# Challenges in Middleware Solutions for the Internet of Things

Moumena A. Chaqfeh and Nader Mohamed
Faculty of Information Technology, UAEU
P.O. Box 17551, Al Ain, UAE
{moumena, nader.m}@uaeu.ac.ae

***Abstract*—The Internet of Things (IoT) aims to interconnect our everyday life items. It provides them with information processing capabilities to enable computers to sense, integrate, present, and react to all aspects of the physical world. This move from "interconnected computers" to "interconnected things" requires simplifying the development of new applications and services by supporting interoperability among heterogeneous devices; so that the programmer can focus on the development of applications enabled by the infrastructure of IoT. Middleware is a software layer interposed between the infrastructure and the applications using it [1] that basically aims to support important requirements for these applications. This paper surveys existing middleware designed for IoT and focuses on various technical challenges in this domain.**

***Keywords*—*The Internet of Things; Heterogeneous Systems; Middleware; Interoperability***

## I. INTRODUCTION

Recent advances in networking will enable the realization of the IoT and pervasive computing visions, since they allow connecting the physical objects of the world to the information technology (IT) infrastructure [2]. This will facilitate the development of a huge number of applications that can significantly improve our lives in various environments currently equipped with "things" with primitive intelligence. By allowing these things or objects to communicate and share information, several applications can be deployed in transportation, healthcare, home, office, plant and social domains. Enabling the move of the Internet from *interconnected computers* to *interconnected things* requires considerable efforts. Therefore, it is feasible to design a middleware system that can simplify the development of needed applications and services.

A Middleware platform for the IoT provides an abstract layer interposed between the IT infrastructure and the applications. It aims to hide the technological details to enable the application developers to focus on the development of the IoT applications [1]. Despite the considerable research efforts in the area of intelligent environments and the IoT, no middleware for pervasive computing has been standardized yet [3]. A single solution that can adapt to all environments will probably not exist. However, there are major requirements and design issues for middleware support in the intelligent environments that have been extensively discussed in the literature [3].

This paper provides a survey of existing research work in designing middleware systems for the IoT. The IoT as an infrastructure and system is surveyed in [1] and [4]. In terms of middleware support, middleware systems for ad hoc networks, robotics, and wireless sensor networks are surveyed in [5] [6] [7], however, no survey could be found for middleware platforms designed specifically for IoT. The contribution of this work is two-fold. First, a classification is provided for middleware systems designed to fit the IoT requirements according to the involved domains. The classification includes are three categories: semantic web and web services, RFID and sensor networks, and robotics systems. Second, the technical challenges of designing middleware systems for the IoT are reviewed. We will show how each domain can support middleware solutions for IoT, by highlighting the list of challenges considered by the approaches surveyed in each of the considered domains.

The remainder of this paper is organized as follows: The second section specifies the technical challenges addressed in the targeted area. The third section reviews existing research work in designing middleware for the IoT, while the forth section provides a discussion on the work reviewed in the third section. The fifth section discuses some open issues, and the last section provides some concluding comments and suggests further work.

## II. TECHNICAL CHALLENGES

The following is a list of major technical challenges that need to be addressed by middleware solutions for the IoT:

### *A. Interoperability*

The IoT represents a huge interoperability challenge for middleware approaches since heterogeneous devices are expected to collaborate together in communication and information exchange. This challenge increases the research effort to design a middleware that can cover a large number of different types of devices, and even new types of devices that may be discovered in the future. The approach proposed in [8] assumes an IEEE 1451compilant sensor device, in which a

significant drawback is raised because not all sensors can integrate that approach. In contrast, the semantic web approaches such as [9] overcome this challenge, since interoperability is a significant advantage of the semantic web technology.

### B. *Scalability*

Since the IoT is expected to support a large number of devices, scalability seems to be one of the major challenges faced by the middleware approaches. This is the result of having thousands of devices that will interact, but fortunately, almost in one place. A reliable IoT middleware is required to effectively manage scalability issues so that the basic functions will operate efficiently in small-scale and large-scale environments [10]. In [9], scalability is one of the proposed approach drawbacks, while in [8], scalability support is an advantage.

