

Internet of Thing Context Awareness Model

Derrick Ntalasha^{1,*}, Renfa Li and Yongheng Wang¹

¹College of Information Science and Engineering Hunan University, Lushan Road Changsha, 410082, China

Abstract

This paper aims to address key concept of context awareness in the Internet of Things (IoT) domain by providing a model that proactively monitors behaviour of devices and services. This concept represents context into two types of information representation, namely set based and mark-up based ontology for the purpose of defining a context awareness model that accurately represent context in IoT. The context model is defined using Resources, Actors, Ambients and Policies. The model adds value to the next stage of IoT evolution by using context ambients to bring about predictive and proactive modelling in understanding context and context awareness. IoT context is represented using the hierarchical hybrid context model. This model provides a method of representing contexts based on context quality and availability in an entity relation hierarchy form. The model is exemplified using the context management application based on agents. Experimental results indicate that context awareness in the Internet of Things can be enhanced by the proposed model.

Keywords: Internet of Things, Context awareness, Context Model, Context Representation.

Received on 05 January 2014, accepted on 15 November 2015, published on 12 February 2016

Copyright © 2016 Derrick Ntalasha *et al.*, licensed to EAI. This is an open access article distributed under the terms of the Creative Commons Attribution licence (<http://creativecommons.org/licenses/by/3.0/>), which permits unlimited use, distribution and reproduction in any medium so long as the original work is properly cited.

doi: 10.4108/eai.12-2-2016.151084

1. Introduction

The Internet of things (IoT) [1] is fast maturing and evolving to the next stage that brings about connected data in way that provides active monitoring, signal detection, event triggering and predictive modeling. The IoT has evolved from creation of sensors and enablement of their qualities to device ecosystems through context awareness. Context awareness [2] plays an important role in the software architectures by enabling services to be customized according to the current situation with minimal human intervention. Context awareness bridges the gap between real things and the virtual world in the IoT through acquiring, analyzing, and interpreting relevant context information [3]. Context information in this new era is measured not only as set of circumstances or facts that surround a particular event or situation but as information about the users' surrounding environment which could influence their understanding of the context. The processing of context information from the environment calls for the identification of relevant context. The IoT environment has an overwhelming number of context sources and they keep increasing on a daily basis. Factors affecting the change in contexts are not only affected by a diverse environment with different computing and non computing devices, but by the

dynamism of the users' behaviour and an adaptive and reconfigurable IoT context aware middleware is a promising solution.

Despite the environment calling for situational aware systems [4], representation and reasoning of contextual information [5, 6] at various levels of abstraction is core to managing the IoT context from multiple sources. The set based and mark up ontological based representation provides the solution of capturing/sensing of relevant context information and interaction between actors and resources in such an environment.

The IoT domain is represented not only with the traditional mobile computing [7] scenarios that use smart phones and portables, but connecting everyday existing objects and enabling intelligence through the use of sensors, WiFi networks, Global Positioning System devices (GPS), User profiles [8], active badges [9], cameras, radio frequency identification (RFID) [10] etc. The domain provides a shared understanding of the situation of its users and their appliances, software architectures and mobile and pervasive computing networks [11] to process and convey contextual information to where it is relevant and computational artifacts to offer autonomous and smart behavior. With this grounding principles in place, context awareness via anything, anywhere and anytime can be accomplished. The main objective of this paper is to define a context model for the IoT domain that will accurately represent

*Corresponding author. dbntalasha@gmail.com

context to achieve context awareness by defining the ambient which brings to the research front the rich notion of context and context awareness. Ambients bring relevance facts of computing to services and applications. The model uses the concepts in [12] to define a model that captures context from a computational IoT environment with specific boundaries through the use of ambients.

The proposed model addresses the problem of the surrounding area or environment definition of where the context is occurring. The existing models only model the context as defined in [13] without taking into account the fundamental shift of how context is measured and any information that influence its understanding. When contexts occur, they do so in a defined location and an ambient is used to associate a context to the surrounding. Ambients enable processes within the system to be aware of the environment in which they are being executed. For any IoT application to be context aware, that application needs to sense the aspects of the environment and then use the information to adjust accordingly. In this way the model brings in the notion of an active zone, consistency in context representation and reasoning since the ambient will restrict contexts to a defined place or location thereby creating relevance to the acquired and reasoned contexts.

