communication layer. In order to categorize the connectivity level, it has issues the lowest category requirements for every connected object to be addressable over the network via TCP/UP or the IP protocol and the requirement to integrate at the minimum the OPC UA information model.
Some of the standards include:

- PV2/SECSGEM

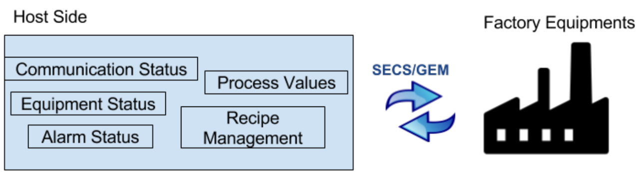

FIGURE 4.6: SECSGEM

SEMI [7]Equipment Communications Standard/Generic Equipment Model standard has been recognized as the accepted standard for communication with PV equipment. It is the interface protocol for data communication between a semiconductor equipment and a host. It is used to retrieve equipment parameters such as operating status, tracking process parameters, recipe data and alarms on a real-time basis. Various useful insight can be gained from these parameters viz. usage statistics, equipment diagnostics and maintenance and real-time monitoring as depicted in Figure 4.6. The standard is the culmination of three separate semi standards:

  - SEMI E5 or SECS-II – communication protocol that defines message structure between equipment and host
  - SEMI E30 – Defines the GEM(Generic Model for Communications and Control of Manufacturing Equipment)

[6]https://thingsboard.io/
[7]http://www.semi.org/en/

- SEMI E37 - High-Speed SECS Message Services (HSMS) defines a communication interface suitable for the exchange of messages between computers in a semiconductor factory
  This task-force set up for the definition of the protocol is the outcome of the efforts of several industries to increase productivity and decrease operational costs. Given that the PV2 protocol is still not an open standard, the adoption rate is considerably slower though it has been adapted to retrieve information in an ordered fashion from the production equipment. As part of the work therefore was also to implement the drivers, interfaces and software on the host side to communicate with the equipment.The following Figure 4.7 represents the GUI implemented based on the SECS-GEM protocol, with the base libraries developed using the open source library defined in [45].

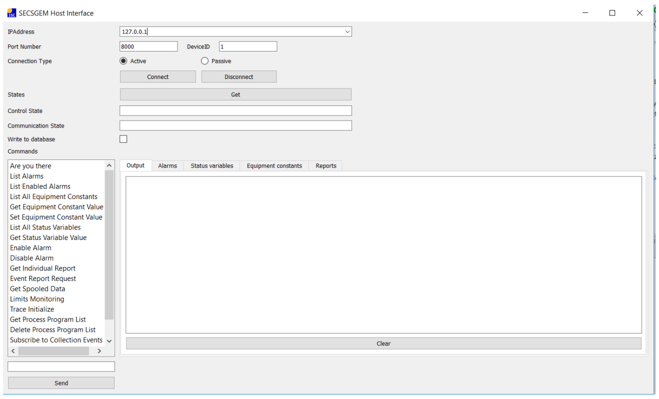

FIGURE 4.7: SECSGEM GUI

- OPC UA[42] It is the open source communication protocol developed for machine to machine communication by the OPC foundation. It is a service oriented architecture , support multiple platforms, has an integrated information model and security features. The biggest advantage of the OPC is its acceptance and use in several associated manufacturing sectors such as automation, robotics, process control and manufacturing.
- Proprietary Some of the machine builders have proprietary protocols defined to communicate.

(b) Virtual Object Layer / Digital Twin
The Virtual Object Layer represents the information models that include all the information coming from the real world. It is enriched with contextually relevant details. Considering the digital twin perspective,

(c) Management Layer
The management layer handles two important functionalities, the equipment and data management. It will act as the interface between the application (Graphical User Interface/ web interface) and the equipment. It covers functionalities amongst others: - Equipment registration to the platform with meta-data - Authentication of the equipment - View equipment information + updates - Equipment restart - Remove equipment The data management layer will be responsible for the collection of the incoming large amount of data and further functions such as filtering and storing

data. This facilitates the application of data analysis methods to retrieve meaningful information. The database requirements for storing the data will be handled at this layer. We are integrating the PostGres relational data management system as the back-end for holding all the data and information coming in from the real world. Figure Figure 4.8 shows a database structure to hold the data coming from the equipment using the SECSGEM protocol.

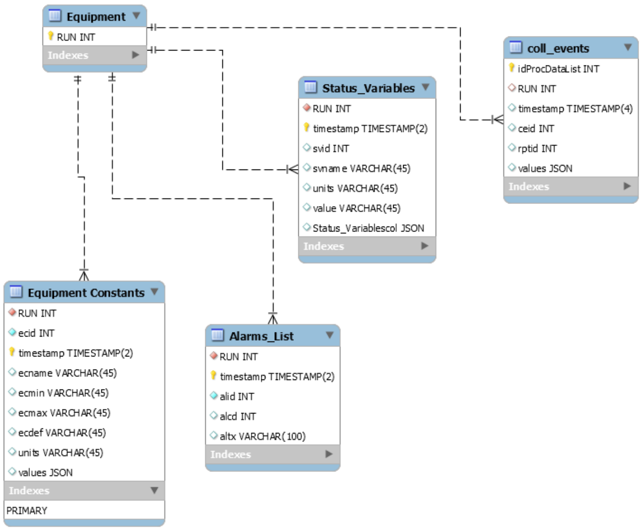

FIGURE 4.8: Data structure based on information retrieved using the SECSGEM Protocol

At this level, we handle the real world cognitive loop, which receives the data from the equipment layer. Statistical models and machine learning models will be integrated with the framework for learning, adapting and improving services.

Some of the features of the platform that have been implemented for our current requirements include: Devices and Assets: Devices include all the equipment/sensors/sources from which data is collected. Figure Figure 4.9 shows a screenshot of the devices connected to the platform. The devices have unique authentication tokens and IDs. They can be assigned to customers and users with specific access rights. Data coming from the devices are stored in the database. The platform provides REST APIs with JWT Bearer authentication to access all the data and meta-data from the platform. Assets can be used to define relationships between the devices. For example, rooms are defined and all the equipment and sensors that are installed in the particular room are linked. Dashboards: Dashboards are widgets or Graphical User Interfaces, used to present the data to the users graphically or as required. The sample data shown in the previous section are dashboards integrated for the current requirements. The dashboards can be customized and users can be provided access rights to view the data. Access Rights: The platform has an administrator and tenants defined. The tenants can add/modify new devices to the platform. Devices and assets can be assigned to clients. Access rights can be controlled by the administrator.

3. Security Layer
The security layer is a vertical which runs across the architecture since there

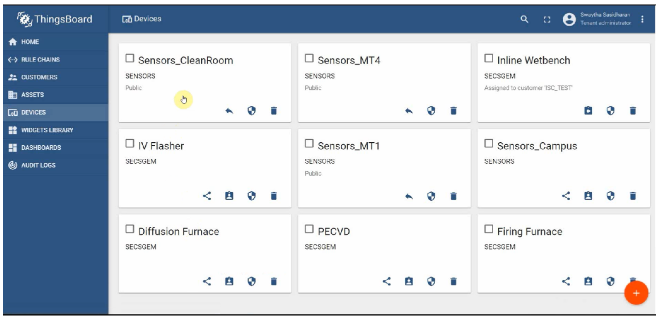

FIGURE 4.9: List of Equipment/Devices

will various features such as authorization with user role definition and access, authentication and certificates validation at the various layers. At the equipment level, the equipment are always behind a firewall and do not have direct access to the outside networks. This is to ensure no malicious software can target the machines. In addition, access rights and roles are defined in order to reach the equipment. At the virtual object level, all the standard authentication mechanisms for REST interfaces with associated security certificates are provided. Access to the database is restricted with most of the users provided only viewing rights. In addition, cyber-security measures should be implemented in order to strengthen the data from being hacked without authorization.

4.5 Case Study: Wafer Tracking - RWK Modelling

The feasibility of tracking wafer carriers in a solar manufacturing facility was carried out at the premises of one of the customers facility. It is a preliminary investigative phase to identify data and pattern requirements which can then be later used to build analytical models and virtual wafer tracking models. Solar cell production is carried out in clean room environments consisting of a series of several defined process steps. During the ramp up phase of a production line, in order to achieve the defined target efficiency, several iterations of parameters tuning need to be performed. These are cost and time intensive cycles. We aim to reduce the number of cycles by observing the data from a production site and optimizations are a part of the process to continuously evaluate and tune parameters in order to reach the target efficiency. Optimizations require cost and time intensive cycles iterated multiple times to identify the source of the problem and make improvements.

A wafer tracking system is essential in (i) quality control of the manufactured solar cells (ii) process optimization (iii) identify causes of performance deviation. The following is a brief overview of the steps involved in the tracking of carriers. A series of RUNs is defined in the production of the solar cells. A RUN is a defined sequence of process steps with pre-defined recipes and parameters. A typical RUN consists of 1000 wafers. Each RUN takes 2-3 days for the completion of all the process steps. These wafers are loaded onto process carriers or transport carriers, with a holding capacity of 100 wafers each. Therefore for every RUN, we require 10 carriers.

The trial RUN was carried out with 1000 wafers. The wafers are tracked through all the process steps until the printing process. Due to the high mixing in the printing sequence, it was not observed for the first run. The wafers were tracked manually/visually based on the initial pattern observations of the equipment. Figure 4.10depicts the wafer in/ wafer out positions. It highlights the wafer mixing that occurs during the various processes.