### C. *Abstraction Provision*

An ideal middleware for an intelligent environment such as the IoT should provide abstractions at various levels such as heterogeneous input and output hardware devices, hardware and software interfaces, data streams, physicality and the development process.

### D. *Spontaneous Interaction*

In the IoT, spontaneous events are generated due to the sudden interactions that are caused by the movement of things, where new objects are coming into the wireless range of other objects [10]. In this context, middleware is required to manage events in an "arrive and operate" [11] fashion.

### E. *Unfixed Infrastructure*

Unlike the traditional distributed environment, where resources are managed by a certain server, each device in the IoT should be capable of announcing its existence and the resources it provides without requiring a fixed infrastructure [10]. Using a dedicated server for resource management does not hold in the IoT, because of the high distribution and mobility of devices. In this context, a middleware for the IoT should provide automatic discovery of devices in addition to management of resources over different types of services.

### F. *Multiplicity*

Two major multiplicity challenges should be taken into the consideration of the IoT middleware design. First, devices in the IoT are often required to communicate with other devices simultaneously [10]. Second, a device that is participating in an IoT environment is required to select the most suitable services from a massive set of services, because such devices will often rely on services that are available at other nearby devices. In addition, they should deal with the results returned from different services, which may contradict with each other.

### G. *Security and Privacy*

Automatic communication of real-life objects represents a huge challenge in terms of trust, security and privacy. Embedded RFID tags in the personal devices, groceries and even in our clothes can be triggered to respond with their ID and other information. This type of surveillance would affect many parts of our everyday life. The management support of security and privacy has to be considered as a main function of the middleware for the IoT [1]. In SOA-based approaches, for example, the functions related to security and privacy can be either built on a single layer or distributed among all other layers. In the latter case, other issues have to be considered, so as not to affect the system's performance or introduce excessive overhead.

## III. EXISTING WORK

A semantic middleware for the IoT is proposed in [9], which is an extension to the Task Computing Framework proposed in a previous work. This extension focuses on the interoperability at the semantic layer by following three steps. First, semantic services are generated according to the various devices discovered by the middleware. After that, users are enabled to build tasks as service compositions using a semantic user interface, and finally, the task (which is a workflow of services) is completed by executing the devices. This approach has the advantages of semantic web approaches such as interoperability, context awareness to applications, and meaningful information to users. Despite the simplicity and user friendly features of the approach, it assumes devices following non-IP Bluetooth and UPnP specifications, in which its implementation is limited. Moreover, the deployment process uses a template library that maps to some types of devices, and thus, the system cannot accept new devices of an unknown type. Furthermore, the increase in the number of templates will increase the management costs thus the scalability of the system is negatively affected.

A triple space-based distributed middleware for the IoT is proposed in [9] in which the semantic data is expressed by the three items (the subject, predicate and object) defined in the Resource Description Framework (RDF). The service provider is responsible for registering its services in the assumed space, and the discovering consumer would create an invocation and advertise it. Then, the service provider will recognize a new event and retrieve the input data from the consumer and perform the desired service. This approach aims to improve existing triple space middleware that already exists so as to make it suitable for the IoT, such that mobile and embedded devices are allowed to run this middleware. Unfortunately, some devices do not have the capacity to implement the proposed advanced primitives, which can allow for service management and complex queries, and thus, they cannot be part of the assumed Peer-to-Peer semantic network. In the latter context, the usefulness of the proposed middleware and the resulting applications are limited. Nevertheless, the triple space approach seems to perfectly fit

the IoT environment, where several objects are connected to each other so as to interact and share semantic knowledge.

A promising middleware solution seems to be based on the Service-Oriented Architecture (SOA), such as the approach proposed in [12], in which each device offers its functionality as standard services, while the discovery and invocation of new functionalities from other services could be performed concurrently. SOA-based vertical integration has the advantages of reducing the cost and effort required for the recognition of new business scenarios since no device drivers or third-party solutions are required. The proposed architecture supports open and standardized communication through all layers of web services. The layers that can be distinguished in the integrated architecture are: application interface, service management, device management, security and platform abstraction and devices layer. However, performance tests are still required on devices with constrains on resources. Moreover, the proposed approach needs to be extended to support additional devices.