Our model adopts a hybrid context representation approach based on two models namely; the set based mark up model and an ontological model through the use of Web Ontology Language Description logic OWL DL [14] and XML [15]. These two models loosely interact to form a more flexible and general solution. The motivation is to have a hybrid model that provides for more general problem solution that will cater for all requirements needed to satisfy a high level context data coming from multiple sources which characterizes the Internet of things. The OWL-DL is used as the ontology language, because it takes advantage of the reasoning services it supports and is now a de-facto standard in various application domains. For interoperability purposes, all profile data will be kept in XML profiles, but linking those attributes modelling non-shallow context data to ontology concepts that formally define their semantics.

The rest of the paper is organized as follows: in section 2, we present related work by highlighting the most important research done. In section 3 the IoT context model using the hierarchical context representation model and the model elements are discussed. In section 4, the context management application and its architecture are explained. In section 5 the experimental results are explained and section 6 makes the conclusions and future work.

2. Related Work

The research works in this field of study concentrate mainly on development of models with static or autonomic features that can be used to represent real

world contexts and the development of context management models for various application domains. In this section we explore the research works that highlight the state of art context aware application and framework development.

A programmable context awareness framework [16] [17] was proposed that would ease the development and deployment of context aware applications by decoupling context management from business logic. This ontology based framework's main aim was to reduce the overhead of running applications with limited resources by offering best mechanisms to support context awareness and behavior adaptation. The authors used an XML based programming language to structure atomic functions of processing context. The framework used six primitive functions to process context namely produce, filter, abstract, select, aggregate, and consume, using well known layered approach of context management. The major benefits for this framework are reductions of development cost and complexity, adaptation efficiency of context aware application by accurately capturing events related to the context changes without involving the application. The disadvantage of this framework is that events in real time have to be correctly ordered to be managed and its ability to handle extensive number of events is limited. Our model is more generic and can handle more contexts in the vast IoT environment.

The authors of [18] proposed a context aware model for a smart office environment that was used to monitor activities of workers within the work premises. The model focused on the presence and transition of workers by gathering information about staff activities. The main goal of this study was design a smart notice board that acquires and applies knowledge about the environment state in order to reason about it so as to define a desired state for its inhabitants and perform adaptation to these desires and therefore improving their involvement and satisfaction with that environment. According to the implementation, the model used a context manager to achieve communication between an object (i.e., user, task, and resources) and the monitoring system. The communication is governed by a monitoring process, which consists of the contextual monitoring components (i.e., contextual activity, event, time, and location). Our model is proposed to deal with context information from an environment with many sources of context unlike the smart office that only used contexts relating to staff.

A new information model [19] that was centered on context information gathering, representation, manipulation and dissemination to other systems in internet of things was described. The researchers focused on a context model that utilized an ontology using the Web Ontology Language "OWL" to represent many different use cases and scenarios for healthcare services. The new model supported the semantic representation of context by identifying a common upper ontology for context information, while defining a set of low level ontologies that can be applied/merged to/with other distinct sub-domains. The architecture focused on various

contexts by initiating a classification scheme that captures the relationships of the entities found therein and their context information and supporting the use of different reasoning engines to reason about various context applications before dealing/acting on them.

In [12] a context aware model was proposed that addressed the key concept of context sensitive system. The model defined the generic way of representing context that could be easily interpreted by a computer. It described the fundamental concept of context management by using a set notation concept. The triple set representation consisted of context resources, actors and policies. Contexts from the operational environment were mapped to an instance of the triple set. Context reasoning and adaptation was through an ontology that defined the exact relationship between the contexts in question. The proposed was illustrated through the use BDI (Believes Desires Intentions) agents that clearly defined the context management problem. A similar study was done in [20] that described a self configuring middleware for managing context acquisition and representation. The domain of interest was smart closed environment. The model used the self configuring property to monitor variations in the contexts within the environment. Context management within the model was achieved through the introduction of autonomy. The problem with this model is that it only uses ontologies to represent the relationships between the context model concepts and does not model how ambients are dealt with in the resources actor interaction. Ontologies are not well suited to represent some dynamic context data such as users' adaptation preferences; these data can be more profitably modeled by lower-complexity, restricted logics. Most of the ontology languages and related reasoning tools do not properly support uncertainty in context data.

