2012 Sixth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing

Convergence of Utility Computing with the Internet-of-Things

John Soldatos
Athens Information Technology
Research and Education Laboratory in Information Technologies
Peania, Greece
jsol@ait.gr

Martin Serrano*, Manfred Hauswirth
Digital Enterprise Research Institute
National University of Ireland
Galway, Ireland
manfred.hauswirth@ deri.org,
martin.serrano@deri.org (*corresponding author)

Abstract— **Despite recent advances in the IoT area for developing control systems into the cloud, the proliferation of inter-connected entities and the collection of diverse information act as the big challenges in order to provide full control over IoT devices. Solutions to the problem of formulating and managing internet-connected objects in cloud environments i.e. environments comprising "entities" and resources (such as sensors, actuators and smart devices) in cloud infrastructures are still in their infancy. This paper formulates design principles for IoT cloud environments and introduces a framework for the converging utility-driven cloud-based computing models, with emerging Internet-of-Things (IoT) infrastructures and applications into the cloud. Following results from analyzing the state-of-the-art and efforts towards the convergence of cloud computing and IoT, this paper also presents the main building blocks of a framework, which emphasizes on-demand establishment of IoT services based on the automated formulation of societies of internet-connected objects. The framework leverages well-known technologies (i.e. Linked Sensor Data) and standards (notably the W3C Semantic Sensor Networks ontology and the IETF COAP protocol).**

Keywords: Internet of Things, Linked Data, Cloud Computing Utility Computing, Wireless Sensor Networks, Open Source.

I. Introduction

The Internet-of-Things (IoT) [1] is an integral component of the future Internet (FI) and should be seamlessly and smoothly integrated within the emerging integrated services delivery models, such as the Internet-of-Services (IoS) and their associated utility computing paradigms [2], where computing resources are offered as a metered service. However, despite the proliferation of cloud computing models and infrastructures, there is still no easy way to formulate and manage IoT based cloud environments i.e. environments comprising IoT "entities" and resources (such as sensors, actuators and smart devices) and dynamically offering on-demand utility-based (i.e. pay-as-you-go) services. Up-to-date several researchers have described the benefits of a sensor-based distributed computing infrastructure [3][4][5] without however providing a systematic and structured solution to the formulation and management of utility-based IoT environments. Similarly, state-of-the-art participatory sensing infrastructures and services [6][7][8][9], provide instantiations of cloud-based and utility-based sensing services (such as the "Location-as-a-Service" in [9]), without however providing any middleware framework and disciplined approach to deploying and providing such services.

The merits of the convergence between cloud computing and IoT, have given rise to the notion of a «cloud-of-things». Despite the rising popularity of the term «cloud-of-things», the concept is still in its infancy and influenced by different definitions and implementations, including sensors clouds [10][11], internet-of-things platforms [12][13], as well as social networks of objects [14]. These definitions and implementations share common characteristics, yet they are also characterized by differences associated with the physical placement of «things», their logical aggregations and virtualization, as well as their participation in business processes. However, these frameworks do not provide ambient automated orchestration capabilities, in response to end-user requests for IoT services. Furthermore, they lack utility-metering functionalities, which could drive a number of added-value features such as billing, security and privacy.

In this paper we introduce the main building blocks of a middleware framework that aspires to enable the dynamic, self-organizing formulation of optimized cloud environments for IoT applications. This framework will enable service providers to deploy cloud/utility based infrastructures that can deliver IoT services through responding to appropriate end-user requests. It is researched and implemented in the scope of the EU co-funded FP7-287305 OpenIoT project, which aspires to create an open source middleware platform that will enable the delivery of IoT services in a dynamic fashion and according to a utility model. The paper presents the main characteristics and building blocks of the framework, along with standards that will boost its implementation. The structure of the paper is as follows: Section II presents the main characteristics of the OpenIoT framework in the form of design principles. Section III, delves into more details about the specification of service requests to the OpenIoT platform based on the exploitation of attributes defined in the W3C SSN (Semantic Sensor Networks Ontology). Section IV specifies the OpenIoT's approach to the orchestration of internet-connected objects by reference of standards-based approaches for accessing and orchestrating sensors and other devices. Section V describes the use of Linked Sensor Data to facilitate the linking and combination of the data outputs of sensors and other devices. Section VI describes some sample applications that will be integrated towards the validation of the middleware framework. Finally, Section VII concludes the paper.