In [2], a vision for middleware for the IoT is described, which forms the basis for the UBIWARE research project that aims to allow the creation of self-managed complex systems by a new generation middleware platform. By utilizing the "agent technology," such middleware will be capable of enabling different components to automatically discover each other and to configure complex functionalities based on the functionalities of those components. The advantages of agent technology include the possibility to allow the mobility of services between different platforms, service discovery in a decentralized manner, utilization of suitable communication protocols, and negotiation-based integration of services. Moreover, interoperability is also possible using metadata and ontologies. In the proposed vision, each connected resource has an autonomous software agent as its representative. The agent is responsible for monitoring the state of the resource, making decisions, discovering the requests, and requesting external help when needed. An adapter or interface is used to connect the resource with its software agent. This adapter may include sensors and actuators, data structures and semantic adapter components as needed. Such a vision of ubiquitous system will allow each relevant component to become a smart resource with self-managing capabilities. UBIWARE is planned to provide semantic communication services and collaboration support services for heterogeneous resources. This will require applying automatic discovery, execution monitoring, communication, negotiation and context awareness. These concepts are mostly related to the semantic web services domain, which seem very promising for the IoT. However, before achieving the objectives of such a middleware, considerable research issues have to be resolved in the design of the agent platform, the representation mechanisms for the distributed resource histories, techniques for information sharing among agents, automatic discovery of other resources in a peer-to-peer fashion, configurability and security.

Another trend of supporting middleware for the IoT is based on wireless sensor networks. TinyREST [13] is an example protocol for integrating sensor networks into the internet. TinyREST focuses on how sensors/actuators and sensor networks can be integrated with the Internet through a framework that establishes home services based on a middleware layer that integrates different sensor network technologies. The current implementation of the TinyREST gateway is a sensor-enhanced middleware for Internet-based access to different types of sensors and actuators that may support different application domains. This approach has the advantage of conforming to the most widespread internet HTTP standard, in addition to enhancing human-device interaction. TinyREST moves a step forward to the full integration into the smart environments test bed, by proving its concepts in home automation and facility management. However, further steps are required for the development of application scenarios to make use of the proposed Internet-integrated sensor network environment.

Another solution based on sensor networking is proposed in [8]. The Global Sensor Networks (GSN) provides a uniform platform for integrating and deploying heterogeneous sensor networks by introducing the Virtual sensor abstraction, which specifies all necessary information required for its usage and deployment. The architecture of GSN follows a container-based model, in which each container hosts and manages a number of virtual sensors simultaneously. These sensors communicate with each other in a peer-to-peer fashion. The identification and the discovery of virtual sensors are made through metadata. The design of the GSN provides four main advantages: Simplicity, adaptively, scalability and light-weight implementation. However, three major disadvantages are present as well. First, human intervention is still required to provide a description for a virtual sensor in an XML file. Second, the proposed solution assumes an IEEE 1451 compliant sensor, which provides a Transducer Electronic Data Sheet TEDS that is stored inside the sensor to provide a simple semantic description of its properties and measurements. This assumption limits the number of sensors that can integrate with the GSN; although a large number of sensors are already compliant to IEEE 1451. Finally, TEDS does not address security, storage and resource management requirements; since it provides only the information required for the interaction with the sensor.

Since the IoT is aiming to create a large wireless network in which every object would have a unique identifier [14], RFID technology seems to fit this requirement by placing an RFID tag on all objects to offer a way for querying them about their identity. The identity of an object can then be used for information retrieval using a name service and shared databases. In [15], a middleware for the IoT is proposed based on RFID technology, relying on three basic functionalities; the tag, the place, and the scenic manager. The first functionality aims to allow for the association of each tag to a certain object, while the second has the objective of supporting the creation and editing of location information

associated to the RFID tag. Finally, the third functionality is used to combine the RFID collected events with the related applications.

