INTERNATIONAL JOURNAL OF ADVANCED STUDIES
IN COMPUTER SCIENCE AND ENGINEERING
IJASCSE VOLUME 5 ISSUE 11, 2016
11/30/2016

A Logic Based Context Modeling and Context-aware Services Adaptation for a Smart Office

Moeiz Miraoui
College of computer at Al-Leeth, Umm Al-Qura University, KSA
mfmiraoui@uqu.edu.sa

Abdulbasit Zaid Abed
College of computer at Al-Leeth, Umm Al-Qura University, KSA
azabid@uqu.edu.sa

Sharif El-etriby
Faculty of computers and information, Menoufia University, Egypt
sherif.el-etriby@ci.menofia.edu.eg

Chakib Tadj
Ecole de Technologie Superieure, University of Quebec, Montreal, QC, Canada
chakib.tadj@etsmtl.ca

***Abstract*—Most of recent offices are equipped with a set of appliances in order to improve worker's comfort. However, many workers become frustrated with the difficulty of using the complex functions of their appliances by spending a non-negligible time in configuring and setting these appliances which prevent them from focusing on their main tasks which directly affect their productivity. Context-awareness is a key feature of services adaptation in smart spaces in general and in smart offices in particular. Context modeling is the first step in building context-aware systems. Previous work of context modeling especially those based on logic suffer from being not based on a clear and concise definition of context. In this paper we propose a logic based approach for both context modeling and context-aware services adaptation in order to render offices more intelligent which improve workers comfort, energy saving and workers productivity. Our approach is based on a clear and concise definition of context as well as on a generic predicate for context description and proactive services adaptation.**

Keywords-Smart office; Context; Logic; Modeling; Service; Adaptation.

Introduction

The widespread of devices with ability of computing and networking has led to the era of pervasive and ubiquitous computing. In such systems, the devices dynamically and proactively (without explicit user's intervention) adapt their behavior to current context changes. Smart spaces or intelligent environment or ambient intelligence are recent research fields under the banner of pervasive and ubiquitous computing. The main objective of smart spaces is to automate the environment customization, enabling it to adapt to user's needs according to the current context which helps on making life easier and more convenient. So far, there is no explicit definition of smart spaces, the most well-known definition of smart spaces is the one gave by D. Cook and S. Das in the MAVHome project in their book [2]: "Smart space is able to acquire and apply knowledge about its environment and to adapt to its inhabitants in order to improve their experience in that environment». Smart spaces are open, distributed, heterogeneous pervasive computing systems. The number of smart appliances and devices in the home, office or other private or public closed spaces has grown dramatically in recent years. Unfortunately, each component usually performs a single function and there is no synchronization with other components or the environment. The principal objective of researches in this field is to move from environments filled with smart devices and appliances to smart environments. Context-awareness is an important feature of smart spaces which allow them to enhance their autonomy and proactively provide services on behalf of users. The first step in the development of context-aware systems consists of understanding context and representing its elements in a concise and clear manner. Several methods were proposed for context modeling having particularities related to the techniques used. In spite of its high level of formality and expressiveness, few previous works on context modeling was done using logic. The proposed approaches suffer from two main weakness points: i) not based on clear definition of context, ii) proposed predicates for context do not cover all aspects of context. Our aim in this paper is to propose a logic based context modeling and reasoning for a closed smart which helps to improve reducing context complexity and enhance its usability in order to ease the development of context-aware applications. The proposed approach will be applied on an example of closed smart space namely the smart office.

The rest of this paper is organized as follows. Section I provides some background information about related work on logic based context modeling. Section II describes the overall environment of an exemplary smart office. In section III we discuss details of context modeling issues addressed by our approach. Section IV presents a logic based context-aware services adaptation approach. The conclusion and future work will be given in section V.