978-0-7695-4684-1/12 $26.00 © 2012 IEEE
DOI 10.1109/IMIS.2012.135

II. Main Characteristics Of a Cloud Based IoT Framework – Design principles

OpenIoT is developing a framework for the convergence of cloud computing and the internet-of-things (IoT). This convergence needs to compromise the radical differences and conflicting properties of pervasive environments (e.g., sensors and WSN (Wireless Sensor Networks) and cloud environments [15]. Indeed, sensor networks are location dependent, resource constrained and expensive to develop and deploy. On the contrary, cloud computing infrastructures are location independent, elastic and provide access to a multitude of computing resources. OpenIoT will bridge the gaps associated with these differences, since it will allow IoT middleware solutions (i.e. the OpenIoT middleware solution) to leverage the rapid elasticity of data storage clouds in order to store the abundance of sensor data streams that are produced in the scope of large scale deployments. Moreover, computing resources of a cloud could be also used to facilitate stream processing and management (especially in the case of computationally intensive signal processing algorithms. Overall OpenIoT will connect sensors with cloud computing infrastructure, while at the same time providing service-based access to sensor data and resources (notably besed on the REST (REpresentation State Transfer) protocols. The main properties of the OpenIoT middleware platform and related solutions are as follows:

1. **Autonomic**: It will support the dynamic formulation of utility-based computing environments of internet-connected objects, in response to dynamically defined end-users' requests.
2. **Utility based**: It will deliver services according to a utility computing model. OpenIoT applications will be provided as a service (e.g., Sense-as-a-Service) over dynamically created and configured clouds of "things" and according to a "pay-as-you-go" model.
3. **Open Source and Royalty free**: The OpenIoT middleware implementation will be open and royalty free. The project will (re)use software libraries of the Global Sensor Networks (GSN) open source project [16], as a means to interface, filter and combine streams stemming from low-level sensors and internet-connected objects. It will also exploit libraries from the AspireRFID project (wiki.aspire.ow2.org) in order to interface to RFID systems [17].
4. **Dynamic**: OpenIoT will research the dynamic orchestration of internet-connected objects and related resources in the cloud environment. This dynamic orchestration will enable response to dynamically defined end-users' service requests.
5. **Scalable**: The OpenIoT framework will support geographically and administratively dispersed IoT applications. The typical OpenIoT operational environment will be very open and dynamic (e.g., smart cities, community sensing environments, large scale supply chains).
6. **Optimal and Self-Managing**: The cloud environments that will be based on the OpenIoT middleware platform will take into account constraints associated with energy efficiency and bandwidth resources optimization. OpenIoT will incorporate techniques for optimizing these parameters, such as multi-query aggregation [18] and utility-based resource sharing of sensor data [19].
7. **Standards based:** OpenIoT will adhere to popular IoT related standards such as the W3C Semantic Sensor Networks (SSN) for sensors' description and annotations and the IETF Constrained Application Protocol (CoAP) for interacting with nodes and devices in the scope of constrained networks [20].
8. **Interoperable with other IoT architectures:** The OpenIoT platform will enable interfacing and reuse of devices that are part of other popular IoT platforms and architectures, such as the SWE (Sensor Web Enablement) architecture of the Open Geospatial Consortium (OGC) and the Pachube platform [13].