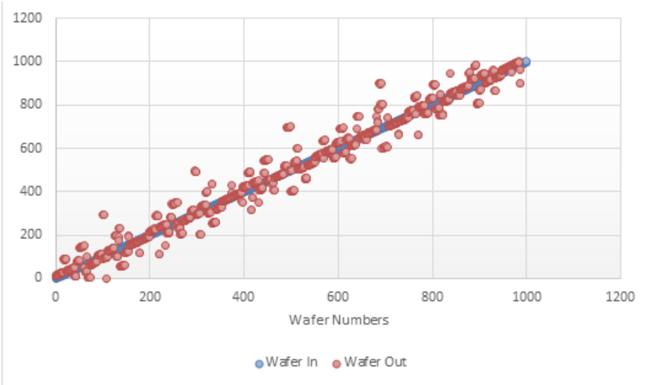

FIGURE 4.10: Wafer Positions in carriers - Beginning vs end of production comparison

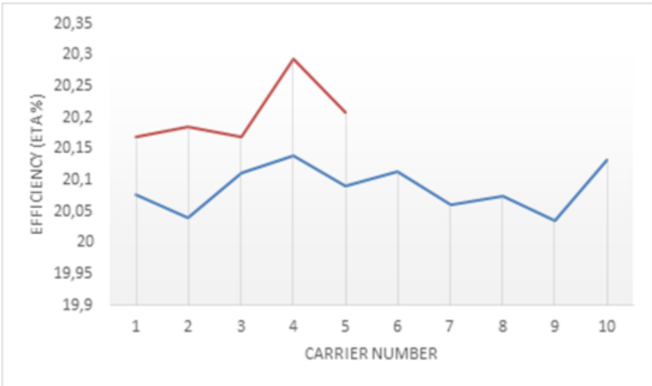

FIGURE 4.11: Wafer Tracking Analysis: Carrier vs efficiency for test run1 and run2

Two Runs were conducted and used to track the path taken by the wafers as shown in Figure 4.11. Tracing the process path of the wafers was observed for inference on which gas zone provides better results. And use this results to improve the parameters of the other zones (temperature, gas flow, etc.). While further tests are required to validate these results.

4.6 Outcome and Observations

The architectural components have been developed and deployed. We have at the application level, a web interface to view and extract data from the real and virtual world. The first step of the digital transformation to retrieve information and make it accessible from one central location is being realized. This facilitates the end-users to extract the data and use it for data analysis using machine learning libraries, statistical tools such as JMP,etc. A use-case highlights the data extraction of parameters and how to identify sources of deviation. Further validation studies will be done and to include a feedback loop.

Chapter 5

Conclusion

5.1 Summary

The initial focus of the thesis lay in understanding the IoT ecosystem and constructing an architectural vision of the same. The need to facilitate the easy integration of connected objects to services is addressed. The need for learning models integrated into the architectural framework is highlighted. While this was carried out with a vision of understanding the methodology to proceed, different architectural elements and understanding was already present.The focus then shifted to the demonstration of the full scope of the architecture through implementations. First we looked at the architecture from a module level, building several prototypes and making qualitative analysis of the same. A smart home prototype to understand the concept of real world objects, virtual object, data management platforms, maintaining historical data and user preferences were studied. The use of data analytical tools was also presented. Moving on we presented modules of a composite virtual objects, which went beyond retrieving values from the sensors were demonstrated through the implementation of an indoor localization system. At the service level, the need for a human user interface and the need for it to be as simple and straightforward was emphasized. Bringing all the elements together, an architecture for the the IoT ecosystem was demonstrated through the smart asset tracking application. We were able to solve the functional requirements of the stakeholders to locate a medical object real time and record object usage statistics. In addition, we could capture object maintenance records, an additional benefit to the hospital authorities. A continuously learning model, which looked for the movement patterns of the object at the room level and provided recommendations to the localization services were performed. Several modules of the code for the implementation has been made available in Github at [4].

The thesis focuses the on the digital vision of a small scale research center. The deployment of sensors was done in the labs maintaining historical records of the data. This represents the real world objects. The addition of production equipment in the real world architecture necessitated the adaptation of communication protocols specific to the semiconductor industry i.e. PV,SECSGEM protocol. Thereby the implementation work on the virtual object layer also included the development of the necessary drivers and interfaces to talk to the machines. We used the open source thingsboard platform for handling the device and data management. Since we have the visualization tools integrated, this was used to display all the data and graphs. With the first phase of establishing connections, retrieving and structuring data in progress, we are now working on the real world knowledge building and the integration of learning algorithms to derive more meaning from the data. A small case study of carrier wafer tracking is also presented.

5.2 Internet of Things - an Interdisciplinary Field - Advantages and Challenges

5.2.1 Introduction

During the first part of the thesis, the work was carried out in parallel to working on a EU FP7 integrated project with partners coming from diverse application domains. Briefly, there were 21 partners from different countries across Europe and Japan coming from different application domains such as ICT manufacturers, telecom operators, software vendors, system integrator, software service/end users [1], SMEs such as of the consortium [2] and research centers [3] and universities [4]. Working in a diverse team lead to interesting discussions to debate, share knowledge and providing an enriching steep learning curve toward the requirements analysis catering to different application domains. The extension of the work carried out in an industrial application exploring Industry4.0 concepts in the solar research laboratory(ISC) further strengthened the ideas and the challenges in creating a cross-functional platforms.

Some of the important concepts related to the inter-disciplinary research include:

5.2.2 Terminologies

In different research areas, the same research concepts are referred by different terminologies leading to ambiguities and sometimes redundancy. For example, the term Internet of Things by itself is ambiguous with various degrees of overlap. There are numerous related terms including Internet of Everything, Cyber-Physical Systems, Ubiquitous Computing, Industrial Internet of Things, Industry4.0, M2M, Ambient Intelligence etc,. While most of the above have some minimum set of features such as connected objects, networking and data analytics. Similarly the sources of data could include people, devices, objects brought to context by additional sensors, etc., These sources have virtual representations of the objects referred to as virtual objects as in this thesis. But in the Industry4.0 the concept of digital twin is elaborated.While the basic concept of virtualization remains the same, the formal representation of the objects differ and requires adaptation as was observed in our work. This also leads to difficulties in standardization. Various task-forces have been formed in order to maintain a consistency and re-usability in the terminologies such as the IoTOne Terms Database [52] and the Industrial Internet's Vocabulary task group [29].

5.2.3 Data Perspective

- Data Governance: Governance can be defined as the system handling the security and privacy, accessibility, relevance and the integrity of the data. Every application domain has its own rules governing the data.When we talk about IoT platforms, where computations can eventually moved to the cloud, and with a lot of intermediate partners and players, regulations and customer-data laws should be defined. It is imperative for the sensitive data to be handled in accordance to the both the business and ethical rules laid down by both the end-users and the service-provides. For applications in the field of smart homes, it refers to sensitive personal information e.g data which tracks the

[1] Siemens, Thales, Alcatel-Lucent, Fiat, Atos, Software AG, NTT and Telecom Italia

[2] Ambient systems, Trilogis, ZIGPOS, Innotec21, SSS and M3S

[3] Create-net, EC JRC, TNO, VTT

[4] University of Surrey,University of Piraeus, Technical University of Delft and University of Genova

movements of the person, understands the behaviour, etc. requires the consent of the user. In contrast to the data from the industries, where the data from the machines which generates data of machines and process which. We therefore require a separation of this sensitive information in the various contexts.The data governance handles screening the data to provide clean and concise rather than highly cluttered databases. It leads to data reliability and consistency. Given the large amounts of data generated, sometimes from multiple sources, it becomes difficult to define clear ownership.

- Data Lifetime: Data is generated from so many sources that a clear definition of the data lifetimes need to be agreed upon. Given that for certain applications, historical data over large periods of time, sometimes running for years is required, certain other applications require data only for a short period of time. Again, the platforms should be capable of defining the data lifetimes. This is a very critical point, else the data and data storage, which is expensive will see an exponential rise. Here, again the ownership and the rights to delete the data should be clearly defined.

- Data Liability: Given that the data will be consumed by different applications and shared across various vendors and maybe even across application domains, it is critical to define the data ownership and thereby the liabilities. While the IoT is striving towards making life simpler, it opens the door to possibilities of personal space infringement, data breaches could occur and thereby a transparent system of the data governance is required in order create a chain of trust between the services and the users.

- Data Storage With no clear standard structure of the data, the challenges are multi-fold. To ensure smooth functioning of several services all accessing the data, decentralized solutions, low cost storage solutions are available. Security, access, connection to data analysis tool are all critical aspects of the data storage that needs to be dealt wit depending on the application requirements.

5.2.4 Human Interaction and workforce

With more and more applications turning smart, handling data from the smart objects is slowly becoming the norm. right from getting advice on whether predictions to complete assistance for elderly , from complete automation to advice on tuning parameters automatically on machine. A technical company now has to include the core competencies of platforms and data analytics. With all these changes happening in a comparatively short time-span, focus now shifts to educate the end-users to be trained to use the technologically advanced applications. It is important for them to understand the value, data usage agreements and the potential threats in the use of the services provided. In order to make this transition of the end-user adaptation, human computer interaction studies should be taken into account. Cultural, national and organizational identities should be accounted for. User design of the end services play an critical role in how well the system can build a cohesive and useful data generation-data consumption model. Users should also be advised on the degree of reliability of the provided services. The change has also resulted in the concept of a digital workspace which is a collaborative working environment which spans geographic boundaries. While the incorporation of technology improves overall efficiency and process control, decreased human interaction is also an outcome which will be challenging to address.

