

Activity Context Modeling in Context-Aware Environment

Samuel King Opoku
Computer Science Department,
Technical University of Kumasi
Kumasi, Ghana

Abstract: The explosion of mobile devices has fuelled the advancement of pervasive computing to provide personal assistance in this information-driven world. Pervasive computing takes advantage of context-aware computing to track, use and adapt to contextual information. The context that has attracted the attention of many researchers is the activity context. There are six major techniques that are used to model activity context. These techniques are key-value, logic-based, ontology-based, object-oriented, mark-up schemes and graphical. This paper analyses these techniques in detail by describing how each technique is implemented while reviewing their pros and cons. The paper ends with a hybrid modeling method that fits heterogeneous environment while considering the entire of modeling through data acquisition and utilization stages. The modeling stages of activity context are data sensation, data abstraction and reasoning and planning. The work revealed that mark-up schemes and object-oriented are best applicable at the data sensation stage. Key-value and object-oriented techniques fairly support data abstraction stage whereas the logic-based and ontology-based techniques are the ideal techniques for reasoning and planning stage. In a distributed system, mark-up schemes are very useful in data communication over a network and graphical technique should be used when saving context data into database.

Keywords: context; context modeling; context-aware computing; hybrid context model; mobile computing; pervasive computing

1. INTRODUCTION

With the proliferation of mobile devices, one looks to pervasive computing to provide personal assistance in this digital world. Pervasive computing refers to obtaining available data any time at any place. One critical aspect of pervasive computing is context-aware computing in which applications are made to track, use and adapt to contextual information. The definition of context by Dey and Abowd [1] which is also adopted in this work is, “Any information that can be used to characterize the situation of an entity. An entity is a person, place or object that is considered relevant to the interaction between a user and an application including the user and application themselves”. The relatively important contexts [2], [3] are location, time, identity and activity. The first three contexts are obtained through well-known mechanisms. The location context can be determined when the device is outdoor through Global Positioning System (GPS) [4] or indoor through radio frequency technologies like Bluetooth [5]. Concerning the time of the day, the clock system of a mobile device can be used in connection with external authenticated system [6]. However, many techniques are employed in tracking activity context. This paper reviews the prominent techniques stating the pros and cons of the various mechanisms. The paper ends with the best approach for activity modeling through the various stages of data acquisition and interpretation.

Context modeling is the process of identifying the appropriate contextual information of interest and establishing relationships and reactions among the pieces of contextual information. Schmohl and Baumgarten in [7] suggested that context modelling requires two phases: the first phase is to determine the conceptual abstraction of real world characteristics. The second phase is to map the concepts on a context model that represent the information. To sense and interpret contextual information, many components are required. The core components are sensors, middleware, context repository, context reasoning and communication interface. Context sensor can be hardware or software that

fetches raw data for contextual information. Because the raw data cannot be interpreted by high level languages [8], there is the need for a middleware that refines the raw data from sensors into the required data structures [9]. The context repository stores the current context based on the data structures generated by the middleware. Context reasoning infers new context based on the current context information obtained from the middleware. The context reasoning relies on a set of rules [10] to infer the new context. Communication interface is required in distributed system. Reasoning mechanism may commit contextual updates into a network or may request contextual information through a network. The context reasoning after inference passes the message to the context-aware application which uses the information to adapt itself through the execution or termination of certain programs. A conceptual view of the context utilization mechanism is illustrated below:

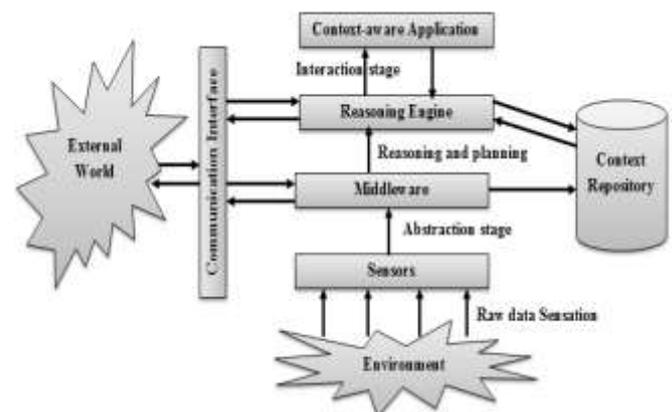


Figure 1 Context Acquisition and Utilization Framework

There are six mechanisms or models that can be used to sense and interpret a context within the various components stated above. These are key-value, mark-up scheme, graphical, object-oriented, logic-based and ontology-based models