Based on these characteristics, the process of executing a request for establishing an IoT service over the OpenIoT platform involves the following steps: Step 1) Services Definition, (1a) Sensing-as-a-Service definition, (1b) The identification and selection of pertinent objects according to the services objectives and specifications, (1c) Services request; Step 2) The formulation of the service description and its subsequent submission to the OpenIoT cloud environment, (2a) The service composition template, (2b) The service description taking into account the application context, the capabilities of the target internet connected objects (ICO), location, real-time constraints, performance and possibly other non-functional requirements; Step 3) The automated configuration of the internet-connected objects that will be involved in the service delivery, (3a) The objects will configure themselves, so as to form a collaborative society of objects, (3b) The process may involve the exploitation of federated objects providing aggregated and/or consolidated information, (3c) The generation of a mashup-based HMI (Human-Machine Interface) and (3d) The metering of utility parameters that will be used to drive accounting and resource sharing. The figure 1 depicts the described steps and following sections provide further insights on the realization of particular operations on these steps.

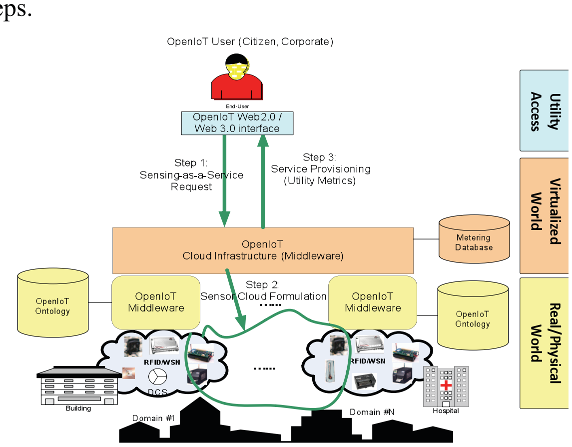

Figure 1. A simple "Sensing-as-a-Service" Scenario.

III. Specification Of Service Requests

Key to the operation of the OpenIoT platform is the specification of the service requests. The goal of OpenIoT is to leverage standards towards defining the attributes of the request in a generic way, which would be later amenable by tools with particular support on specified standards). The Semantic Sensor Networks (SSN) Ontology (specified by W3C) serves as base-line for specifying service requests on the basis of specific sensors, streams and their properties that should be combined as part of a services request.

The SSN ontology is a good example of a coordinated initiative to specify sensor networks structured data exchange. "The mission of the Semantic Sensor Network Incubator Group is to begin the formal process of producing ontologies that define the capabilities of sensors and sensor networks, and to develop semantic annotations of a key language used by services based sensor networks"[1].

The Sensor and Sensor Network ontology presented by the group, known as the SSN ontology, answers the need for a domain-independent and end-to-end model for sensing applications by merging sensor-focused (e.g. SensorML), observation-focused (e.g. Observation & Measurement) and system-focused views. Although the ontology leaves the observed domain unspecified, domain semantics, units of measurement, time and time series, and location and mobility ontologies can be easily attached when instantiating the ontology for any particular sensors in a domain. The alignment between the SSN ontology and the DOLCE Ultra Lite upper ontology has helped to normalise the structure of the ontology to assist its use in conjunction with ontologies or linked data resources developed elsewhere.

While the OGC SWE[2] standards provide description and access to data and metadata for sensors, they do not provide facilities for abstraction, categorization, and reasoning offered by semantic technologies. The semantic annotation method defined by the XG should help the users of OGC standards to retrofit XML-based web services to better support semantic mashups and to ease the integration with linked open data applications relying on semantic web technologies like RDF and SPARQL. The SSN ontology is also the base of the SPITFIRE ontology [21] and is considered in OpenIoT as the base line for semantic sensor networks representation. In parallel activity and as proceed with the SPITFIRE framework implementation, we will take note of each issue that might arise while using the SSN ontology and then, we will propose to the other W3C group members, all the changes we believe are necessary to be applied on the ontology. Finally, given that the SSN is focused on sensing devices and does not cover other ICO in the IoT domain (such as actuators, embedded devices), we will also explore recent works towards turning an SSN ontology into an IoT ontology[3].