An ontology based middleware [21] [22] was developed using context awareness to enable service discovery and adaptation for ambient intelligent applications. The ontology based middleware was developed to be scalable, adaptive and autonomous in responding to multi agents that covered a large collection of devices and people. The context aware middleware had a formal definition of context expressiveness and reasoning using the ontology that supported multiple programming applications. The middleware managed context through collection, processing and adaptation to various aspects of context changes. This meta model ability to support multiple and diverse applications improved context awareness in the middleware.

A context aware model [23] that not only included expression of context but automatic deployment and execution of context was proposed. The model defined an executable architecture applicable to context aware systems. The authors used a two phased lifecycle to manage contexts. The first was the design phase, which concentrated on describing situational context and how to configure a service application to use user context. The other phase was the running phase, was the engine used to manage the entire model. The execution process included

model navigation, managing user transitions and providing the right application that offers the required service. The proposed model used context changing events to manage the user's scene transitions and invoke the context-aware services according to the scene's configuration. Our model is generic in nature and can be tailored to work for any context aware application that uses context.

3. IoT Context Awareness Model

The IoT context model is defined by $\text{IoT_C} = \langle \mathbf{R}, \mathbf{A}, \mathbf{A}, \mathbf{P} \rangle$ where R is the set of contextual information resources; A is a set of context aware actors (CAA) that interact with context resources.; A is a set of ambients that define computational interactional relevance between context resources and context aware actors within a computational boundary; P is the set of context rules to be followed in the execution environment. The model is defined for the Internet of Things paradigm. Figure 1 shows how the model represents contexts typical to the IoT domain using the OWL DL ontology.

The relationship type applicable to the model is generally the is-a type relationship. This will be used to map to the database and mark-up based ontology for storage and reasoning respectively.

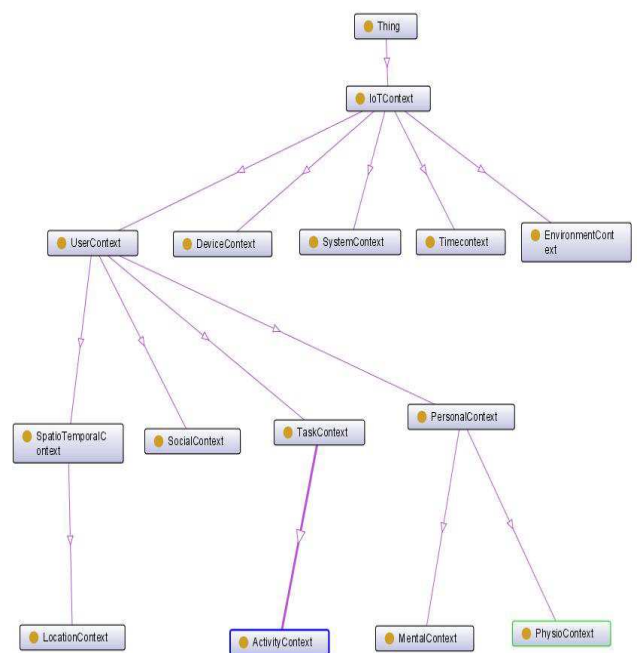


Figure 1. shows the OWL Context Representation

According to Dey and colleagues, they identified four primary context types that characterize the situation of an entity but in the Internet of Things context has more than location, identity, time and activity. IoT context has extra context types. Our model defines context in a way that each definition is composed of bits of descriptive information with a concrete meaning. Every context is

then defined by a bunch of Context information instances. As shown in figure 1, the IoT context model representation is a hierarchical structure showing contexts. Each context includes model attributes (Resources, Actors, Ambients, and Policies).

- User context is any information that focuses on the user information and needs, social interaction, interests, expertise, Activities, tasks, location. Each of these contexts in our model is briefly discussed.
- *Location*
Any user or indeed an entity in the IoT domain will have a physical location and information about nearby objects. Generally, location can be described and collected either by physical (GPS) or symbolic positions (Motion sensor)
- *Activity*
This type of context takes into account activities that are currently being performed by users which can easily help in performance adaptation. This context type defines the physical activities of the user and can be obtained through body worn sensors and environment sensors.
- *Personal*
Personal or demographics context type provides more information about the user. It gives more information on the personal data. For instance, the age, gender of a user may influence the preference to the type of devices, applications, they access and the activities they do.
- *Mood*
This context type is derived from user context and provides more emotional dimension to the user. The user situation can be derived through the mood variables attributed to the user at any time.
- *Time*
Time context types represent facts about relations among instants and intervals, durations and date time information.
- *System*
System context type typically is information about the kind of application. The application provides services through a well defined interface. These properties and many more will identify that application's contexts.
- *Environment*
The environment context gives account of the surroundings of the entity in the IoT Environment. All elements of the environment can be captured to give information about the surroundings where the entity is based.
- *Device*
Device type context like system context describes device capabilities, for example, memory, computing power, user interface, and protocol used. This context is important for device identification and information capture.