2. MODELING TECHNIQUES

The six modeling techniques that can be used to model activity context are discussed in details below. To illustrate how each model is implemented, the following case study is model during the discussion of the various techniques. “A computer science book (BK) is read by computer scientist (CS). BK has Title, ISBN; Author, Barcode as its attributes. The title and author are string data type whereas ISBN and Barcode are integer data type. The author is usually computer scientist who of course is a human. Humans are characterized by name and age being string and integer data types respectively.”

2.1 Key-Value Models

They require simple data structures to associate context attributes with specific values of conceptual information [7]. Schilit et al in [11] used key-value pairs to model the context by assigning the value of context information to an environment variable in an application. From the case study above, the characteristics of the entities – BK and CS – which form the keywords for the modeling are:

- **BK:** title, ISBN, author, barcode
- **CS:** name, age

Since human and CS are the same in terms of characteristics, human is ignored in the modeling. The characteristics that can be used to identify BK are ISBN and barcode. Both features are of integer data types, though barcode are sometimes alphanumeric. This work restricts itself to only numeric barcodes. It is therefore not applicable to alphanumeric barcodes. There is other integer feature called age that belongs to CS. However, the length of age should not be more than three. Barcode has varying length but usually more than three. Similarly, the length of ISBN is more than three. Hence, every set of data consisting of numeric characters with length greater than 3, is associated with BK. The other features of BK are author and title. They cannot be used in our model since author and title consist of set of characters similar to the features of CS. Thus to avoid ambiguity, author and title are ignored. Similarly, the name feature of CS is also ignored. Regarding CS, the characteristics the can be used to identify CA is the age. Thus if the data is integer with length less than or equal to three, then CS is being referenced. The figure below gives the key-value model for the above case study

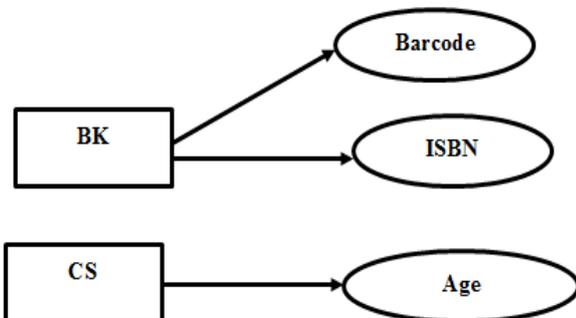


Figure 2 Key-Value Sample Model

There is no relationship established between BK and CS as discussed above. The characteristics provided should be followed closely so that entities can be identified and reasoning inferred. It can be deduced that key-value model is flexible and easy to manage in a small system. However, the

model limits the amount of data. The model is also application dependent and is not adaptive. There is no validation support and relationship modeling.

2.2 Mark-up scheme models

They used hierarchical data structures based on mark-up tags with attributes and comments. It is based on serialization of Standard Generic Mark-up Language (SGML), the superclass of all mark-up languages. Among the commonly used mark-up schemes are eXensible Mark-up Language (XML) and Resource Description Framework (RDF). XML is used to package data or information and RDF is used for conceptual description or modelling in web resources using a data serialization format. The above case study is designed using XML. XML file cannot contain multiple root elements. Thus each root element – BK, CS and Human – are contained in separate files. In the figure below, only section of BK is implemented.

```

1  <?xml version="1.0" encoding="utf-8" ?>
2  <CSBooks>
3  <BK>
4  <title>some data</title>
5  <author>some data</author>
6  <isbn>some data</isbn>
7  <barcode>some data</barcode>
8  </BK>
9  <BK>
10 .
11 .
12 .
13 </BK>
14 .
15 .
16 .
17 </CSBook>
18
  