[1] http://www.w3.org/2005/Incubator/ssn/
[2] http://www.opengeospatial.org/
[3] http://purl.org/IoT/iot

IV. Orchestration Of Internet-Connected Objects

OpenIoT's ability to create on-demand IoT services will be based on existing capabilities for discovering, linking and orchestrating ICOs. Following early ad-hoc implementations middleware frameworks enabling the development of complete Service-Oriented-Architectures (SOA) over WSN have emerged such as TinySOA [22] and relevant graphical tools [23]. The rising popularity of RESTful services has also led to the development of RESTful access over sensors and actuators [13]. Other works has also focused on the blending of information stemming from the real-world into business processes. The first true implementations (over realistic enterprise systems) have appeared during the last three years [23], and include the blending of sensor events and processes within conventional Business Process Management (BPM) standards [24], [25].

OpenIoT will coordinate ICO, and the main activity to achieve this task is developing methods that enable self-configuration of sensor networks, including mechanisms like bootstrapping, node discovery, service discovery and service brokers' control. In the context of OpenIoT, a key element is to use Semantic Web technologies to implement applications spanning the Internet and the embedded domain. Hence, our goal is to provide all data through so-called RESTful web services i.e. services based on HTTP that follow the REST principles as defined by Roy Fielding [26].

The baseline design conception is that in a wireless sensor network (WSN) resources are addressed using standard semantic descriptions in the format of URIs, for example "http://test.com/sensor", where at the same time offering different representation formats e.g., HTML, XML, RDF, thus using HTTP content negotiation and standard HTTP operations such as GET, POST, PUT, or DELETE are used to interact with this resource and to manipulate its state as it is depicted in the figure 2.

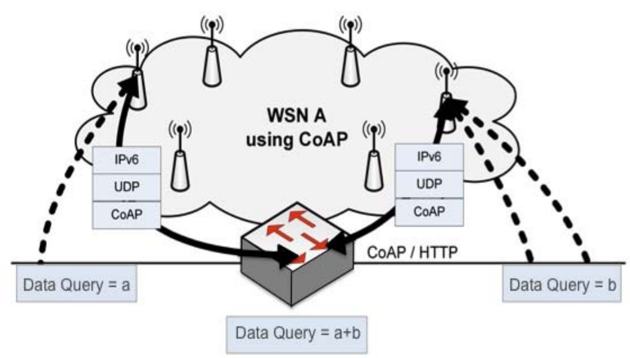

Figure 2. Reach of sensors in WSN by using CoAP

The use of the Constrained Application Protocol (CoAP), specialized RESTful webs transfer protocol for use with constrained networks and nodes [20], will facilitate this objective. We have studied and argue two main use cases for using CoAP, the first implies the use of CoAP in combination with DNS (Domain network Servers) for enabling the use of user-friendly fully qualified domain names for addressing sensor nodes. The second do not require the inclusion of DNS but concentrates on M2M communication to avoid unnecessary traffic and thus energy consumption in the sensor network. Both provide a flexible solution to achieve self-organization with a minimum of pre-configuration.

CoAP emerges from the new IETF CoRE working group specifications, which deal with Constrained Restful Environments. CoAP provides exactly the subset of HTTP methods (GET, PUT, POST, and DELETE) which is necessary to offer RESTful web services in a WSN-compatible manner. CoAP provides optional transport reliability, normally a core functionality of TCP, which is due to the resource constraints by nature not available in WSNs. This is particularly useful, since CoAP is designed to be used in combination with UDP which does not offer any reliability but is adequate for WSNs due to its low impact on resources. In addition to the CoAP protocol there is an optional extension (still in draft status) that offers status observations for CoAP resources [27]. Clients are allowed to register on a service to be informed about the next status change of the resource. The advantage of this new specification, there is no need to keep any connection alive but only to provide an open CoAP port on the client.