3.1. IoT Context Model Elements

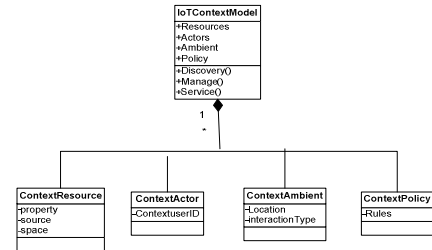


Figure 2. shows the model elements

3.1.1 Context Resources

A context resource within the model is given by a structure. Although every context resource is different, they all share common properties defined by the contextual information. So each contextual information has properties and features; resource properties, resource sources, and resource property space.

i. *Resource Properties*

A resources property is the specific relevant contextual information that a resource can provide. These properties can be transitive, symmetric or functional. A resource property is in the form

$$R_n(R_D) \rightarrow R_R$$

where R_n is the name of the property, R_D is the domain of the execution and R_R is the property range.

ii. *Resource Sources*

This defines the source of the events which the resource is interested, which an actor has registered and expressed interest in. Context can be acquired from diverse sources. Context sources can be monitored and collected by a variety of sensing devices. Resource sources contain methods of how to obtain the context information required and it is defined by;

$$R_{so} [S_{pub}, S_{pus}] = R_s$$

Where R_{so} is the resource an actor is interested in, S_{pul} is the pull method used to obtain and send the information to the event source as soon as it changes its state. S_{pus} is the push method used to obtain contextual information when the source notifies the change. R_s is the obtained resource.

iii. *Resource Property Space*

This is the space in which a resource actor can capture contextual information. This space is defined by a cardinality which has a minimum and maximum ranges.

$$CK_{sc} = [MinCardinality - MaxCardinality]$$

The property space of a context resource attached to an actor is said to have minimum cardinality if and only if one actor offers relevant context information. For the maximum

cardinality there is more than one actor providing contextual information in a space.

3.1.2 Context Actors

Actors can be human or systems that use contextual information captured/sensed/generated to effect an activity that is relevant to the context. Therefore, these can be real or virtual entities using contexts to fulfill needs. The context actor is uniquely defined by the context information it provides, captures or generates. In our model the actor is represented as;

$$A_{ext} = (A_{nam}, A_{id}, A_{st}, A_{info}, A_{mtd})$$

Where A_{ext} is the context actor, A_{nam} is the name of the context, A_{id} is the unique identity of the context, A_{st} is the state of the context source or generator, A_{info} is the definition of the context generated and A_{mtd} is the definition of the methods used to obtain information.

3.1.3 Context Ambient

Ambients are bounded places in which a computation is occurring. Contexts resources generate and/ or process context for actors to effect their goals and ambient provide relevance of their interaction. Each actor can sense contextual information which may not be relevant. So using the ambient, each interaction is better defined and ambiguity is reduced. An ambient can be a user, a location, a device or software agent, with a collection of local processes and can have other ambient. Ambient usage in the context model is exemplified below.

Context

1. System [Interface A [# Device 1] | Interface B [# Device 2]]

2. Location [Building A [# Device 1] | # Device2] Building B []]

Ambient

1.Device 1

2.Device 2

Using this illustration contexts are mapped to the specific actor that has relevance for that interaction required.

3.1.4 Context Policies

A policy represents a set of rules followed by actors or resources within a computational space.

4. Context Management Application for IoT

The IoT awareness model addresses how context awareness plays a critical role in deciding what data needs to be captured, processed/reasoned, stored and much more. It defines how concrete contexts and their subsets can be attained through the use of IoT enabled devices, web services, applications and users.

Our context awareness model has features that provide a basis of how contexts can be acquired, managed and represented. To demonstrate the viability of the proposed model, we developed an application to manage contexts in

the IoT environment through the use of agents. Experimental results carried out using Wi-Fi, GPS and Web Services indicate that the proposed model can support context awareness in the vast IoT environment.