```

Figure 3 Mark-up Sample Model

Some researchers extended RDF capabilities to Composite Capabilities/Preferences Profile (CC/PP) [12] and User Agent Profile (UAProf) [13] which allow the definition and preferences in context delivery. For instance, in [14] and [15], the researchers designed Comprehensive Structured Context Profile (CSCP) and CC/PP Context Extension respectively to handle the limitation of CC/PP. The limitation of CC/PP is that it allows specific values only. Another context modelling approach which does not follow CC/PP is the pervasive Profile Description Language (PPDL) [16]. This is an XML-based language for designing interaction patterns on limited scales. In order to reduce bandwidth consumption, XML can be combined with JavaScript Object Notation (JSON) [17] which reduces the transmission time of context-aware applications. JSON is lightweight data exchange format that allows computers to generate and parse data easily and faster. Mark-up schemes are flexible and structured. There are available tools for processing and very useful when data is travelled along a network or communication link. Unfortunately, mark-up schemes depend on the application and information is hard to extract.

2.3 Graphical models

They represent contextual entities and their relationships graphically using tools like Uniform Modelling Language (UML) and Entity Relationships (ER) schemas. UML is very

expressive since it uses directed graph to indicate the relationships between concepts. However, it is difficult to work with UML [18] due to its increasing complexity. As of 2016, there are more than fourteen different diagrams making it virtually impossible for developers to recognize. ER on the other hand describes conceptual pieces as entities and the interactions as relationships. The major limitation of ER is the lack of semantics between the entities due to limited representation of relationships, data manipulation, constraints and specifications between entities. The above case study is modeled using ER. The entities in the case study are CS, BK and Human

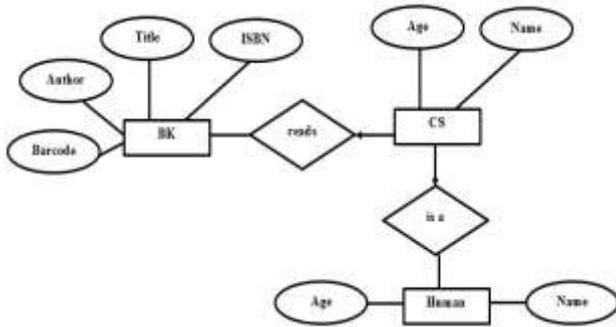


Figure 4 Sample Entity Relations Graphical Model

To handle the limitations of ER, extensions are provided by researchers through the additions of extra features in context modeling. Bauer in [19] used UML extensions to model air traffic management system. Another type of graphical model is the Object-Role Model (ORM) [20] which is a conceptual level modelling method to handle the semantics of data and its interrelationship among the data. ORM models concepts as facts [21]. The work in [22] extended ORM by employing contextual classification and description properties which introduced history fact to cover time aspect of the context and fact dependencies. Thus a change in one fact automatically leads to a change in another fact. According to Mohan and Singh in [21], formal semantics of ORM and Context Modeling Language (CML) can be supplemented to provide integration with other implementations. Graphical modeling technique as its strength provides relationship modeling, flexible implementation and it is very useful for data storage and historic context store [23]. However, the limitations of graphical modelling technique include its complexity to retrieve information which requires obligatory configuration. It does not support interoperability between heterogeneous implementations.

2.4 Object-Oriented Models

They employ the use of object-oriented characteristics such as encapsulation and reusability. Encapsulation hides the implementation of objects while reusability allows models to be reused. The Active Object Model of the GUIDE project in [24] was based on object-oriented model. In object-oriented model, entities are modeled using classes which are implemented as objects. The attributes are implemented as data fields or variable in object-oriented model whereas entity behaviors are modeled as methods (in Java) or functions (in C++). The above case study is modeled using Java in the figure below.

```
public class BK{
    private String title;
    private String author;
    private int isbn;
    private int barcode;

    //other methods and fields
    .
    .
    .
}

public class CS extends Human{
    //other methods and fields
}

public class Human{
    //other methods and fields
}
```

Figure 5 Object-Oriented Sample Model

The object-oriented model provides relationship modeling. The tools for processing are available and it allows every integration while supporting data transformation over network or communication link. It is however, limited by its complexity to retrieve information.