UZL and COA have implemented a first prototype version of the current CoAP draft for the iSense hardware platform [28]. In addition, IBBT has implemented a CoAP implementation for the IDRA platfor[4]. The implementations provide a resource- efficient embedded CoAP web server and applications can register handlers for specific URIs. For instance, a simple service could offer the current sensor reading of the car detection sensor or another handler could provide a list of services running on a specific device.

Furthermore UZL has implemented CoAP and an HTTP/CoAP-Gateway application for HTTP to CoAP translation and vice versa in JAVA. All JAVA applications in the following with a need to communicate via CoAP use this protocol implementation. As an example UZL implemented a CoAP Browser for the mobile OS Android to be used within mobile use cases. Within OpenIoT, the use CoAP as part of RESTful web services on sensor nodes is due CoAP blends particularly well with HTTP and a conversion between HTTP and CoAP is easily possible.

Overall, the on-demand creation and deployment of IoT services as part of OpenIoT, will rely on a middleware infrastructure supporting discovery, linking and coordination of ICO. However, OpenIoT will also enable the off-loading and execution of application logic on ICOs in case this is required by the service.

V. Sensor Data As Linked Data

Linked data for information sharing is widely an accepted best practice to exchange information in an interoperable and reusable fashion way [29], over the Internet different communities use the semantic web standards to enable interoperability and exchange information. Recently linked sensor data has been explored in order to enable the interconnection of multiple new mobile services demand by the rapid development of smart technological devices. Taking a broad view about the state of the art on linked data and its applications in WSN, many of the current problems in WSN still remain in the basic notion of exchanging pieces of information. Data exchanges at sensor level result almost impossible due the capacity for processing information is limited and the stream data processing is very complex.

In OpenIoT linked sensor data will be exploited in order to facilitate the linking of sensors and ICO related processes at a higher semantic level (within the OpenIoT middleware), where data exchange operations can be conducted. In order to facilitate this process has been necessary to identify the persistent problems as listed below, in order to be addressed and provide alternative solutions.

1. In WSN only a small numbers of services can be offered, which cannot be personalized to meet dynamic wireless sensor configurations;
2. The offered services in WSN are typically technology-driven and static, designed to maximise usage of capabilities of the sensor network rather as individual services and not to satisfy user requirements per se.
3. The information from sensors cannot be readily adapted to their changing operational context, so sensors cannot communicate their surrounding sensors for configuration purposes, within the objective of changing service usage patterns and rapid new services deployment.

In summary, sensors can only be optimized, on an individual basis, to meet specific low-level objectives, often resulting in sub-optimal operation. By addressing linked data issues, sensor systems can be able to exchange information and thus facilitate customization of sensor networks services.

A way to cope with these requirements and by establishing a WSN design practice in OpenIoT, it is necessary the composition of data models to generate aggregated data streams. The creation of a data model implies sensor data to be endowed with certain level of information flexibility, operability and management control and most important provide the control platform for interconnected sensors [30]. This every day more popular activity focuses in the semantic enrichment task of the information to generate sensor data models with ontological data to provide an extensible, reusable, common and manageable linked data plane i.e. a plane where aggregate sensor data could be manipulated and operated.

OpenIoT supports the use of the cited ontology (i.e. W3C SSN) as the mechanism to generate formal descriptions, which represents the collection and formal modelling for sensor networks. Using a formal methodology the user's contents represent values used in various service management operations, thus the knowledge-based approach over the inference plane [31] aims to be a solution that uses otologies to support interoperability and extensibility re-

[4] http://idraproject.net/

quired in the systems handling end-user contents for more interoperable applications [30]. Beyond of the formal description OpenIoT will endow such models with the necessary semantic richness and formalisms to represent different types of information needed to be integrated in network management operations.