5.3 Future Work

Adaptive models focusing on using the incoming data from the real world, deriving meaning knowledge and providing feedback and recommendations is becoming more relevant. The challenges of obtaining the ground truth and validation of the models at near-real time is still a challenge. This includes the need to study extensively on the change detection and change adaptation modules. With applications such as human behaviour recognition, which are very important in the context of smart homes, assisted living use case, require high adaptability features since patterns tend to change and drift with time.

In the Industry4.0 context, moving towards the self-learning fabrication is the next step. With the transition from a siloed approach to a central point of control, we now have access to a large data-pool. With the right analytical tools, we could generate knowledge to make sense of all the information. Including the prective analytics is the next immediate step of this work and slowly integrating other aspects to slowly transition to a factory which can observe, learn and tune parameters to achieve the optimum working targets.

Appendix A

Appendix A

A.1 Introduction

Internet of Things (IoT) is the technological infrastructure that links real world objects, such as sensors, electrical appliances, RFID tags, computing devices and other IoT compliant devices, to the Internet and has been instrumental in the increase of the number of connected devices, in the order of millions. Some of the enabling technologies include sensing platforms, communication technologies, and middleware. IoT fosters application development in several domains, such as transportation, logistics, health care, smart environments [5]. With all the information coming from these objects, several research efforts have been focused on methodologies to extract knowledge from the raw data so as to improve both the application capabilities and the system performances. For instance, a cognitive framework which provides the core enabling technologies to support IoT, along with cognitive technologies empowering the IoT with self-x and learning capabilities, is presented in [56] and [33].

Given the large number of objects pervading the most disparate environments, one of the most relevant challenges is real-time localization and tracking of such objects, both in outdoor (e.g., navigation) and indoor (e.g., indoor routing) scenarios. In this context, a particularly useful application is assets tracking to facilitate continuous inventory and maintenance. Indoor positioning systems (IPS) are used for objects localization and tracking inside the buildings and/or closed areas [10]. Such applications are particularly useful in hospital scenarios, where it is necessary to inventory all available medical objects for identifying their locations when needed and monitoring their usage for scheduling their maintenance. While an IPS may provide reliable and accurate services, it is an established fact that a ranging task i.e. finding the location of an object starting from distance estimations is expensive in terms of energy and computation. In the domain of IoT, objects are normally battery-powered and hence energy is a critical resource.

Virtualization of these objects opens up the possibility of dynamically controlling their functional parameters, such as sensor power modes, sleep intervals which in turn affects the energy consumption. This paper aims to assess the advantages of controlling the parameters related to ranging operations based on learning the behaviour of objects. Hospitals are sensitive domains and hence setting up an experimental test-bed is not feasible. Here, we present a framework to simulate the scenario of a hospital characterized by a number of devices to be tracked. A knowledge building process is described which facilitates the predictive modelling movement patterns of medical objects, both in the spatial and temporal dimensions. A simulated energy model of the IPS helps to estimate the power consumption necessary to localize the tracked objects. On the basis of the predictions, recommendations to switch the power modes are provided and the estimated energy savings on adoption of the proposed power modes is analytically assessed.

The rest of the paper is structured as follows.Section 1 provides the technological background and energy challenges in an indoor positioning system;Section 2 presents the knowledge modelling framework;Section 3 describes the simulation scenario and the implementation details. Finally, discussion of the results and concluding remarks are presented inSection 4 and Section 5 respectively.

A.2 Indoor Localization Technologies

This section describes the basics of a positing system and the associated energy challenges. [22] presents a survey of the IPS with a brief overview of the available methodologies employed in computing the location of an object. GPS is a widely employed system to locate objects, but given that the signals are attenuated and reflected within buildings, it is not reliable method in indoor positioning systems. Received Signal Strength Information(RSSI) enables to compute the distance between a transmitter and a receiver. This provides a rough estimate of the location of the object. The attenuation factor within the building walls makes this technique unreliable. Various technologies based on triangulation and multi-lateration are used to improve the accuracy. Integration with other sensor data and other technologies such as using optical based, radio technologies [6] [9] or based on acoustics are also employed. In this paper, the simulation studies are based on a radio technology based on the IPS services provided by [61]. The positioning module consists of a sensor, a radio, IEEE 802.15.4 compatible communication unit, phase measurement unit and the associated software components which together compute the accurate position of the object being tracked. IPS uses nodes, which are referred to as anchor nodes, in the proximity to locate the mobile nodes (tags).Anchor nodes are deployed along the perimeter of the rooms of the hospital and are responsible for triangulating the position of a mobile node which is attached to the medical object being tracked. Triangulation of the objects is based on RSSI and angle/time of arrival of the signal. Figure A.1 shows the layout of the hospital with the anchor nodes deployed and the ranging perimeter of one object, as visualized in the GUI of the simulation implemented in this work,where blue circle represents object to be tracked and green boxes represent anchor nodes. The number of anchor nodes required is dependent on the area to be covered and the range of coverage of the anchors. The last room shows an example triangulation scenario in which the location of the object is based on the information obtained from the six anchor nodes. An algorithm then estimates the position jointly from all the measured values. The IPS belongs to the class of wireless sensor networks and hence characterized by constrained resources owing to the small size of the sensing nodes that are typically battery powered. In order to maximize the lifetime of the network of nodes after deployment, energy efficient mechanisms are designed. One of the methods is using a dynamic power management [46] where some parameters such as sleep cycles, data rates, computation, and transmission distance are configured dynamically. It allows the senor nodes to be shut down when no events are observed. This dynamic configuration is not straightforward and requires policies that handle the transitions between the various power modes, wake-up and sleep transition times, *etc.*. Predictive models are used to estimate the workload to control the parameters which leads to energy savings between 10 to 40 percent. In this paper, we propose the use of a dynamic power management method based on knowledge, which is derived from learning patterns of the objects, to reduce the energy consumption. The devices are turned on to the active state when required and turned to sleep mode when not required. Thus the key idea is to control the power management based on anticipated usage. Turning the device to the sleep state(various types of sleep states can be configured) forces the idle listening mode to be turned off, which is power intensive.

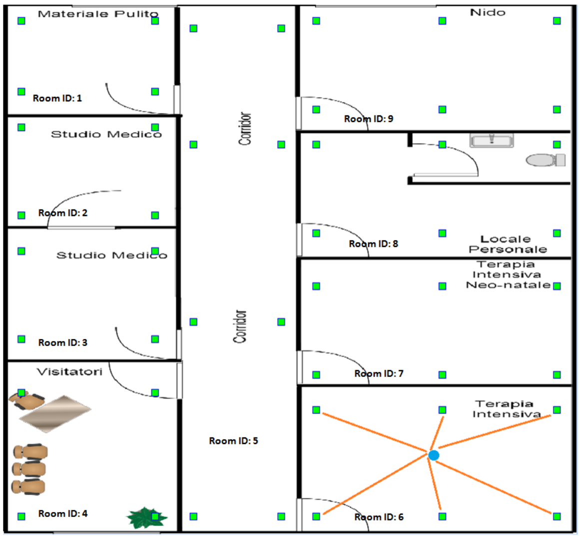

FIGURE A.1: Hospital Floor plan with the deployed anchor nodes

A.3 Simulation Framework

A.3.1 Scenario Description

A.3.2 Real World Knowledge Modelling

With the IoT paradigm as a design criteria, both the anchor and the tracked nodes are virtualized. The virtual counterparts contains the meta-data and other system related information. This virtual representation is exploited to extract and learn patterns based on sensed values in spatial and temporal dimensions, enables to link associations and create aggregations. In this study we limit our scope to the study of aggregation patterns. While virtualization is an enabler, the focus of this simulation is the workflow after the data is retrieved from these virtual objects. The virtualization aspect is described in detail in [34]. Knowledge refers to a set of rules derived on the basis of observation of events occurring within a defined scope of interest. The scope defines the application domain and thus provides the boundary to the events of interest, which are observed from the objects in the environment. [35] presents a service platform architecture in the context of IoT objects facilitating enhanced service creation and execution. It introduces the concept of real world knowledge and real world information(RWInfo). As an extension of the principles explored particularly, in this paper, the defined domain of interest is a hospital with services to locate and track objects. Information from the objects and the associated contextual properties is referred to as RWInfo. This RWInfo represents the statistical data and evidence required for the formulation and acquisition of knowledge.The challenge here lies in obtaining and building the knowledge. This is facilitated by the knowledge models, which are leveraged to provide awareness and thereby improving service and object related performances. Figure A.2 represents the cognitive loop which progressively evolves into knowledge, the knowledge building process. It inherently includes the functionalities of perception i.e. capturing the RWInfo, comprehension i.e. modelling knowledge based on the RWInfo and decision making i.e., actuation based on the derived knowledge. The inner loop is the control loop and the outer loop represents the learning loop. Together, the two loops contributes to knowledge acquisition and application. The learning loop observes the RWInfo coming in from the real world. This information includes sensor readings, meta-data and other related contextual parameters including the spatial and temporal properties of the sensors. In this loop, knowledge is derived based on the historical experience gleaned from the outer loop. The control loop controls actuation and execution of the resultant actions. Knowledge modelling is realized by means of machine learning algorithms. In this paper, we use predictive models which rely on historical data to predict future events. Predictive models are used to predict future events based on statistics, machine learning and data mining techniques. Formally the problem formulation can be represented as: Given a series of data points with a set of m predictor variables x(1,i),x(2,i),.....x(m,i) with an associated set of dependent variable yi, a predictive model establishes the relationship between the predictor and dependent variables to predict the outcome/dependent variable yi, given a new set of predictor variables. The choice of the predictive and dependent variables is dependent on the application.The predictive models are designed based on regression and machine learning techniques. The regression methods include: (i) Linear regression – models the relationship between the dependent and the predictor variables as linear predictor functions. The relationship between the variables is assumed to be linear in nature (ii) logistic regression – is employed to make binary outcomes as predictions (iii) time series models – the incoming data to be modeled is temporally ordered. Time series forecasting techniques predict the future values based on the past observed values. Some of the popular methods of time series modeling include moving average, autoregressive models, hidden Markov models, *etc.*. The machine learning methods popularly employed for predictions include artificial neural networks, support vector machines, Bayesian models and k-nearest neighbor algorithms amongst others [7].