The solution uses the discovery agent called General/Unlimited Inquiry Access Code (GIAC) [24]. The choice of this discovery agent is because it offers first level of filtering for finding Bluetooth enables devices and services, GPS enabled Bluetooth, Wi-Fi enabled services. These agents are reactive and can be easily extended to other devices that are wireless enabled.

For this application, the agents have facilitated the acquisition of contexts from a defined IoT environment. The agents that the application uses are able to interact seamlessly with distributed contexts providers and adapters (Actors). Our context management application makes use of the following; context discovery agent, location management agent, user carrier agent, web services agent, and Wi-Fi management agent. All these agents are able to perform the basic context and service discovery, and context management. The application contains a layer that formats the acquired contexts into eXtensible Markup Language (XML) files. Above all we have the discovery listen that coordinates the model as contexts are being acquired.

4.1 Context Management Architecture for IoT

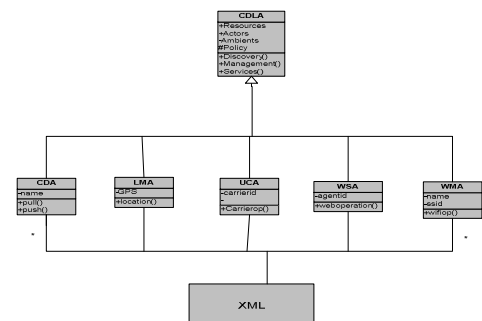


Figure 3. shows class architecture for the application

The architecture for the model has five (5) components and each component's operation is discussed in section 4.1.1

4.1.1 Context Discovery Listening Agent (CDLA)

The CDLA allows the application to receive contexts discovered from devices and services through events discovery. Its main objective is to coordinate the application as it receives contexts from the sources. Other services offered by the CDLA are defining context ambient and context policies. As contexts are being discovered and acquired, the CDLA associates the discovered and acquired contexts to the context actors. This will imply that the contexts are filtered and only forwarded to the relevant actors. CDLA also provides context policies to discover devices and their associated

contexts. For the web service contexts, the CDLA only allows the available services to be reported as requested by the actor. The discovered context is written to an XML file.

4.1.2 Context Discovery Agent (CDA)

The context discovery agent performs the context discovery process from the identified resource (Devices and Services). Each local device agent provides contexts whenever the CDA calls/queries the local resource. This agent uses two methods to discover contexts. It can either start and send inquiry or receive contexts whenever the local resources have contexts that the agent has interest in.

4.1.3 Location Management Agent (LMA)

The LMA provides the application (CDLA) access to the resource location service. Its gets updates of the resource location through the GPS location provider. Because of privacy concerns of location awareness the actual location is encapsulated and given an identity.

4.1.4 User Carrier Agent (UCA)

This agent is in charge of acquiring contexts about resource carrier, which carries information about the type of connection the resource uses to access actors or other resources.

4.1.5 Web Services Agent (WSA)

The WSA acquires and publishes context for the specified web services. It initially populates its registry by broadcasting packets and then the returned services are added to the local service directory.

4.1.6 Wi-Fi management Agent (WMA)

The WMA manages all aspect of Wi-Fi contexts. It provides contexts for available networks and their capabilities, states, access points and a number of various intent actions.

5. Discussion of the Experimental Results

The experimental results of the RAAP model described above used the context management architecture to illustrate the functionality of the model. The context management system is the heart of a context-aware architecture and processes instantiations of this context model. It is responsible for context discovery and retrieval, structured storage of context, formatting into XML for the IoT devices and services. The discovery of context from devices and services can be understood as an action of looking for requested context/services in the defined domain. Furthermore the model uses hierarchical attribute-value pairs, using eXtensible Markup language (XML) for formatting context discovered from devices and services for worldwide and local area network usage. Context information is described and categorized hierarchically by using XML for all cases in our experiments.