In OpenIoT this design approach will be used for providing tools to integrate sensor data within the service operations, and offers a more complete understanding of WSN contents based on their continuous data acquisition. By using sensor linked data is expected a more inclusive governance of the management of information and control on devices, likewise facilitates networks (WSN), systems and more adaptive services. By linking sensor data information exchange within different devices is pursued.

VI. Simple Validating Applications

In this section three classes of validating applications for the OpenIoT concept are outlined. The three use cases leverage the capabilities of the OpenIoT platform in formulating societies of internet-connected objects, creating services, measuring utility, and ultimately delivering the services in a pay-as-you-go-fashion. These three use cases address two major issues that are not adequately confronted by existing IoT frameworks: (a) Support for large scale deployments and (b) Support for intelligent on-demand utility-driven selection of ICOs

A. E-Science

A validating application in the area of e-Science will use the OpenIoT platform in order to setup a public infrastructure enabling on-demand access to multiple sensors that are available by one or more research organizations (i.e. universities and research centers) wishing to share their data. Such an infrastructure could then provide a range of sensing-related services (e.g., experiments) to researchers and scientists. OpenIoT will alleviate the current situation in the development of wide area sensing applications, where most of the sensor network and data analysis systems are developed in-house and hard-wired to local data sources. The use case will provide an effective collaboration platform among scientists including shared access to online sensor data repositories. Relevant on-demand services to be provided by this e-science platform will involve remote sensing, sharing of sensor network data, access to digital repositories, as well as communication and decision making for scientific experiments. The OpenIoT platform will therefore glue distributed sensor networks, high performance computing infrastructures, as well as scientific instruments to request the on-demand completion of large scale sensing experiments independent of time and geographic location.

B. Manufacturing

The manufacturing use case will make use of the OpenIoT platform in order to provide/setup a private cloud based infrastructure enabling multiple traceability and manufacturing intelligence services over multiple production lines. The services will be used by employees within the manufacturing plant, including plant managers and operators. A specific scenario includes the on-demand submission of performance monitoring requests concerning one or more KPIs (Key Performance Indicators) that are associated with the plant operation. The requests will concern the on-demand calculation of KPIs, associated with the intelligent selection of multiple sensors and internet-connected objects of the plant. The KPIs to be specified in the scope of the performance monitoring requests, will have the following characteristics: (a) They will be composite i.e. they will combine information from multiple devices and/or other indicators (such as materials consumption rates, production rates etc.), (b) They will be virtual and reusable by other KPIs (services), (c) They will be connected to actionable logic and M2M interactions e.g., enabling the configuration of machines in case some thresholds are exceeded, and (d) they could span multiple plants and warehouses.

C. Smart Cities

A Smart City in conjunction with an ICT expert can use the OpenIoT platform to setup a public cloud infrastructure, which could support multiple (rather than a single) services based on internet-connect objects (e.g., smart phones, sensors). Citizens could then use the provided services through submitting requests to the OpenIoT infrastructure and accordingly having them executed by the middleware. Requests will target information that could be provided on the basis of urban sensors, such as meteorological information, traffic information, parking occupancy information, in conjunction with city maps. OpenIoT will offer the ability to automatically define and formulate such requests, while at the same time providing services in a utility based fashion.

VII. Conclusions

The convergence of cloud computing with the internet-of-things (IoT) is associated with a host of opportunities, as well as with several technical challenges stemming from the different characteristics of the two technologies (cloud, IoT).

In this paper we have identified these challenges and introduced the main building blocks of a platform that could support the convergence of utility computing and the internet-of-things. Based on this platform, service providers can request the on-demand formulation of IoT services. The later are accordingly used by end-users in a pay-as-you-go fashion. Among the main characteristics of the introduced system is that its implementation could be based on existing open source platforms, which could be enhanced on the basis of popular and emerging standards for IoT.