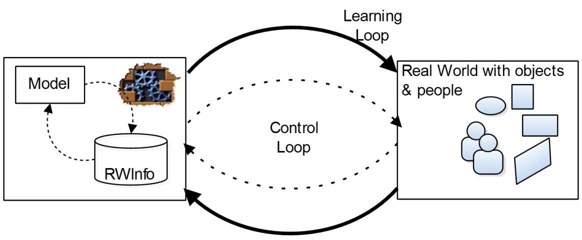

FIGURE A.2: CognitiveLoop

A.4 Simulation

This section provides a description of the simulated scenario along with implementation details.

A.4.1 Scenario Description

A hospital consists of multiple medical objects (e.g., infusion pumps, defibrillators, hypothermia units, portable X-ray devices) which are used on a regular basis, and are often misplaced. Hence medical personnel, who use these objects and maintenance personnel, who service these objects, spend a lot of time and effort to locate these objects and when not found need to be replaced by new ones. To tackle this problem, an IPS is deployed. Details of the positioning system is described below. Tracking applications are power hungry and considerable energy of the battery powered nodes are spent in idle listening mode. Once the IPS is deployed, the configurations are usually fixed. With the IoT capability to connect to each node individually and control its parameters viz. power modes, provides the flexibility to dynamically control the parameters in real-time based on the derived knowledge of the environment. Thus, in order to improve the IPS in terms of performance as well as energy efficiency, the proposed knowledge modelling is applied to the scenario.The knowledge models learn the movement patterns of the objects and provide predictions. This knowledge feeds a simulated energy model of the IPS. On the basis of the energy model, the comparatively better power mode(sleep/active) configuration that ensures higher energy efficiency is chosen. This information is communicated to the tracking sensor nodes, which switch to the desired power mode thus improving the lifetime of the nodes.

Location data, representing the RWInfo, of the tracked objects is logged from the IPS. Data processing and feature extraction techniques are applied on the data. In this implementation, we demonstrate prediction of aggregation patterns. Aggregation patterns refer to the density of the number of devices in the rooms at a particular time. These patterns have a direct impact on the number of active anchor nodes necessary in a particular room. Predictive models are trained with historical data, of object movement patterns, on the basis of which prediction of future patterns is computed. The output predictions include the location, density, usage estimations at a given time of the day. This enables the system to control the power modes of the positioning system (e.g., when the predicted density in a room is low, a few of the anchors can be switched to sleep mode). The energy model is a rule based system that assists in the decision making process to select the number of minimum required anchors without compromising on the accuracy of the position (e.g., given that the a low density of objects will be present in a room, the number of active anchors is set to three and in the event when a high density is predicted, the number of active anchors remains six). With the two loops of the knowledge model, adaptation of the model to the changing patterns ensures a high predictive accuracy.

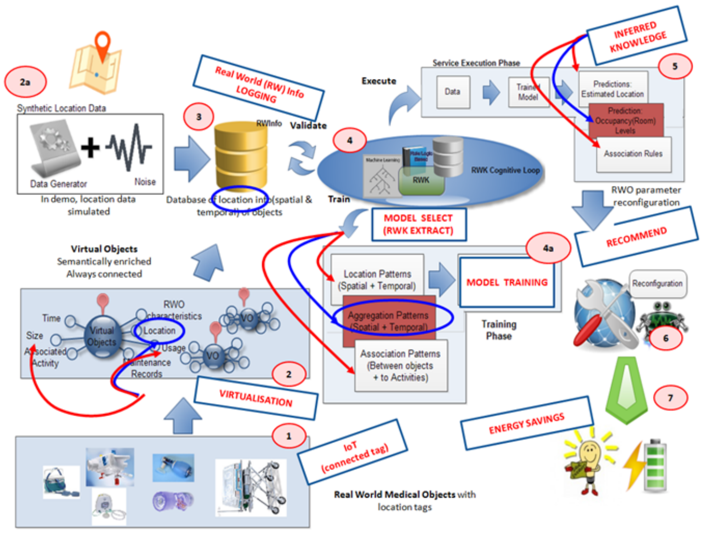

FIGURE A.3: Simulation Workflow

The scenario and the simulation framework is depicted in Figure A.3. The simulation framework is implemented from the step 2a. Step 1 represents all the medical objects and step 2 represents their virtual counterparts. In the paper, this data is simulated as shown in 2a. Step 3 represents the logged RWInfo. Step 4 represent the modelling and step 5 represents the prediction. Recommendations to reconfigure the power modes to save energy are done in the ensuing steps 6 and 7. The main components are briefly summarized:

Data Generation Synthetic generation of data of objects in spatial and temporal frames. The data contains the position of objects, in xy co-ordinates, along with the timestamp. This data is repeated for over 50 times with a randomness factor introduced to distort 20 percent of the patterns. This is done to simulate real scenarios where the occurrence of noise has a high probability. This represents the RWInfo gathering step of the cognitive loop presented in the real world knowledge section.

Knowledge Model Based on the generated data, pattern recognition algorithms are used to model the object movement patterns at both the room level and individual object level. This represents the learning cycle of the cognitive loop.In the current implementation, aggregation patterns, which provides the number of objects in a particular room at a given time, serve as input to the predictive model. Formally,a predictive model can be described as: given a set of input-output response pairsx-train,ytrain , the functional mapping between the variables is established. Based on this relationship, predictions can be made. In the implementation, a linear regression model is used for prediction. Linear regression models the relationship by fitting a linear equation to the observed data. In the real world, there will be differences in the data. Since the simulation is performed with controlled parameters, a linear association between the variable is established and hence the linear regression method was chosen for prediction. The matlab toolbox for prediction has been used to realize the predictive model.

Outcome This represents the control cycle in the cognitive loop. The predicted movement patterns is exploited to improve the performance of the system by providing recommendations to reconfigure parameters of the IPS, for example power modes of operation in order to improve the energy efficiency. The prediction accuracy is checked with the previous historical patterns and only in the case where the prediction accuracy is above 80 percent, they are provided as input to the simulated energy model. The projected energy savings are demonstrated through a simulated energy model of the object tracking system. The simulation framework is implemented in Matlab with a graphical user interface(GUI) to visualize the movement of objects on a floor plan of the hospital. The sequence of steps thus can be summarized as: In the first step, the location data for a configurable number objects is generated for a defined number of time snapshots. The snapshot determines the frequency at which the patterns are to be observed (e.g., snapshots could be taken every one hour). Next, the objects are classified into rooms and the number of objects in each room at a given time snapshot is computed. These represent the patterns that forms the history for prediction of future density patterns in each of the rooms. A custom heat-map is implemented to visualize the density patterns in each room for the various time-stamps. This pattern is then used to build a predictive model. These models can be reused to learn and predict the movement pattern of a single object. In which case the input to the predict model is the generated location data. While this aspect was explored, the concept and performance remained similar and hence the results of only the aggregation patterns are presented here.

A.5 Results and Discussion

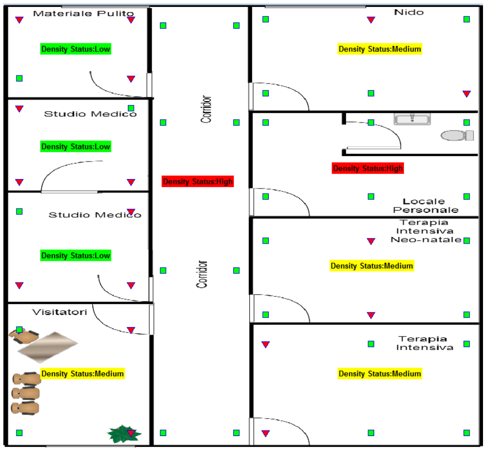

FIGURE A.4: Predicted density of objects in rooms with corresponding power modes of anchor nodes: active(square) and sleep(triangle).Highlighted text indicates density, high(red), medium(yellow),low(green)

The outcome of the simulation is the prediction of patterns specifically density patterns, which is used to configure the power modes of the anchor nodes. The performance of the prediction algorithm is inherently tied to the degree of randomness included in the pattern. For the simulation, a maximum of 20 percent of randomness resulted in consistent predictive accuracy measures of above 90 percent. The predictions are mapped onto the simulated energy model and the estimated energy savings with changed power modes based on the anticipated room density is computed. The density predictions is visualized in the GUI is shown in Figure A.4.

It shows the prediction of the density of objects in every room ('Density Status') of the hospital. Based on the prediction outcomes, the anchors are switched from active to sleep mode (depicted as the green square and red triangles respectively).It is observed that when the density status is high, all the anchors are in active mode, indicating that in such scenarios, the energy consumption remains equivalent to that of the IPS. A graphical representation of the energy savings is shown in Figure A.5 in relative percentages for each room, derived from the simulated energy model based on the object tracking system. The energy consumption in each room is the sum total of the energy utilized by the anchor nodes in the room. There is considerable difference in the energy utilization while in active and sleep mode and hence the relative energy savings, computed as a ratio of the original with the new power mode configurations, is significant. Rooms 5 and 6 in Figure A.5 depicts no energy savings as there are no recommended power mode settings. These results illustrate the validity of our approach to improve energy efficiency in IPS.