Figure 4 shows the context information obtained from three (3) context discovery agents namely; Context Discovery Agent, Location Management Agent and User Carrier Agent. These agents gathered context information from the sources and the application hierarchically formatted it into the attribute value pairs.

```
<SimpleContext
name="Ntalasha"
id="5C57C86E81ED"
carrierid="f268de"
ComponentAddressid="c2cf83" Locationid="43da1b"
dateTime="2013-07-22T17:19:18"
class="ocp.context.SimpleContext">
</SimpleContext>
```

Figure 4. shows context for CDA, LMA, UCA

As shown in the diagram, the hierarchical context shows the model representation where the resource context is defined by the context name (name), identity (id) and the resource source (carrierid) of the context. The context actor is represented by the source component identity (componentaddressid) and the context ambient is the encapsulated location. In addition the time when the context was retrieved is also given. The Policy is embedded within the model class on how each context can be extracted from the source that has been bound to the application. This ensures validity of contextual information retrieved.

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope
xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
xmlns:wsd="http://schemas.xmlsoap.org/ws/2005/04/discovery"
xmlns:wsp="http://schemas.xmlsoap.org/ws/2006/02/devprof"
xmlns:pub="http://schemas.microsoft.com/windows/pub/2005/07">
  <soap:Header>
    <wsa:To>urn:schemas-xmlsoap-org:ws:2005:04:discovery</wsa:To>
    <wsa:Action>http://schemas.xmlsoap.org/ws/2005/04/discovery/Hello</wsa:Action>
    <wsa:MessageID>urn:uuid:b8742543-4582-49ef-b9c9-79936853698d</wsa:MessageID>
    <wsd:AppSequence InstanceId="162"
SequenceId="urn:uuid:f193a68b-3c3b-4a86-abea-a9b0e0529ef5" MessageNumber="1500">
      </wsd:AppSequence>
    </soap:Header>
    <soap:Body>
      <wsd:Hello>
        <wsa:EndpointReference>
          <wsa:Address>urn:uuid:2d4f81d0-b584-4e73-87af-a3c935dd4b12</wsa:Address>
          </wsa:EndpointReference>
          <wsd:Types> wsp:Device pub:Computer
        </wsd:Types>
        <wsd:MetadataVersion>8</wsd:MetadataVersion>
      </wsd:Hello>
    </soap:Body>
  </soap:Envelope>
```

Figure 5. shows context for WSA

The web service agent uses SOAP [25] to retrieve the context information from the source. Using SOAP as a transport service, context information was discovered from the computer where the context management application installed. The message is an ordinary XML document containing the Envelope, Header and the Body as shown in figure 5. The resource context in the header of the message defined the properties, functionality and the namespace. The feature of this message contains the discovered message that was sent to the computer as the resource actor and the ambient being the web services address giving a boundary to the context discovered. The policy is that only context defined within the namespace can be retrieved.