Another project based on RFID is the Fosstrak [16], which is an open source RFID platform that is focused on the management of RFID related applications. In [14], the authors propose a general Tag Data Translation (TDT) system that extends the standard of EPCGlobal which only targets Electronic Product Code (EPC). The objective of this system is to provide advanced data translation techniques by integrating a set of existing technologies for identifying items. Fosstrak's implementation of EPC TDT is used in the proposed system as the core of EPC translation. The significant advantage of such a system is that it can offer a way to design a unified architecture of RFID middleware for the IoT encompassing existing useful standards. Nevertheless, integrating more standards is still required to conform to the system objective.

In [3], the potential of a robotic-based middleware for distributed, heterogeneous, sensor-actuator-based, communicating intelligent environments and the IoT has been investigated. A successful application based on two existing middleware architectures from the robotic domain: Play/Stage [17] and Robotic Operating System (ROS) [18], if provided. It has the advantage of supporting heterogeneous devices and interfaces, since this challenge in the robotics domain is very similar in the context of intelligent environments. The authors offer an open test bed for pervasive computing as a cognitive office, which is assumed to represent the intelligent environment.

The office environment proposed in [3] is a one person room that is actively using all kinds of office work. An inclusive list of sensors and actuators were connected via miscellaneous interfaces to the ROS middleware to support a realistic office. Data delivered from these heterogeneous devices is centrally managed by a middleware server instance located in the office. A set of context inference services are implemented on top of the middleware to provide convenient services to the office user, such as the length of day, date and time, weather information, appointments and calendar information and status information. Moreover, computed location information is shared on a public display outside the office for visitors without affecting privacy, since only public abstracted information is displayed, such as "in lecture" or "away for the day". Furthermore, a set of end-user services is also supported, such as travel information and convenient lighting. Visualization tools on top of Twitter are used to support visualization of services for normal users. Despite the advantage of providing abstractions on many aspects such as heterogeneous devices, data streams and physical attributes (e.g. location, context), the office environment is just one example of many other possible intelligent environments in the IoT. Further scenarios need to be investigated to foster research in this direction.

## IV. DISCUSSION

The surveyed middleware solutions aiming to support the IoT can be categorized according to the involved domains into three major categories as shown in Table I. These domains are: semantic web [2] [9], and web services [12], RFID and sensor networks [8] [15] [16], and robotics [3]. This discussion summarizes how each domain can support middleware solutions for the IoT, by highlighting the list of challenges considered by the approaches surveyed in this paper.

TABLE I. CHALLENGES IN MIDDLEWARE APPROACHES FOR IOT.

| **Domain** | | | **Semantic web and web services** | | | | **Sensor networks and RFID** | | | **Robotics** |
|---|---|---|---|---|---|---|---|---|---|---|
| **Approach** | | | *Task Computing Framework* | *Triple space-based* | *UBIWARE* | *SOA approach* | *GSN* | *Fosstrak* | *TinyREST* | *Robotic-based* |
| **Addressed Challenges** | *Interoperability* | | √ | √ | √ | √ | √ | | √ | √ |
| | *Scalability* | | | | √ | √ | √ | √ | √ | |
| | *Abstraction* | I/O hardware devices | | √ | √ | √ | | | √ | √ |
| | | H/S Interfaces | | | √ | √ | | | | √ |
| | | data streams | √ | √ | √ | √ | √ | √ | √ | √ |
| | | Physicality | √ | √ | √ | √ | √ | √ | √ | √ |
| | | Development process | | | √ | √ | √ | √ | | |
| | *Spontaneous Interaction* | | | √ | √ | √ | √ | √ | √ | |
| | *Unfixed Infrastructure* | | √ | √ | √ | √ | √ | √ | √ | |
| | *Multiplicity* | | √ | √ | √ | √ | √ | √ | √ | |
| | *Security and Privacy* | | | √ | | √ | | | | √ |

### A. Semantic Web, and Web Services

A considerable research in the area of middleware support for the IoT is focusing on the semantic web as a promising technology to enable the infrastructure. The semantic web is originally aiming to make information understandable by machines, or things, so that they can perform intelligent tasks based on the meaning of the information, in which the provision of interoperability among devices and information becomes possible. Moreover, semantic web solutions can provide context-awareness to applications, in which the search space for automatic service discovery and composition is reduced [9]. Furthermore, the semantic information supports a better understanding to users so they can compose multiple services, and improve privacy decisions.