A.6 Conclusion

In this paper, an integration of various technologies such as Internet of Things, indoor positioning system, knowledge modelling through machine learning is realized

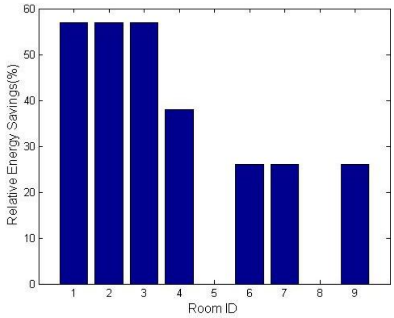

FIGURE A.5: Relative energy savings in the various rooms

in a simulated environment. The cognitive IoT architectural choice facilitates the possibility of building knowledge by observing the real world. Through this, the potential of exploiting knowledge gained by observing patterns in the movement of objects and its value added role in reducing the energy consumption is demonstrated. The simulated framework thus helps to recreate the scenario and visualize the various steps to realize an efficient energy methodology for the positioning system. Future work includes deploying actual sensors and analyzing the energy savings under real conditions. We foresee many challenges in the process specifically in the availability of patterns, removal of noise and ramifications of switching the power modes in an indoor positioning system.

Acknowledgment

This work was supported by the EU Integrated Project iCore, “Internet Connected Objects for Reconfigurable Ecosystems”, funded within the European 7th Framework Programme (contract number: 287708).

Appendix B

Presence Detection in Homes based on Bluetooth Beacons

Abstract

Bluetooth beacons represent a class of inexpensive sensor tags which are increasingly used in localization and thereby location based applications. With the perceived pervasiveness of such beacons in the near future, applications particularly within the scope of Internet of Things are on the rise viz. retail, smart home, smart city contexts. We explore the use of such beacons in a perceived smart home setting. In this work, we have developed a presence detection algorithm using the bluetooth beacon features for use within the rooms inside buildings. The algorithm was tested in a residential setting for a few days and the results are presented.

B.1 Introduction

Internet of Things (IoT) is the technological infrastructure that links the physical world objects to the internet [36]. Smart IoT based applications rely on providing services that improves the quality of life of people providing better access, comfort, and security features. Location information of users plays an integral part in creating contextual applications such as HVAC(Heating ventilation and air conditioning),indoor localization and other location based services. Presence detection thus has become an important aspect to such location based applications. Indoor localization can be realized by leveraging on mobile devices equipped with wireless communication technologies such as Blueooth, ZigBee, Infrared and WiFi [25]. GPS based and cell phone based signal triangulation is an established means of localization for outdoors. Sensor based solutions include the use of (PIR) detectors, RFID based tracking, vibration shock sensors, active systems using infrared, light, ultrasound or microwave sensors. In addition to which video cameras are also widely used. In our work, we explore the use of bluetooth smart beacons to detect presence in rooms, within a residential setting. Beacons are small and inexpensive wireless sensor tags that broadcasts bluetooth signals, on the basis of which contextual actions can be triggered in nearby smartphones. These are popular in the retail sector in order to improve the shopping experience of the buyers. In our work, the scope is limited to realizing and analyzing presence detection based on bluetooth beacons. It is performed with a vision to eventually integrate within smart homes to control home appliances. The particular scenario of interest is a smart heating solution, providing individual room heating control. The existing solutions in the market, such as the hugely popular NEST thermostats[41] provide solutions to control the centralized heating control. It relies on activity sensors to detect movements thereby deriving occupancy details. Another solution in the market, the Honeywell lyric thermostat [26] uses geofencing features to control the thermostat properties. The works [58][57] explore the concept of presence detection using as CO_2 and gas sensors such as the NetAtmo. In our work, the scope is limited to the use of only beacons and involves no other sensors. Leading to the algorithm development, the characteristics of the beacons are studied and the various approaches to presence detection that were considered are briefly explained. Based on the requirements, the geo-fencing and monitoring based functionality methodology was chosen. The algorithm is integrated in an android application. The experiments conducted to evaluate the implemented algorithm is elucidated along with the results. Additionally feedback was collected from the users in order to evaluate its utility to integrate the algorithm to a heater appliance.
The remainder of the paper is organized as follows: Section II introduces the bluetooth based beacons, Section III summarizes the various methodologies considered for development of the presence detection algorithm. Section IV presents the overall architectural framework of the presence detection algorithm with the various required components and Section V provides the results and validation of the algorithm. Section VI provides a brief summary of the user feedback and finally Section VII provides the concluding remarks.

B.2 Bluetooth Based Beacons

It refers to the class of low-powered, low-cost bluetooth based transmitters that notifies mobile devices in the immediate vicinity of their presence. The term iBeacons is an Apple trademark for the indoor positioning devices. But it can refer to the range of devices that supports the low energy bluetooth. These beacons provide proximity estimation where the exact co-ordinates are not provided but a relative location is determined. Estimotes [15], one of the popular bluetooth based beacons is used in this study. It leverages on the features of the low energy bluetooth (BLE) technology. BLE uses the same 2.4 GHz radio frequencies as the Classic Bluetooth, thereby allowing devices with dual modes to share a single radio antenna. However, it uses a simpler modulation system. It uses the frequency hopping spread spectrum(FHSS). This aims to reduce the interference with other wireless technologies using the 2.4 GHz spectrum. Factors that can affect the propagation of the signals include multipath, wave diffraction, absorption or interference. It can decrease the wireless range between devices, decrease data throughput and can also cause intermittent or complete loss of wireless connection. The sources of interference include electrical and electronic objects (operating in the 2.4 Ghz or 5.0 Ghz range), objects which can block the signal transmission or reduce its strength (high: metal, water, concrete, plaster; medium: bricks, marble; low: wood, synthetic materials, glass).

Hardware - Estimote

FIGURE B.1: Estimote Beacons, *image adapted from [15]*

Estimote beacons consists of a Nordic Semiconductor nRF51822. It has a 32-bit ARM Cortex M0 CPU with 256KB of flash and 16KB of RAM with a built-in 2.4GHz radio supporting both Bluetooth LE as well as 2.4GHz operation, where the 2.4GHz mode is on air compatible with the nRF24L series products from Nordic. Estimotes have an advertising range of around 230 feet (70 meters). But the actual range depends on the interferences in the deployment environment. It can also be deployed outdoors and in adverse conditions. Operating beacons in temperatures over 60degC (140degF) or below -10degC (14degF) can reduce battery life. It can be placed in rain but water is a source of interference for the radio signals.The battery is a coin sized lithium ion which lasts for about two years. Low battery exhibits connectivity issues and should be replaced.

Characteristics

The beacons broadcast signals with the following information, all of which are configurable. In order to discover a beacon these properties are to be configured.

- Proximity UUID(16bytes): This property is the identifier of the beacons. The default value of the proximityUUID is the same for all the estimotes, *b9407f30-f5f8-466e-aff9-25556b57fe6d*
- Major value(2bytes): The major property is a value that can be used to group related sets of beacons. For example, all the beacons in a room can be assigned the same major value.

- Minor value (2bytes): The minor property specifies the individual beacon within a group. For example, for a group of beacons in the same room in a house, this value might be assigned to a beacon in a particular location in the room.

The signal characteristics of the estimote beacons are summarized below: (i) Received Signal Strength Indicator(RSSI): It is a generic metric that measures the strength of the received radio signal. It serves as an indication of the power level being received by a mobile device's Bluetooth LE antenna. (ii) Signal Direction: The signals are emitted in all directions (iii) Transmission Power Levels: Ranges from -30dB to +4dB. It represents the physical power of the transmitted signal. Trasnmission power set to maximum (+4) provides RSSI range from -26 (few inches) to -100 (40-50 m). (iv) Advertising Interval: This defines the broadcasting frequency of the radio signals. It can range from low values, such as once per second (1,000 ms), or as frequent as 20 times a second (50 ms). A higher frequency translates to more data that can be exploited by the algorithms to provide accurate location estimates. The advertising interval affects the battery life of the beacons. B.1 sourced from the estimote portal depicts the relationship between the advertising interval, transmission power and battery life of the estimotes.

TABLE B.1: Battery life of estimotes with various transmission power and advertising interval combinations

Advertising Interval	Broadcasting Power		
	-30dB[low]	-4dB	+4dB[high]
2000ms	3.3years	3years	2.3years
1000ms	1.9years	1.7years	1.3years
600ms	1.2years	1years	300days
200ms	160days	140days	104days
50ms	40days	35days	26days

Functionalities

The beacons has two functionalities namely Monitoring and Ranging. A number of applications are built around these two features.

Region monitoring A region of interest similar to a geo-fence boundary is defined. The region is specified by the UUID, major value that differentiates between beacons within the same proximity UUID and minor values, to differentiate between beacons with the same proximity UUID and major value. The region monitoring functionality triggers events when a user enter/exits the defined region. Application can utilize this feature to show alerts or provide contextual information as a user enters or exits region. A single region can contain multiple beacons.

Ranging The determination of distance between two entities is called "ranging" providing the relative distance between a device and beacons. The distance is computed based on the measured RSSI signals. The distance of the device can be obtained in terms of proximity and also in meters. The proximity is defined in three zones: immediate, near and far zones. Given that the bluetooth signals are subject to interferences, the estimate of proximity is higher as the receiver(phone) gets closer to the transmitter(estimote).Depending on implemention, devices could probe the signal every second (1 Hz) or 10 times a second (10 Hz). The responsiveness depends on the scanning interval. Results are also dependent on the placement of beacons and whether a user's mobile device is in-hand, in a bag or a pocket. Clear line-of-sight between a mobile device and a beacon yields better results. For consistent ranging the application should be in the foreground while the user holds the device in-hand.