```
<SimpleContext ssid="unknown" BSSID="none"
WifiMangerid="801059" capabilities="0"
Locationid="43da1b" level="0" frequency="0"
Datetime="2013-07-22T17:19:18"
class="ocp.context.SimpleContext">
</SimpleContext>
```

Figure 6. shows context for WMA

As shown in figure 6, the wifi management agent in the context management application retrieves information context from the available wifi resources. The application gives an output according to the model representation. Since wifi networks are private, the context retrieved only showed the ambient context for the wifi context resource boundary and the other information was null.

6. Conclusion and Future Work

This paper discussed the proposed IoT Context awareness model. The model showed how context can be represented to achieve context awareness in the IoT domain. It also briefly discussed how the context in the IoT complex domain can be captured and represented in a hybrid context model using Web Ontology Language Description logic OWL DL and XML. This model provides a basis on how the Internet of Things context can be modelled and later reasoned to achieve some goals of computational awareness. Future works will be looking at how to combine OWL DL and CC/PP to achieve high expressionless in the definition of context.

Acknowledgements

This research work is done under the projects funded by the Study of Proactive Complex Event Processing for Large-scale Internet of Things a project of National Natural Science Foundation of China (61371116) and National High Technology Research and Development Program of China (Program 863) No. 2012AA01A301-01

References

- [1] D. Miorandi, *et al.*, "Internet of things: Vision, applications and research challenges," *Ad Hoc Networks*, vol. 10, pp. 1497-1516, 2012.
- [2] C. Perera, *et al.*, "Context aware computing for the internet of things: A survey," *Communications Surveys & Tutorials, IEEE*, vol. 16, pp. 414-454, 2014.
- [3] M. d. L. Martinez-Villasenor and M. Gonzalez-Mendoza, "Sharing and Reusing Context Information in Ubiquitous Computing Environments," in *Artificial Intelligence (MICAI), 2014 13th Mexican International Conference on*, 2014, pp. 227-230.
- [4] O. Voutyras, *et al.*, "Achieving autonomy in IoT systems via situational-aware, cognitive and social things," in *Proceedings of the 18th Panhellenic Conference on Informatics*, 2014, pp. 1-2.
- [5] K. Park, *et al.*, "Semantic reasoning with contextual ontologies on sensor cloud environment," *International Journal of Distributed Sensor Networks*, vol. 2014, 2014.
- [6] A. M. Khattak, *et al.*, "Context Representation and Fusion: Advancements and Opportunities," *Sensors*, vol. 14, pp. 9628-9668, 2014.
- [7] T. Bresnahan and S. Greenstein, "Mobile computing: The next platform rivalry," *The American Economic Review*, vol. 104, pp. 475-480, 2014.
- [8] J. Hu and K. Heussen, "Survey and Characterization of User Profiles and User Requirements: RTLabOS D2. 2," Technical University of Denmark, Department of Electrical Engineering 2014.
- [9] H. K. Fard, *et al.*, "Indoor positioning of mobile devices with agile iBeacon deployment," in *Electrical and Computer Engineering (CCECE), 2015 IEEE 28th Canadian Conference on*, 2015, pp. 275-279.
- [10] S. Zhang, *et al.*, "Research on smart sensing RFID tags under flexible substrates in printed electronics," in *Electronic Packaging Technology (ICEPT), 2015 16th International Conference on*, 2015, pp. 1006-1009.
- [11] A. Aziz, "Mobile and Pervasive Computing," 2014.
- [12] I. Salomie, *et al.*, "RAP-a basic context awareness model," in *Intelligent Computer Communication and Processing, 2008. ICCP 2008. 4th International Conference on*, 2008, pp. 315-318.
- [13] G. D. Abowd, *et al.*, "Towards a better understanding of context and context-awareness," in *Handheld and ubiquitous computing*, 1999, pp. 304-307.
- [14] Q. Ji, *et al.*, "Conflict Resolution in Partially Ordered OWL DL Ontologies," in *ECAI 2014: 21st European Conference on Artificial Intelligence*, 2014, p. 471.
- [15] T. Bray, *et al.*, "Extensible markup language (xml) 1.0," ed, 2011.
- [16] B. Chihani, *et al.*, "Programmable context awareness framework," *Journal of Systems and Software*, vol. 92, pp. 59-70, 2014.
- [17] G. Sehic, *et al.*, "A programming model for context-aware applications in large-scale pervasive systems," in *Wireless and Mobile Computing, Networking and Communications (WiMob), 2012 IEEE 8th International Conference on*, 2012, pp. 142-149.
- [18] N. O. Nor Azlina Aziz Fadzillah, Siti Zaleha Zainal Abidin, and Nor Azylia Ahmad Azam, "Context Awareness Monitoring Model for Smart Office Environment," presented at the The Third

- International Conference on Advanced Collaborative Networks, Systems and Applications, France, 2013.
- [19] M. Aly, *et al.*, "A new information model towards context-aware service provisioning in the Internet-of-Things," in *Intelligence in Next Generation Networks (ICIN), 2015 18th International Conference on*, 2015, pp. 160-167.
- [20] I. Anghel, *et al.*, "Middleware for smart environments management," *International Journal of Computers, Communications & Control*, vol. 5, 2010.
- [21] A. Babu and R. Sivakumar, "Development of ontology based middleware for context awareness in ambient intelligence," in *Hybrid Intelligent Systems (HIS), 2014 14th International Conference on*, 2014, pp. 219-224.
- [22] H. Guermah, *et al.*, "Context modeling and reasoning for building context aware services," in *Computer Systems and Applications (AICCSA), 2013 ACS International Conference on*, 2013, pp. 1-7.
- [23] C. Weijie, *et al.*, "A context-aware services development model," in *Service Sciences (IJCSS), 2012 International Joint Conference on*, 2012, pp. 194-199.
- [24] J. Lewandowski and H. E. Arochena, "MOBILE ATTENDANCE AND TIME MONITORING SYSTEM FOR M-LEARNING APPLICATIONS: DESIGN AND PILOT RESULTS," *mobile*, p. 3, 2011.
- [25] W. W. W. Consortium, "Simple object access protocol," URL: <http://www.w3.org/TR/SOAP>, 2007.