Since millions of devices are expected to interconnect and cooperate to provide and consume services, the SOA can offer a promising solution where each device provides a set of standard services, and is capable of consuming services from other devices on-demand. For the IoT, all challenges mentioned in this survey can be addressed by the semantic web and the SOA solutions. In particular, the SOA approach proposed in [12] could overcome all challenges, with the exception of the assumptions related to storage capacity, as shown in Table I. UBIWARE has also addressed similar challenges; however, the proposal in [2] provides a research vision with a description on how UBIWARE objectives can be achieved, but does not offer specific details of the approach, because a set of open issues still require a considerable research work to achieve the project objectives.

### B. *RFID Technology and Sensor Networks*

It is generally recognized that sensors and sensor networks will be a significant part of the IoT [19]. Sensors can monitor the physical world by detecting and measuring different types of environmental information. By feeding suitable applications with such type of information via various types of physical world objects, the Internet would move from "interconnected computers" to "interconnected things." Intelligent context-aware networking is rapidly approaching the position of seamless networking systems, where the development of tiny sensors and actuators can perfectly realize such networks in large factory environments, automotive networks, smart homes and offices, and social services support including earthquake warnings, patient monitoring and context-aware support in emergency situations [19].

The solutions categorized under the domain of sensor networks and RFID technology have addressed the interoperability, scalability, unfixed infrastructure, spontaneous interaction and multiplicity issues as shown in Table I, however it seems more challenging to provide abstraction among I/O hardware devices and hardware/software interfaces in this domain, because a certain type of devices and interfaces are often assumed in the middleware design in this context.

### C. *Robotics*

Various middleware systems have been developed to support intelligent environments such as the IoT; however, this effort does not lead to an evolution of a standardized middleware for pervasive computing or intelligent environments [3]. This is considered as a key issue that limits research in this direction. Although the robotic-based solution in [3] addressed interoperability and abstraction among some levels; it does not address spontaneous interactions as shown in Table I. In fact, it is more challenging to provide such an interaction in the robotics domain, because moderate to high mobility devices are not often considered. In addition, a fixed infrastructure is usually assumed to construct an intelligent environment such as smart homes and cognitive offices.

## V. OPEN ISSUES

A huge research effort is required to make the IoT feasible, since many open issues still persist in this area. For example, the scientific community is offering several attempts to fully standardize the IoT paradigm. Moreover, the large number of nodes which are expected to participate in the IoT forces an effective addressing scheme. Such issues are generally discussed in the literature [2] [4]. This section aims to address the open issues for the IoT in terms of middleware support.

### A. *Standardization*

A single standard for a generalized middleware for the IoT will probably not exist due to the large number and different types of domains and applications involved. However, there are considerable efforts to provide a standardized middleware solution specified for a certain domain. For example, the vision proposed in [2] moves towards a standardized middleware for the semantic web applications domain, while the solution provided in [8] is trying to offer an abstraction for a unique platform for sensor networking environments. For fixed-infrastructure intelligent environments such as smart offices, the solution provided in [3] can perfectly fit the requirements of middleware support in this context. Thus, it is foreseeable that the IoT middleware platforms will have more than one standard to enable applications in different domains. The set of middleware standards that are expected to be offered for all possible applications enabled by the IoT infrastructure can form a complete standardization platform for research and industry. This will enable the selection of the desired standard that fits a certain application within an identified domain.