Bluetooth beacon functionality evaluation

Prior to the implementation of the presence detection algorithm, a study was performed to observe the characteristics of the estimote beacons.Figure B.2 shows the RSSI fluctuations when both the phone and the beacons are kept stable. The three error bars correspond to different experiments at different distances. As depicted, the RSSI fluctuates. It was also observed the positioning of the beacons causes changes in the fluctuation levels.

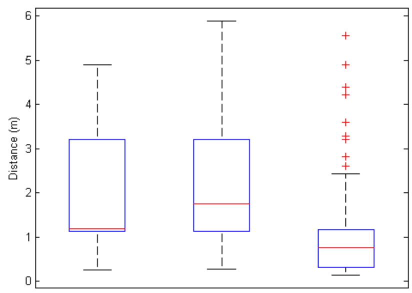

FIGURE B.2: RSSI fluctuations when both the mobile phone and the beacon are kept stable

B.3 Presence Detection Methodologies

An application scenario of smart heating in homes is considered. Smart homes are equipped with different types of sensors and actuators. Given the size and the the portability features of the bluetooth beacons, it is envisaged that these beacons can be integrated with the sensors. The blue tooth 4.0 compliant smart watches are poised to play a big role in the coming year. In a smart home it is desirable to know the location of the residents in order to heat rooms individually, switch lights or even trigger personalized applications. Thus the key requirement that has been evaluated in this work is to reliably detect presence on the basis of the location of the mobile phone. In order to facilitate easy integration with various smart applications, the presence detection algorithm is developed and tested as a mobile application. Another key requirement is the 'easy to configure' feature. Given that smart homes encompasses a wide range of users, it is essential that the initial configuration and set up of the environment should be kept as minimal and simple as possible. The developed application should run as both a foreground and a background process. With these requirements under consideration, three approaches were considered. A prototype simulation of all the three methods were evaluated before choosing the method for implementation.

Ranging and monitoring functionalities of beacon Using the basic features of monitoring and ranging, defined in the previous section, a sample application was developed to observe the trigger of events for a single beacon and the distance measurements were compared to scale. While the monitoring functionality works reliably given that the regions are set correctly, the distance returned shows fluctuations. These fluctuations are due to the fluctuations in the RSSI values. These two features combined held promising results and ranked high in catering to the fulfilment of the requirements.

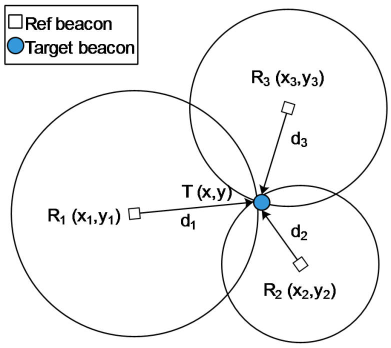

FIGURE B.3: Trilateration Estimation

Lateration algorithm Lateration can be defined as the process of computing the absolute or relative locations of points on the basis of distance measures, using the concepts of geometry of circles, spheres or triangles. Briefly, the distances between reference locations and the unknown location can be considered as the radii of many circles with centers at every reference location. The unknown location is the intersection of all the sphere surfaces as shown in Figure B.3. To use lateration techniques, at least three reference beacons are required.The distinction with triangulation being that angles are also used in the computation. Both are used in navigation applications. (Tri)lateration algorithm employs a distributed beacon positioning and uses distance estimation to compute the 2D position based on the RSSI. In order to compensate for the RSSI fluctutations, a simple moving average filter is used.

A simple co-ordinate system of the room was drawn up, with the location and distance between the three beacons known. Data was collected with a test raster defined i.e.with the user standing at pre-defined points. The basic tests performed showed that the while trilateration could work, it requires a configuration process which requires to be done before the start of the application. This involves defining the room co-ordinates and setting up the thresholds for the individual rooms. This leads to the conclusion that while trilateration algorithm works for a single room, extending to many rooms requires individual configuration in each room which is a tedious and time-consuming process. This does not comply with one of the requirements of our application, which is an easy configuration step which can be performed by all users in a home setting.

Artificial Neural networks Algorithm Localization can also be performed by machine learning algorithms such as artificial neural networks, support vector machines and other statical methods such as the particle filters,etc. Neural networks belongs to the class of machine learning statistical algorithms inspired by the biological neural networks. It is used to provide estimates of outputs based on previously known behaviour. Localization using such algorithms is a two phase process as explored in the works of [2]. The first phase is the training phase, where the rooms are fingerprinted i.e. at known locations the phones are placed and data is collected for a pre-determined amount of time. Data, here represents all the signals that are received by the mobile phone(receiver) from the various beacons. This is an offline process and has to be performed over key multiple positions. Then the neural networks is implemented based on this training data. The second phase is the real-time phase where the trained neural network is fed with the signals from the beacons and the neural networks provides the location estimates.As observed, this methodology is tedious to be implemented where multiple rooms are concerned, as in our case.

Based on the requirements, an easy configuration is of primary importance in addition to reliably detecting room presence. Particularly since these solutions will be deployed in houses and used by people from a wide range of backgrounds. The trilateration approach requires a proper definition of co-ordinates and thereby expert guidance. In the third approach, a training phase is required and doing the same for a big area is very tedious. Given these justifications, this work focuses on the first approach using monitoring and ranging functionalities for the implementation and analysis.

B.4 Presence Detection Architecture

B.4.1 Android Application

The presence detection algorithm is developed using the estimote sdk features of ranging and monitoring. The sdk is open source and available on GIT [14]. An android application is developed incorporating two features:a configuration and set-up phase of the beacons and a presence detection phase. The android application uses the inbuilt SQLite database [48] that is embedded in every android device. SQLite is an open source standard relational database that supports the SQL syntax. Additionally, the online Mongolab database is used as the cloud server. Mongolab is cloud database service that hosts MongoDB databases. MongoDB represents a cross-platform document-oriented database. Mongolab [37] offers REST APIs in order to facilitate easy query and retrieval of the contents of the database. The sandbox version provides free database hosting upto 0.5GB for development and testing and offers flexible plans for deployment purposes.

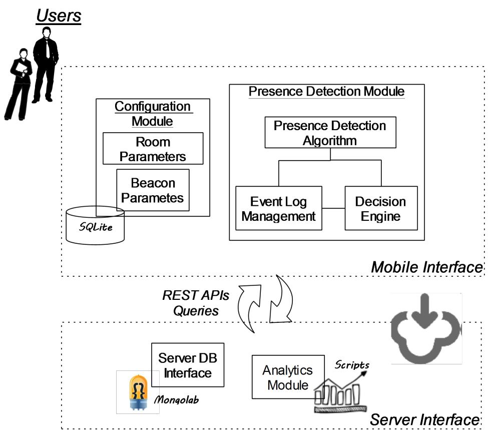

FIGURE B.4: Android Application Process Flow

The overall sequence flow of the android application is depicted in Figure B.4. A configuration phase, which consists of two steps, initializes the application. The first step is the configuration of the room parameters as listed below. They are stored in a SQLite database in the android device itself. If required these configuration parameters can also be stored on the server.

- Number of rooms that require the presence detection algorithm i.e. number of rooms where the beacons will be deployed
- Name of the rooms
- Number of beacons in each room
- Beacon details of each room. In the application, a list of beacon IDs are displayed and it is possible to directly select the appropriate beacon for each of the rooms.

The application provides some pointers to the placement of the beacons and the configuration of the beacon parameters viz. transmission power and advertising interval.

The second step of the configuration, refers to setting the ibeacon parameters listed below:

- Transmission Power: Based on the size of the rooms, the transmission power of the beacons is set. To ensure coverage over the entire room, one of the beacons is set with a higher transmission power and one of more beacons are placed near the entry/exit in order to ensure the event triggers.
- Advertising Interval: Advertising interval in the range of 300ms to ensure faster response of the algorithm. The response also is dependent on the scanning interval parameter set in the phone i.e. how often the phone scans for the beacons.
- Placement of the beacons: The beacons are placed in an upright position on the walls/placed on holders. It is should placed at a height above the normal average height of the people so as to ensure a good line of sight between the beacons and the phone held by the users. This is reduce the interference. It should be noted that water absorbs the bluetooth signals and hence if the phones are placed in the pockets/covered by hands, the performance of the algorithm is degraded. Depending on the size of the room, the beacons are to be placed. For every small and medium rooms, place 2/3 beacons such that at every given point in the room falls within the range of atleast two beacons. Ideal to place a beacon near the door of a room to ensure the events are triggered on entering/exiting a room. Ensure minimum overlapping regions. While minimal overlap is handled, multiple overlaps results in the entry/exit events never being triggered causing the algorithm logic to fail.

Once the configuration is complete, the presence detection feature is initiated.