The authors intend to establish a relevant open source project, in order to enable community development of IoT services over cloud environments.

ACKNOWLEDGMENTS

Part of this work has been carried out in the scope of the project ICT OpenIoT Project (Open source blueprint for large scale self-organizing cloud environments for IoT applications), which is co-funded by the European Commission under seventh framework program, contract number FP7-ICT-2011-7-287305-OpenIoT. The Author(s) acknowledge help and contributions from all partners of the OpenIoT project.

REFERENCES

[1] Harald Sundmaeker, Patrick Guillemin, Peter Friess, Sylvie Woelfflé (eds), "Vision and Challenges for Realising the Internet of Things", March 2010, ISBN 978-92-79-15088-3, doi:10.2759/26127, © European Union, 2010.

[2] Michael Armbrust, Armando Fox, Rean Grith, Anthony D. Joseph, Randy H. Katz, Andrew Konwinski, Gunho Lee, David A. Patterson, Ariel Rabkin, Ion Stoica, and Matei Zaharia. "Above the Clouds: A Berkeley View of Cloud Computing". Technical Report UCB/EECS-2009-28, EECS Department, University of California, Berkeley, Feb 2009.

[3] Hock Beng Lim , Yong Meng Teo , Protik Mukherjee , Vinh The Lam , Weng Fai Wong , Simon See, "Sensor Grid: Integration of Wireless Sensor Networks and the Grid", Proceedings of the The IEEE Conference on Local Computer Networks 30th Anniversary, p.91-99, November 15-17, 2005, doi>10.1109/LCN.2005.123.

[4] Lina Yu, Xiang Sun, Qing Wang, Wanlin Gao, Ganghong Zhang, Zili Liu, Zhen Li, Jin Wang, "Research on Resource Directory Service for Sharing Remote Sensing Data under Grid Environment", gcc, pp.344-347, 2009 Eighth International Conference on Grid and Cooperative Computing, 2009.

[5] Kapadia, A.; Myers, S.; XiaoFeng Wang; Fox, G. "Secure cloud computing with brokered trusted sensor networks", International Symposium on Collaborative Technologies and Systems (CTS), 17-21 May 2010, pp.581 – 592.

[6] Catherine Havasi, James Pustejovsky, Robert Speer, Henry Lieberman, "Digital Intuition: Applying Common Sense Using Dimensionality Reduction", IEEE Intelligent Systems 24(4), pp. 24-35, July 2009.

[7] Nicolas Maisonneuve, Matthias Stevens, Maria E. Niessen, Luc Steels, "NoiseTube: Measuring and mapping noise pollution with mobile phones", Information Technologies in Environmental Engineering (ITEE 2009), Proceedings of the 4th International ICSC Symposium, Thessaloniki, Greece, May 28-29, 2009.

[8] Page, X. and A. Kobsa "Navigating the Social Terrain with Google Latitude". iConference 2010, Urbana-Champaign, IL, p.174-178.

[9] Marshall Kirkpatrick "The Era of Location-as-Platform Has Arrived", ReadWriteWeb, January 25, 2010

[10] Mohammad Mehedi Hassan, Biao Song, Eui-nam "Huh: A framework of sensor-cloud integration opportunities and challenges". ICUIMC 2009: 618-626

[11] Madoka Yuriyama, Takayuki Kushida "Sensor-Cloud Infrastructure - Physical Sensor Management with Virtualized Sensors on Cloud Computing". NBiS 2010: 1-8

[12] Parwekar, P, "From Internet of Things towards cloud of things", 2nd IEEE International Conference on Computer and Communication Technology (ICCCT), September 2011.

[13] Pachube Internet of Things "Bill of Rights" Ed Borden, in pachube.com (2011)

[14] Matthias Kranz, Luis Roalter, and Florian Michahelles, "Things That Twitter: Social Networks and the Internet of Things", What can the Internet of Things do for the Citizen (CIoT) Workshop at The Eighth International Conference on Pervasive Computing (Pervasive 2010), Helsinki, Finland, May 2010.