### B. *User Interface Provision*

It is required to provide an interface between the IoT and the users as part of a middleware support. It is widely accepted that mobile devices will be capable of providing an intuitive bridge as a set of services between the users and the physical objects of the real world [20]. Various approaches are targeting the provision of applications that take such interactions into consideration, but most of these approaches are designed for a special type of application, and they do not address abstraction issues in the description of real world services. Nokia Local Interactions Server is an example for a real-time web service that acts as a back end for RFID-based mobile interactions.

### C. *Storage Capacity*

Smart devices in the IoT are likely to have different capabilities, because the physical world contains miscellaneous types of things [10]. The storage capacity of devices connected to the IoT is another challenge that has to be considered by the middleware designers. For example, the semantic web approach proposed in [9] is difficult to integrate in low-storage devices, as its service management functions and querying system require considerable storage.

Generally, in most of the middleware solutions designed for the IoT, a certain device is assumed to provide the required technology (Hardware). In other cases, a set of devices are identified to operate under the provided platform. Nevertheless, no solution from the reviewed list in this survey is trying to address the storage capacity, such as trying to minimize its requirements. When the IoT becomes available, standardization schemes for middleware solutions will crucially define storage assumptions for different types of enabled applications.

### *D. Security and Privacy*

Security and privacy support is definitely crucial for the functions of an IoT middleware solution. An overview of security and privacy issues in the IoT is provided in [21], while security and privacy standardization issues are specified in [22]. In addition, some security middleware issues in ubiquitous systems were discussed in [24]. The main IoT security requirements are data authentication, access control, and client privacy [23]. Middleware solutions in this context rely on one of two basic schemes to support security and privacy. The first scheme relies on a single layer for providing such support, where security is seen as an ad-on feature of the system. UBIWARE [2] for example, will remain a research prototype without an implementation potential, until an adequate security infrastructure is embedded into it. The second scheme suggests distributing security support among all system layers. Either way, it remains challenging to minimize the overhead of integrating security functions into middleware platforms. For example, authentication usually requires certain infrastructures and servers to enable the exchange of appropriate messages. In the IoT, such approaches are not feasible because passive RFID tags for example cannot exchange too many messages with the authentication servers [1]. That is why security functions have to be provided with minimum overhead.

## VI. CONCLUSION

This paper provided a survey of existing research in designing middleware systems for the IoT. A classification is provided based on the involved domain. It includes three categories: semantic web and web services, RFID and sensor networks, and robotics systems. The technical challenges of designing middleware systems for the IoT include interoperability, scalability, and provision of abstraction, spontaneous interaction, unfixed infrastructure, multiplicity, security and privacy. A discussion is provided to show how each domain can support middleware solutions for the IoT by highlighting the list of challenges considered by the approaches surveyed in each of the domains. Open issues are also highlighted. As part of our future work, we are looking into surveying more projects and approaches to fit them into the presented classification or even introduce a classification from other perspectives. In addition, we need to further investigate the open issues and suggest possible approaches to resolve them.

## REFERENCES


[1] L. Atzori, A. Iera and G. Morabito, “The Internet of Things: A Survey”, Computer Networks, 54(15): 2787-2805, 2010.

[2] A. Katasonov, O. Kaykova, O. Khriyenko, S. Nikitin, and V.Y. Terziyan, "Smart Semantic Middleware for the Internet of Things", in Proc. ICINCO-ICSO, 2008, pp.169-178.

[3] L. Roalter, M. Kranz, and A. M¨oller. “A middleware for intelligent environments and the internet of things,” In Ubiquitous Intelligence and Computing, volume of LNCS, pp. 267–281. Springer Berlin / Heidelberg, 2010.

[4] D.L.Y.F.L. Yi and D. Liang, “A Survey of the Internet of Things.” In Proc. of ICEBI, 2010.

[5] S. Hadim, J. Al-Jaroodi, and N. Mohamed, "Middleware Issues and Approaches for Mobile Ad hoc Networks," in Proc. of IEEE CCNC, Las Vegas, Nevada, USA, 2006.