B.4.2 Presence Detection Algorithm

The presence detection algorithm uses the monitoring functionality of the estimote sdk. Estimotes provides support to both android and iOS mobile platforms. While this study does not provide a comparison of the performance between the two platforms, this application is envisaged to work in both the platforms. The initial step is to define the region of interest. As described previously, the monitoring region is specified by the UUID, major and minor values. In our algorithm, every room has a distinct major value and the minor values are unique. Consider the scenario of a home with three rooms. Rooms 1,2 and 3 are assigned the major values 1001, 1002 and 1003. Each room is equipped with three beacons. The beacons have unique identifiers. Once the presence detection algorithm is called the following sequence of steps is executed as depicted in Figure B.5:

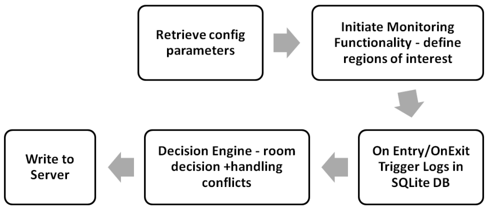

FIGURE B.5: Sequential flow of the presence detection algorithm

(i)The configuration parameters that are set in the previous step are retrieved from the SQLite database(number of rooms, room names, number of beacons in each room and identity of beacons in each room) (ii) Based on these parameters, the regions to be monitored are determined (iii) A start monitoring function is called, which initiates the monitoring functionality i.e. check when entry/exit events for the defined region occurs. It should be noted that multiple regions can be monitored at the same time. And the all events from the monitored regions can be received and processed. (iv) Monitoring Functionality Features: OnEntry Event Trigger: When a user enters a region of interest of beacon in a room, an entry event is triggered based on the signal received from the beacon. This event holds the following information - the id of the beacon which trigerred the event along with the major and minor value. Additionally the proximity and distance information can also be retrieved e.g. immediate zone or 0.5 meters.
OnExit Event Trigger: When a user exits an entered region, the exit event is triggered. This event sends the following information - the id of the beacon which triggered the event and the major, minor values of the region which was exited.

(v) On reception of either entry/exit triggers, the event is updated in the database along with the ID of the beacon that triggered the event, the timestamp and distance at which the event was triggered.

(vi) A decision counter is maintained to facilitate a waiting period to make a decision i.e. which room triggered the event. This is set to a value of 2 i.e. atleast two entry events are required to call the decision making engine.

(vii) Decision Engine: The decision engine retrieves the contents of the database. For example an entry into a room will trigger atleast two entry events i.e. one event due to the beacon placed at the entrance and one which monitors the entire area of the room. The decision engine also retrieves the configuration file details and determines which room the phone is in. In case of conflicts, i.e. equal number of events then the decision is made based on the distance of the triggers. The conflicts arise due to the overlap of the regions. For example, the signal of beacon in room1 will be available in room2 and hence can trigger an entry event of room2 even though the user is in room1. In order to ensure that the correct decision is made, the distance metric is used. The priority is then given to the room with the lesser average distance of the entry event triggers.

(viii) Once the decision is made, the result is displayed as a notification to the user and the information is updated to a server.

B.4.3 Ground Truth Validator

A ground truth validator Android application was developed to obtain the ground truth. This was used during the testing purposes in order to capture the room presence details and thereby use as a comparison metric with the estimated room location provided by the algorithm. The application consisted of icons for entry and exit events for each room. The events are continuously written to the Mongolab server. Additionally the application had provisions to capture user comments during the testing process.

B.5 Validation

The presence detection algorithm has been deployed and the results are summarized as graphs. Two android applications are provided to the user, where both the applications are self-descriptive and the user is asked to follow the instructions. A brief demo of the usage of both the applications is provided before-hand. (i)Presence Detection: Uses estimotes to determine the presence in the room (ii)Room Validator: User should use this application in the foreground and press the relevant icons when entering/exiting a region

FIGURE B.6: Floorplan of the test-bed in house1

An initial test-bed was set up in the premises of the Bosch IoT Lab, University of St.Gallen and the basic functionalities were accessed. The final testing and analysis was carried out in three residential houses. The users were provided with the android application and a minimal explanation of the functionality of the estimote beacons. The following are the results from the three houses. The testing in the first house consisted of three rooms, Figure B.6 represents the floorplan of the house in which the testing was carried out. The blue dots represents the estimote beacon placements within the room. All the tests were conducted in a similar setting given that most of the users were students living in small apartments. For the computation of the accuracy of the estimated presence, scripts were used for the analysis. The data was split into time-bins of one-hour intervals and the number of entry/exit events in a particular time-bin for each room was computed. Using the algorithm, the exit events were always recorded with a delayed time interval. This is an inherent property of the monitoring functionality. This is used to ensure the device has actually exited the perimeter. Due to the mismatch in the time granularity of the events recorded by

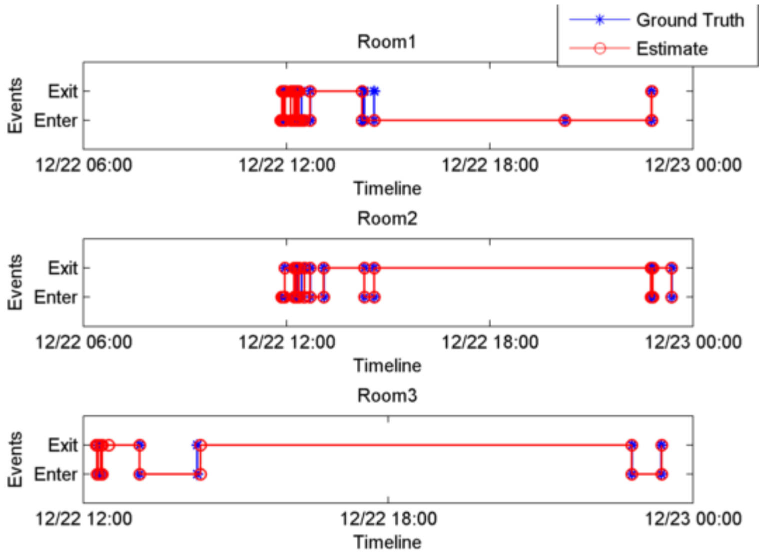

FIGURE B.7: Results of the actual and estimated room presence for one of the test residential houses

the user and the algorithm. Figure B.7 represents graphically the estimated room presence and the ground truth comparison for one of the houses. As represented, we observe that both the actual and estimated values follows the same pattern of events.B.2 summarizes the results of the four houses.It was observed to provide an accuracy of about 90%. This considers a minimal post-processing step where some of the events are discarded.

	Number of Days	Accuracy
House1	3	96%
House2	2	90%
House3	3	88.41%

TABLE B.2: Results of the tests carried out in residential homes

B.6 User Feedback

The experiments also included a simple questionnaire to understand the user experience. The users who participated were all doctoral students and hence had a familiarity of the use of mobile phones and other technical gadgets. The questionnaire evaluations are summarized below:

Feedback on the application usage i.e. set-up process, comprehension and overall usage experience The overall configuration process was easy to understand and thereby easy to set up. The application worked well both as a a foreground and background process on the android phone. With regard to the performance, there were false decisions but within a few seconds, the decision reverts to the correct room. The reason being that sometimes it requires a few seconds to get the correct event due to the order of trigger of overlapping events. Keeping the bluetooth on all the time resulted in a battery drain. There were some missed events with the ground truth verification tool, as some forgot to record the truth during the course of the testing.

Need for individual heater solution and the suitability of beacons solution for presence detection in this scenario The individual room heating was considered a desirable feature particularly to improve the energy savings. The users prefer to control the heaters individually and not tie it to presence. If the heater temperature is lowered when there is nobody in the house, it is a good solution. But when within the house, reducing the temperature of other rooms while in a one room, was not feasible in actual scenarios. This is due to the fact that the heaters take considerable time to reach a particular threshold and the residents usually keep the heaters on most of the time. Given that most do not carry their smart homes around in the house, this technique is not a reliable means to detect presence in rooms. With smart watched gaining popularity, this could be improved. But still would not represent a non-intrusive ideal solution for the heaters.

Estimote Hardware Regarding the use of estimotes, it was considered aesthetic in appearance but the users did not like to have too many beacons around in the homes. It will be preferred if it comes integrated with other products. Given that in the current approach a minimum of three beacons was used in order to detect presence, the users felt it was too many devices to detect presence and thereby expensive.

B.7 Conclusion

Reliable presence detection is a need for most of the indoor IoT based applications. In our work we have explored the use of bluetooth beacons to detect presence. The work provides a quick description of the popular estimote bluetooth beacons. The presence detection algorithm is elucidated and the validation of the algorithm in a residential setting is provided.

Acknowledgement

This work was supported by Bosch IoT lab and the University of St.Gallen, Switzerland.

Bibliography

[1] K. Aberer, M. Hauswirth, and A. Salehi. "Infrastructure for Data Processing in Large-Scale Interconnected Sensor Networks". In: *Mobile Data Management, 2007 International Conference on*. 2007, pp. 198–205. DOI: 10.1109/MDM.2007.36.

[2] Marco Altini et al. "Bluetooth indoor localization with multiple neural networks". In: *Wireless Pervasive Computing (ISWPC), 2010 5th IEEE International Symposium on*. IEEE. 2010, pp. 295–300.

[3] Kevin Ashton et al. "That 'internet of things' thing". In: *RFID journal* 22.7 (2009), pp. 97–114.

[4] *AssetManagementcode*. 2014. URL: https://github.com/iot-icore/icore-smart-hospital-trial.

[5] Luigi Atzori, Antonio Iera, and Giacomo Morabito. "The internet of things: A survey". In: *Computer Networks* 54.15 (2010), pp. 2787–2805.

[6] Paramvir Bahl and Venkata N Padmanabhan. "RADAR: An in-building RF-based user location and tracking system". In: *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*. Vol. 2. Ieee. 2000, pp. 775–784.

[7] Christopher M Bishop and Nasser M Nasrabadi. *Pattern recognition and machine learning*. Vol. 1. springer New York, 2006.

[8] Pricivel M. Carrera and Andrew R.H. Dalton. "Do-it-yourself Healthcare: The current landscape, prospects and consequences". In: *Maturitas* 77.1 (2014), pp. 37 –40. ISSN: 0378-5122. DOI: http://dx.doi.org/10.1016/j.maturitas.2013.10.022. URL: http://www.sciencedirect.com/science/article/pii/S0378512213003332.