2.5 Logic-based Models

They use rules and expressions to define a context. Logics are used to define conditions for formulating expressions or facts. It is best used at the reasoning section of a context-aware application. The first logic based context modelling approach occurred in 1993 by McCarthy [25] and refined in 1997 [26]. The refined work introduced abstract mathematical entities complemented with useful properties in artificial intelligence which allowed simple axioms to be used in common sense phenomena. Akram and Surav in [27] tried to give theoretical semantic model of natural language in a formal logic system. Many researchers had focused on first order logic in their implementations. Gray and Salber in [28] used it to represent contextual propositions and relations. In [29], first order logic was used in connection with Boolean algebra to design middleware infrastructure called Gaia which allowed various rules to describe context information. It was used in [30] to describe context information properties and structure and the kinds of operations they can perform. Another application is to use logic based model with other modelling techniques. Gu et al in [31] proposed Service-Oriented Context-Aware Middleware (SOCAM) architecture for building context-aware services. In their model, first-order predicate calculus was used to model a context and used different modeling techniques, ontology based model written in Web Ontology Language (OWL) as a collection of RDF triples, to describe the context predicate. Using the first order logic, the following model can be obtained from the above case study:

$$\forall x, (is-isbn(x) \vee is-barcode(x) \Rightarrow BK(x))$$

$$\forall x, (\forall y (is-name(x) \wedge is-age(y) \Rightarrow CS(x,y)))$$

$\forall x, (is-CS(x) \Rightarrow is-Human(x))$

$\forall x, (is-BK(x) \Rightarrow \neg is-CS(x))$

$\forall x, (is-BK(x) \Rightarrow \neg is-Human(x))$

$\exists x, (\forall y ((is-author(x) \wedge is-title(x)) \vee \neg is-age(y)) \Rightarrow BK(x,y))$.

The above first order logic can also be described using fuzzy logic. The figure below illustrates the fuzzy logic representation.

```

if (data set is isbn) or (data set is barcode) then
    data set represents BK
else if (data set is name) and (data set is age) then
    data set represents CS
    data set also represents Human
end if
    
```

Figure 6 Fuzzy Logic Sample Model

The strengths of logic-based model are that it generates high-level context based on low-level context. It is simple to use and simple to model. Its limitation is the complication of its applicability. It is also difficult to maintain due to partial validation

2.6 Ontology-based Models

Ontologies are used to represent concepts and their interrelationships. This model enables contextual knowledge sharing and reusability [7]. Ontologies are the most expressive context representation models [8] that support dynamic aspects of context awareness. However, they require ontology engines which have high requirements on resources producing negative performance impact on local context processing where resource-constrained devices are employed. Among the prominent proposal on ontology based modeling techniques are SOUPA [33] for pervasive environment and CONON [34] for smart home environment. The figure below illustrates the ontology model of the above case study.

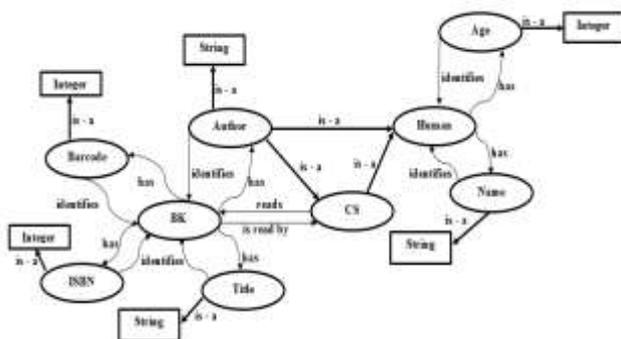


Figure 5: Ontology Model Sample

The ellipses define the basic classes and every arrow defines a relation between these entities. The rectangles are added to show data type values for completion.