[6] N. Mohamed, J. Al-Jaroodi, and I. Jawhar “Middleware for Robotics: A Survey,” in Proc. of RAM, Chengdu, China, pp. 736-742, Sep. 2008.

[7] S. Hadim and N. Mohamed, "Middleware for Wireless Sensor Networks: A Survey," in Proc. of COMSWARE, New Delhi, India, Jan. 2006.

[8] K. Aberer, M. Hauswirth, A. Salehi, “Middleware support for the Internet of Things,” In: 5th GI/ITG KuVS Fachgespräch Drahtlose Sensornetze, Berlin, Germany, Sep. 2006.

[9] A. Gómez-Goiri, D. López-de-Ipiña. “A Triple Space-Based Semantic Distributed Middleware for Internet of Things,” In LNCS Vol. 6385, pp. 447-458. Springer, July 2010.

[10] F. Mattern, C. Flörkemeier. From the Internet of Computers to the Internet of Things. Informatik-Spektrum, 33(2), 2010.

[11] K. Paridel, E. Bainomugisha, Y. Vanrompay, Y. Berbers, and W.D. Meuter, "Middleware for the Internet of Things, Design Goals and Challenges", ECEASST Journal, ISSN 1863-2122, 2010.

[12] P. Spiess, S. Karnouskos, D. Guinard, D. Savio, O. Baecker, L. Souza, V. 1736 Trifa, “SOA-based integration of the internet of things in enterprise services,” in: Proceedings of IEEE ICWS 2009, Los Angeles, Ca, USA, July 2009.

[13] T. Luckenbach, P. Gober, S. Arbanowski, A. Kotsopoulos, and K. Kim, “TinyREST: A protocol for integrating sensor networks into the internet,” Proc. of REALWSN, Stockholm, Sweden, June 2005.

[14] L. Schmidt, N. Mitton, and D. Simplot-Ryl, “Towards unified tag data translation for the Internet of Things,” In Proc. of VITAE, pp 332-335, 2009.

[15] E. Welbourne, L. Battle, G. Cole, K. Gould, K. Rector, S. Raymer, M. Balazinska, and G. Borriello, “Building the internet of things using RFID: the RFID ecosystem experience,” IEEE Internet Computing, 13(3):48–55, 2009.

[16] Fosstrak: Open Source RFID Software Platform available at http://www.fosstrak.org/ accessed on April 28, 2011

[17] T.H.J. Collett, B.A. MacDonald, and B.P. Gerkey, “Player 2.0: Toward a practical robot programming framework,” in Proc. of ACRA, Sydney, Dec. 2005.

[18] M. Quigley, B. Gerkey, K. Conley, J. Faust, T. Foote, J. Leibs, E. Berger, R. Wheeler, and A. Ng, “ROS: an open-source Robot Operating System,” ICRA Workshop on Open Source Software, Kobe, Japan, May 2009.

[19] T. Luckenbach, P. Gober, S. Arbanowski, A. Kotsopoulos, and K. Kim, “Tinyrest: a protocol for integrating sensor networks into the internet,” in Proc. of REALWSN, Stockholm, Sweden, June 2005.

[20] S. Siorpaes, G. Broll, M. Paolucci, E. Rukzio; J. Hamard, M. Wagner, A. Schmidt, “Mobile Interaction with the Internet of Things,” in Proc. of Pervasive, Ireland, 2006.

[21] C.M. Medaglia, A. Serbanati, “An overview of privacy and security issues in the internet of things,” in: Proc. of TIWDC, Pula, Italy, 2009.

[22] A. Nilssen, “Security and privacy standardization in internet of things,” in: eMatch’09 – Future Internet Workshop, Oslo, Norway, 2009.

[23] R. Weber, “Internet of Things-New security and privacy challenges,” Computer Law & Security Review, 26:23– 30, 2010.

[24] J. Al-Jaroodi, I. Jawhar, A. Al-Dhaheri, F. Al-Abdouli and N. Mohamed, “Security Middleware Approaches and Issues for Ubiquitous Applications,” Computers and Mathematics with Applications, Elsevier, 60(2):187-197, July 2010.