[15] K Lee and D. Hughes. "System architecture directions for tangible cloud computing". In International Workshop on Information Security and Applications (IWISA 2010), in Qinhuangdao, China, October 22- 25, 2010.

[16] K. Aberer, M. Hauswirth, and A. Salehi. "Infrastructure for data processing in large-scale interconnected sensor networks". In MDM'07, pages 198–205, 2007.

[17] Nikos Kefalakis, John Soldatos, Nikolaos Konstantinou, Neeli R. Prasad "APDL: A reference XML schema for process-centered definition of RFID solutions". Journal of Systems and Software 84(7): 1244-1259 (2011)

[18] Wei Yu. Thang Nam Le, Dong Xuan, Wei Zhao, "Query aggregation for providing efficient data services in sensor networks", IEEE International Conference on Mobile Ad-hoc and Sensor Systems, pp. 31-40, October 2004.

[19] F. Bian, D. Kempe, et al. "Utility based sensor selection". In Proc. IPSN, 2006.

[20] Z. Shelby, B. Frank, and D. Sturek. "Constrained application protocol (CoAP)". https://datatracker.ietf.org/doc/draft-ietf-core-coap/, May 2011.

[21] M. Leggieri, A. Passant and M. Hauswirth, "A Contextualized Cognitive Perspective for Linked Sensor Data", in proceedings at the 3rd International Workshop on Semantic Sensor Networks 2010 (SSN10), collocated with the 9th International Semantic Web Conference (ISWC2010)

[22] Avilés-López, Edgardo and García-Macías, "TinySOA: a service-oriented architecture for wireless sensor networks", in Service Oriented Computing and Applications (Springer), Volume 3, Number 2, June 2009 , pp. 99-108(10).

[23] Patrik Spiess, Stamatis Karnouskos, Dominique Guinard, Domnic Savio, Oliver Baecker, Luciana Moreira Sá de Souza, Vlad Trifa: SOA-Based Integration of the Internet of Things in Enterprise Services. ICWS 2009: 968-975.

[24] Serbanati, A., Madaglia C.M., and Ceipidor U.B., "Building Blocks of the Internet of Things: State of the Art and Beyond", 2011. In RFID/Book 3, eds. C. Turcu. Rijeka: InTech. (In press.)

[25] K. Sperner , S. Meyer, C. Magerkurth "Introducing Entity-based Concepts to Business Process Modeling", 3rd International Workshop and Practitioner Day on BPMN, Lucerne, Switzerland

[26] R. Fielding. "Architectural Styles and the Design of Network-based Software Architectures". PhD thesis, University of California, Irvine, 2000.

[27] Z. Shelby, Sensinode, K. Hartke, and C. Borman. "Observing resources in CoAP". https://datatracker.ietf.org/doc/draft-ietf-core-observe/, March 2011.

[28] Coalesenses iSense "A modular hardware and software platform for wireless sensor networks". http://www.coalesenses.com/isense.

[29] Kalinichenko, L., Missikoff, M., Schiappelli, F., Skvortsov, N. 2003, "Ontological Modeling" In proceedings of the 5th Rusian Conference on Digital libraries", St. Petesburgo, Russia, 2003.

[30] Hauswirth, Manfred., Pfisterer, Dennis., Decker, Stefan., "Making Internet-Connected Objects readily useful". In Interconnecting Smart Objects with the Internet Workshop, Prague, 2011.

[31] M. Serrano, J. Strassner and M. ÓFoghlú, "A Formal Approach for the Inference Plane Supporting Integrated Management Tasks in the Future Internet" 1st IFIP/IEEE ManFI International Workshop, In conjunction with 11th IFIP/IEEE IM2009, 1-5 June 2009, at Long Island, NY, USA.