OWL-DL is the choice for model context [32] or its variations since it is supported by a number of reasoning services. Web Ontology Language – Description Logic (OWL-DL) allows definition of classes, individuals, characteristics of individuals and relations between various

individuals. The figure below illustrates the OWL – DL version for some part of the model.

```

1 <owl:Class rdf:ID="Human">
2   <rdfs:subClassOf rdf:resource="http://www.w3.org/
3     2002/07/owl#Thing"/>
4 </owl:Class>
5 <owl:DatatypeProperty rdf:ID="age">
6   <rdfs:domain rdf:resource="#Human"/>
7   <rdfs:range rdf:resource="http://www.w3.org/
8     2001/XMLSchema#nonNegativeInteger"/>
9 </owl:DatatypeProperty>
10 <owl:DatatypeProperty rdf:ID="name">
11   <rdfs:domain rdf:resource="#Human"/>
12   <rdfs:range rdf:resource="http://www.w3.org/
13     2001/XMLSchema#string"/>
14 </owl:DatatypeProperty>
15 <owl:Class rdf:ID="CS">
16   <rdfs:subClassOf rdf:resource="http://www.w3.org/
17     2002/07/owl#Thing"/>
18 </owl:Class>
19 <owl:ObjectProperty rdf:ID="reads">
20   <rdfs:domain rdf:resource="#CS"/>
21   <rdfs:range rdf:resource="#BK"/>
22 </owl:ObjectProperty>
    
```

Figure 6: OWL-DL Model Sample

Generally, ontology based models support semantic reasoning, provide an easier representation of context and provide support by standardization. It is however, complex to retrieved data. Also, it can be deduced from the OWL-DL model in Figure 6 that it is inadequate in defining complex context descriptions and thus affects context ontological reasoning.

3. BEST CHOICE TECHNIQUE

What is the best technique(s) to use to model activity context? The works of [21] and [35] introduced hybrid approach by combining different modelling techniques to improve performance. Perera et al in [36] surveyed high-level context modelling techniques and concluded that diversification of modelling techniques is the best way to provide efficient results which will lessen each other's weaknesses. Finally, [23] discussed the context modelling techniques and used logic based technique to model spatiotemporal context. They finally concluded that no modelling technique is ideal to be used in a standalone manner. From the above discussion, it is therefore laudable to combine modeling techniques to produce efficient results. Thus the best technique from context acquisition to utilization is hybrid

There are many stages involved in sensing and interpreting context. At each stage, it is better to study the strengths and limitations of each modelling technique and decide the better technique. For example, at the middleware layer, if the abstracted context is to be sent via network, it will not be appropriate to use logic-based modeling, graphical modeling and key-value modeling techniques. However, mark-up and object-based modelling techniques can be used depending on hardware, software and architectural heterogeneity and interoperability between the communicating devices. Similarly, it will be ideal to model context reasoning using the logic-based modeling technique when the application chooses to use the usual if-then rule set mechanisms. Hence the dynamic behaviour of the context-aware system cannot be overlooked when choosing context modeling techniques. The

figure below illustrates the appropriate modeling techniques that can be used at the various modeling stages from which the desired method can be selected.