[9] Ning Chang, Rashid Rashidzadeh, and Majid Ahmadi. "Robust indoor positioning using differential wi-fi access points". In: *Consumer Electronics, IEEE Transactions on* 56.3 (2010), pp. 1860–1867.

[10] Kevin Curran et al. "An evaluation of indoor location determination technologies". In: *Journal of Location Based Services* 5.2 (2011), pp. 61–78.

[11] Drools. *Drools, The Business Logic integration Platform". JBoss Community*. 2014. URL: http://www.drools.org/.

[12] *Eclipse IoT Platform*. 2014. URL: https://iot.eclipse.org/.

[13] Mica R Endsley. "Toward a theory of situation awareness in dynamic systems". In: *Human factors* 37.1 (1995), pp. 32–64.

[14] *Esitomte SDK for devices with Android OS*. 2014. URL: https://github.com/Estimote/Android-SDK.

[15] *Estimote,Bluetooth 4.0 enabled beacons*. 2014. URL: http://estimote.com/.

[16] Shifeng Fang et al. "An Integrated System for Regional Environmental Monitoring and Management Based on Internet of Things". In: *Industrial Informatics, IEEE Transactions on* 10.2 (2014), pp. 1596–1605. ISSN: 1551-3203. DOI: 10.1109/TII.2014.2302638.

[17] L. Foschini et al. "M2M-based metropolitan platform for IMS-enabled road traffic management in IoT". In: *Communications Magazine, IEEE* 49.11 (2011), pp. 50–57. ISSN: 0163-6804. DOI: 10.1109/MCOM.2011.6069709.

[18] Apache Software Foundation. *MQTT 3.1.1 specification*. 2014. URL: http://mosquitto.org/,https://www.ibm.com/developerworks/webservices/library/ws-mqtt/.

[19] Apache Software Foundation. *Open NLP*. 2014. URL: https://opennlp.apache.org/.

[20] *GIS, Asset Management, Business Intelligence and Location Based Services*. URL: http://www.trilogis.it/.

[21] A. Gluhak et al. "A survey on facilities for experimental internet of things research". In: *Communications Magazine, IEEE* 49.11 (2011), pp. 58–67. ISSN: 0163-6804. DOI: 10.1109/MCOM.2011.6069710.

[22] Yanying Gu, Anthony Lo, and Ignas Niemegeers. "A survey of indoor positioning systems for wireless personal networks". In: *Communications Surveys & Tutorials, IEEE* 11.1 (2009), pp. 13–32.

[23] Martin Hankel and Bosch Rexroth. "The reference architectural model industrie 4.0 (rami 4.0)". In: *ZVEI* 2 (2015), p. 2.

[24] Debiao He and S. Zeadally. "An Analysis of RFID Authentication Schemes for Internet of Things in Healthcare Environment Using Elliptic Curve Cryptography". In: *Internet of Things Journal, IEEE* 2.1 (2015), pp. 72–83. ISSN: 2327-4662. DOI: 10.1109/JIOT.2014.2360121.

[25] Jeffrey Hightower and Gaetano Borriello. "Location systems for ubiquitous computing". In: *Computer* 34.8 (2001), pp. 57–66.

[26] *Honeywell Lyric Thermostats*. 2014. URL: http://wifithermostat.com/Products/Lyric/.

[27] Jonathan W Hui and David E Culler. "Extending IP to low-power, wireless personal area networks". In: *IEEE Internet Computing* 4 (2008), pp. 37–45.

[28] *IoT Figure source*. 2017. URL: http://cdn2.business2community.com/wp-content/uploads/2016/06/internet-of-things.jpg.

[29] *IoT Terms Database*. 2017. URL: https://www.iotone.com/terms.

[30] *IPSO alliance:Enabling the internet of things*. 2017. URL: http://ipso-alliance.org/.

[31] *Kaa Platform*. 2014. URL: https://www.kaaproject.org/.

[32] Tim LM van Kasteren, Gwenn Englebienne, and Ben JA Kröse. "Human activity recognition from wireless sensor network data: Benchmark and software". In: *Activity recognition in pervasive intelligent environments*. Springer, 2011, pp. 165–186.

[33] Dimitris Kelaidonis et al. "Virtualization and cognitive management of real world objects in the internet of things". In: *Green Computing and Communications (GreenCom), 2012 IEEE International Conference on*. IEEE. 2012, pp. 187–194.

[34] Dimitris Kelaidonis et al. "Virtualization and Cognitive Management of Real World Objects in the Internet of Things". In: *Green Computing and Communications (GreenCom), 2012 IEEE International Conference on*. IEEE. 2012, pp. 187–194.

[35] Roelands Marc. "IoT Service Platform Enhancement through 'In-situ' machine learning of real world knowledge". In: *Local Computer Networks Conference* (2013).

[36] Daniele Miorandi et al. "Internet of Things: Vision, Applications and Research Chalenges". In: *Ad Hoc Networks* 10.7 (2012), pp. 1497–1516.

[37] *Mongolab, CLoud DataBase Service*. 2014. URL: https://mongolab.com/.

[38] Mohamed Tarik Moutacalli, Abdenour Bouzouane, and Bruno Bouchard. "Unsupervised activity recognition using temporal data mining". In: *SMART 2012, the First International Conference on Smart Systems*. 2012.

[39] M. Nati et al. "SmartCampus: A user-centric testbed for Internet of Things experimentation". In: *Wireless Personal Multimedia Communications (WPMC), 2013 16th International Symposium on*. 2013, pp. 1–6.

[40] Ricardo Neisse et al. "A model-based security toolkit for the internet of things". In: *Availability, Reliability and Security (ARES), 2014 Ninth International Conference on*. IEEE. 2014, pp. 78–87.

[41] *NEST Thermostats*. 2014. URL: https://nest.com/thermostat/life-with-nest-thermostat/.

[42] *OPC UA*. 2017. URL: https://opcfoundation.org/.

[43] *Predix Platform*. 2014. URL: https://d154rjc49kgakj.cloudfront.net/GE_Predix_Architecture_and_Services.pdf.

[44] Swaytha Sasidharan et al. "Cognitive management framework for Internet of Things:—A prototype implementation". In: *2014 IEEE World Forum on Internet of Things (WF-IoT)*. IEEE. 2014, pp. 538–543.

[45] *Secsgem python*. 2017. URL: https://pypi.org/project/secsgem/.

[46] Amit Sinha and Anantha Chandrakasan. "Dynamic power management in wireless sensor networks". In: *Design & Test of Computers, IEEE* 18.2 (2001), pp. 62–74.

[47] A. Somov, C. Dupont, and R. Giaffreda. "Supporting smart-city mobility with cognitive Internet of Things". In: *Future Network and Mobile Summit (FutureNetworkSummit), 2013*. 2013, pp. 1–10.

[48] *SQLite, Open Source SQL Databse Engine*. 2014. URL: http://www.sqlite.org/.

[49] J.A. Stankovic. "Research Directions for the Internet of Things". In: *Internet of Things Journal, IEEE* 1.1 (2014), pp. 3–9. ISSN: 2327-4662. DOI: 10.1109/JIOT.2014.2312291.

[50] I. Stojmenovic. "Machine-to-Machine Communications With In-Network Data Aggregation, Processing, and Actuation for Large-Scale Cyber-Physical Systems". In: *Internet of Things Journal, IEEE* 1.2 (2014), pp. 122–128. ISSN: 2327-4662. DOI: 10.1109/JIOT.2014.2311693.

[51] Masashi Sugiyama and Motoaki Kawanabe. *Machine learning in non-stationary environments: Introduction to covariate shift adaptation*. MIT press, 2012.

[52] *"Technology Working Group", The industrial Internet Consortium*. 2017. URL: http://www.iiconsortium.org/wc-technology.htm.

[53] *Thingsboard Platform*. 2014. URL: https://thingsboard.io/.

[54] *Thingsworx IIoT Platform*. 2014. URL: https://www.ptc.com/en/products/iiot/thingworx-platform.

[55] P. Vlacheas et al. "Enabling smart cities through a cognitive management framework for the internet of things". In: *Communications Magazine, IEEE* 51.6 (2013), pp. 102–111. ISSN: 0163-6804. DOI: 10.1109/MCOM.2013.6525602.

[56] Panagiotis Vlacheas et al. "Enabling Smart Cities through a Cognitive Management Framework for the Internet of Things". In: *IEEE Communications Magazine* 51.6 (2013), pp. 102–111.

[57] Dominic Wörner, Thomas von Bomhard, and Felix Wortmann. "Occupancy-based heating control for residential buildings using environmental sensors". In: *Proceedings of the 5th ACM Workshop on Embedded Systems For Energy-Efficient Buildings*. ACM. 2013, pp. 1–2.

[58] Dominic Wörner et al. "Look twice: uncover hidden information in room climate sensor data". In: *Proceedings of 4th IEEE international conference on the internet of things, IOT, Cambridge, USA, October*. 2014, pp. 6–8.

[59] Qihui Wu et al. "Cognitive Internet of Things: A New Paradigm Beyond Connection". In: *Internet of Things Journal, IEEE* 1.2 (2014), pp. 129–143. ISSN: 2327-4662. DOI: 10.1109/JIOT.2014.2311513.

[60] L.D. Xu, W. He, and S. Li. "Internet of Things in Industries: A Survey". In: *Industrial Informatics, IEEE Transactions on* 10.4 (2014), pp. 2233–2243. ISSN: 1551-3203. DOI: 10.1109/TII.2014.2300753.

[61] *ZIGPOS,eeRLTS Positioning*. 2014. URL: http://www.zigpos.com/.