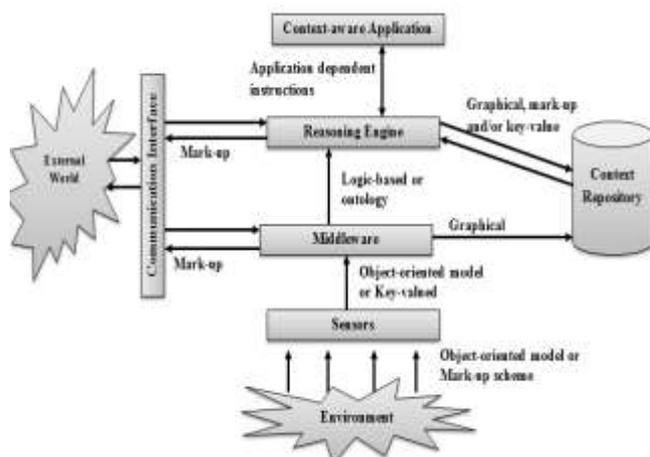


Figure 7 Modeling Implementation of Activity Context Life

4. REFERENCES

- [1] Dey, A. K., and Abowd, G. D. Towards a better understanding of context and context-awareness. Workshop on the What, Who, Where and How of Context-Awareness, affiliated with the 2000 ACM Conference on Human Factors in Computing Systems (CHI 2000), 2000
- [2] Dey, A. K. Context-aware computing. *Ubiquitous Computing Fundamentals*, 321-352, 2010
- [3] Castillejo, E., Almeida, A. & Lopez-de-Ipina, D. Modelling users, context and devices for adaptive user interface systems. *International journal of pervasive computing and communications*, 10(1), 69-91, 2014
- [4] Bajaj, R., Ranaweera S. L., & Agrawal, D. P., GPS: Location-Tracking Technology. *IEEE* 35(4), 92–94, 2002
- [5] Opoku, S. K., An indoor tracking system based on Bluetooth technology. *Cyber Journals: Multidisciplinary Journals in Science and Technology, Journal of Selected Areas in Telecommunications (JSAT)*, 2(12), 1-8, 2011
- [6] Opoku, S. K., An automated biometric attendance management system with dual authentication mechanism based on Bluetooth and NFC technologies. *International Journal of Computer Science and Mobile Computing* 2(3), 18-25, 2013
- [7] Schmohl, R. & Baumgarten, U., Context-aware computing: a survey preparing for generalized approach. *Proceedings of the international multi-conference of engineers and computer scientists, UMECS*, 1(3), 19-21, 2008
- [8] Strang, T. & Linnhoff-Popien, C (2004). A context modeling survey. In *Proceedings of the Workshop on Advanced Context Modelling, Reasoning and Management* associated with the 6th International Conference on Ubiquitous Computing (UbiComp), Nottingham.
- [9] Gellersen, H. W., Schmidt, A. & Beigl, M., Multi-sensor context-awareness in mobile devices and smart artifacts. *Mobile Networks Applications*, 7(5):341–351, 2002
- [10] Jacob, C., Linner, D, Radusch, I. & Steglich, S. (2007). Loosely coupled and context-aware service provision incorporating the quality of rules. In *ICOMP 07: Proceedings of the 2007 International Conference on Internet Computing*. CSREA Press, 2007
- [11] Schilit, B, Adams, N., R. Want, et al. (1994). *Context-aware Computing Applications*, Xerox Corp., Palo Alto Research Center.
- [12] W3C. Composite Capabilities / Preferences Profile (CC/PP). <http://www.w3.org/Mobile/CCPP>.
- [13] Wapforum. User Agent Profile (UAProf). <http://www.wapforum.org>.
- [14] Held, A., Buchholz, S. & Sachill, A. (2002) Modeling of context information for pervasive computing applications, In *Proceedings of SCI 2002/ISAS 2002*
- [15] Indulska, J. & Robinson, R. (2003). Experiences in using cc/pp in context-aware systems, In *LNCS 2574: Proceedings of the 4th International Conference on Mobile Data Management (MDM2003)* (Melbourne/Australia, January 2003), M.S. Chen, P. K. Chrysanthis, M. Sloman, and A. Zaslavsky, Eds., *Lecture Notes in Computer Science (LNCS)*, Springer, pp. 247–261.
- [16] Chtcherbina, E. & Franz, M. (2003). Peer-to-peer coordination framework (p2pc): Enabler of mobile ad-hoc networking for medicine, business, and entertainment, In *Proceedings of International Conference on Advances in Infrastructure for Electronic Business, Education, Science, Medicine, and Mobile Technologies on the Internet (SSGRR2003w)* (L'Aquila/Italy, January 2003).
- [17] Erfianto B, Mahmood, A. K. & Rahman, A. S. A. (2007, December 11-12). Modelling context and exchange format for context-aware computing. *Proceedings of 5th student conference on research and development*. Malaysia
- [18] Henriksen, K., Indulska, J., Modeling Context Information in Pervasive Computing Systems. 167–180, 2002
- [19] Bauer, J. (2003, March). Identification and Modeling of Contexts for Different Information Scenarios in Air Traffic. *Diplomarbeit*
- [20] Halpin, T. A., (2001). *Information Modeling and Relational Databases: From Conceptual Analysis to Logical Design*, Morgan Kaufman Publishers, San Francisco.
- [21] Mohan, P. & Singh, M. (2013). Formal models for context-aware computing. *International Journal of Computer Applications Technology and Research (IJCATR)*, 2(1), 53-58
- [22] Henriksen, K., Indulska, J., and Rakotonirainy, A. (2003). Generating Context Management Infrastructure from High-Level Context Models, In *Industrial Track Proceedings of the 4th International Conference on Mobile Data Management (MDM2003)* (Melbourne/Australia, January 2003), pp. 1–6.
- [23] Ameyed, D., Miraouli, M. & Tadj, C. (2016). Spatiotemporal context modelling in pervasive context-aware computing environment: A logic perspective.

International Journal of Advanced Computer Science and Applications, 7(4), 407-414

- [24] Cheverst, K., Mitchell, K. & Davies, N. (1999). Design of an object model for a context sensitive tourist GUIDE, *Computers and Graphics* 23(6), 883–891
- [25] McCarthy, J. (1993). Notes on formalizing contexts, In *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence* (San Mateo, California, 1993), R. Bajcsy, Ed., Morgan Kaufmann, pp. 555–560.
- [26] McCarthy, J. & Buva. (1997). Formalizing context. In *Working Papers of the AAAI Fall Symposium on Context in Knowledge Representation and Natural Language* (Menlo Park, California, 1997), American Association for Artificial Intelligence, pp. 99–135.
- [27] Akram V. & Surav, M. (1997). The use of situation theory in context modelling. *Computational intelligence*, 13, 427-438
- [28] Gray, P. & Salber, D. (2001). Modelling and using sensed context information in the design of interactive applications, in *Engineering for Human-Computer Interaction*, ed: Springer, 317-335.
- [29] Roman, M., Hess, C., Cerqueira, R., Ranganathan, A., Campbell, R. H., & Nahrstedt, k. (2002). A middleware infrastructure for active space. *IEEE pervasive computing*, 1(4), 74-83
- [30] Ranganathan, A. & Campbell, R. H. (2003). An infrastructure for context-awareness based on first order logic. *Personal and Ubiquitous Computing*, 7, 353-364
- [31] Gu, T., Pung, H. K. & Zhang, D. Q. (2005). A service-oriented middleware for building context-aware services. *Journal of Network and computer applications*, 28, 1-18
- [32] Horrocks, I., Patel-Schneider, P.F., F. van Harmelen, (2003). From SHIQ and RDF to OWL: The making of a web ontology language, *Journal of Web Semantics*, 1 (1), 7–26.
- [33] Chen, H., Perich, F., Finin, T. W., Joshi, A. (2004). SOUPA: Standard Ontology for Ubiquitous and Pervasive Applications, in: *1st Annual International Conference on Mobile and Ubiquitous Systems (MobiQuitous 2004)*, IEEE Computer Society
- [34] Zhang, D., Gu, T., Wang, X. (2005). Enabling Context-aware Smart Home with Semantic Technology, *International Journal of Human-friendly Welfare Robotic Systems*, 6 (4), 12–20
- [35] Bettini, C., Brdiczka, O., Henricksen, K., Indulska, J., Nicklas, D., Ranganathan, A. et al. (2010). A survey of context modelling and reasoning techniques. *Pervasive and Mobile Computing*. 6, 161-180
- [36] Perera, C., Zaslavsky, A., Christen, P. and Georgakopoulos, D. (2014). Context aware computing for the internet of things: A survey. *Communications Surveys & Tutorials*, IEEE, 16, 414-454