I. RELATED WORK

Context modeling is a fundamental step for the development of context-aware systems. The existence of well-designed context models will ease the construction of such systems. Context modeling consists of analyze and design of contextual information contained in the system in an abstract form on the level of data structure as well as the semantic level. Several approaches were proposed for the representation of context. Strang et al. [1] surveyed the most relevant approaches for context modeling. They concluded that ontology makes the best description of context because it provides a good sharing of information with common semantics. However, this does not mean that the other approaches are unsuitable for ubiquitous computing environment. Bettini et al [2] discussed the requirements that context modelling and reasoning techniques should meet followed by a description and comparison of current context modelling and reasoning techniques. They have selected a requirements set for context models. They did not mention logic based context model, instead they introduced hybrid approaches as an attempt to combine different formalisms and techniques to better fulfill the identified requirements. Perera et al. [3] surveyed context awareness from an Internet of Things perspective. They discussed context modelling techniques at a high-level. Their focus was on conceptual perspective of each modelling technique no on specific implementation. In their conclusion they mentioned that logic based modelling provides much more expressive richness compared to the other models However, lack of standardization reduces their re-usability and applicability. Most important, they concluded that incorporating multiple modelling techniques is the best way to produce efficient and effective results, which will mitigate each other's weaknesses. Therefore, no single modelling technique is ideal to be used in a standalone fashion. Most of the previous work has focused on ontology based context modeling and less effort has been spent on logic based context modeling. In the following, we are going to focus only on related work based on context modeling approaches using logic. Among the earliest work on logic based context modeling we can cite the one done by Carthy & Buvac [4] where they introduced the context as a formal object. Their objective was to define simple axioms for phenomena with a common sense and to treat the context associated with a particular situation. The basis of this relationship approach is the formula ist (c,p) which means that the proposition p is true in the context c. it defines such phrases as: c0: ist (context-of (Sherlock Holmes stories), Holmes is detective).This model also uses the notion of inheritance. Ranganathan & Campbell [5] proposed a model of context that is based on first order predicate calculus. The first order model allows complex rules involving contexts to be written. It also enables automated inductive and deductive reasoning to be done on contextual information. Their model for context describes the properties and structure of context information and the kinds of operations that can be performed on context. The name of the predicate is the type of context that is being described. It is also possible to have relational operators inside predicates. The predicate form is not general and the meaning and number of parameter depends on the described context element. The context model makes no restriction on the types of values that different arguments in the context predicate can take. So, predicate arguments can be arbitrarily complex structures. One or more arguments of a context predicate can be functions (written in C) that return some value. They used rules to deduce new contexts based on existing contexts. Roman et al. [6] presented an experimental middleware infrastructure called Gaia (an Active Space System Software Infrastructure) where they used a model for context that is based on first order logic and Boolean algebra, which allows them to easily write various rules to describe context information. They represented context through a 4-ary predicate, whose structure is borrowed from a simple clause in the English language of the form <subject><verb><object>. An atomic context predicate is defined in the following way: Context (<ContextType>, <Subject>, <Relater>, <Object>) e.g. Context (location, chris, entering, room 3231). In some cases, one or more elements of a predicate may be empty. It is possible to construct more complex contexts by performing first order logic operations. Gu et al. [7] proposed a Service-Oriented Context-Aware Middleware (SOCAM) architecture for the building and rapid prototyping of context-aware services. In their model, contexts are represented as first-order predicate calculus. The basic model has the form of Predicate (subject, value). The structures and properties of context predicates are described in an ontology. The ontology is written in OWL as a collection of RDF triples, each statement being in the form (subject, predicate, object). Nalepa & Bobek [8] proposed a new rule-based context reasoning platform tailored to the needs of an intelligent distributed mobile computing devices. They made a comparison of existent context modeling approaches and they took into consideration the following aspect of context modeling methods: formalization, simplicity, expressiveness, support for inference, handling of uncertainty, and existing tools that support design. They also proposed an inference service that uses HeaRT inference engine to provide on-line efficient reasoning for mobile devices.

In spite of the high level formality of logic, less efforts have been spent on logic based context modeling and most of previous work on context modeling have been centered on ontology based context modeling. The previous proposed work on logic based context modeling suffer from two main weakness points: i) the context predicates are not enough generic and their components are not fixed and vary according to the predicate usage. ii) Predicates components do not cover all aspects of context because they are not based on a clear and concise definition of context which limit their usage to some specific applications and negatively affect the expressiveness.

II. SMART OFFICE DESCRIPTION

Most people spend a considerable amount of time working in spaces like offices. Increasingly offices appliances and equipment have embedded computing and communication ability giving them a certain degree of intelligence. However, many users become frustrated with the difficulty of using the complex functions of their appliances by spending a non-negligible time on configuring and setting these appliances which prevent them from focusing on their main tasks. The general objective of research on smart office is to fulfill the office worker's requirements for comfort while reducing energy which enables them to work in a more efficient way. C. Le Gal [9] defined a smart office as an environment that is able to help its inhabitants to perform everyday tasks by automating some of them and making the communication between user and machine simpler and effective. Marsa-Maestre et al. [10,11] defined smart offices as an environment that is able to adapt itself to the user needs, release the users from routine tasks they should perform, to change the environment to suit to their preferences and to access services available at each moment by customized interfaces.

In this paper we will focus on one kind of offices namely the one-person office. An exemplary smart office is composed of a set of equipment of three main types: a) appliance which can be smart heater and air conditioner, b) Furniture which can be window blinds, a set of lighting lamps, office desk and office chair, c) communication device which can be a land-line phone or the user's mobile phone and d) a computer desktop. All these devices should provide a set of services through different forms (or modes) to the user occupying the office. These services should take into account the user's preferences and should be triggered according to the current context collected from different sensors installed in the office (Figure 1).

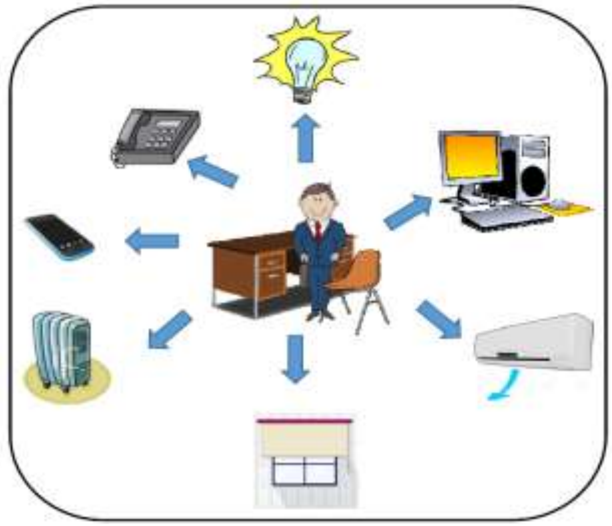

Figure 1. Appliances and devices in an exemplary office.

III. CONTEXT DEFINITION AND MODELING

A. *Context Definition*

Context-awareness is an important feature of smart spaces which allow them to support natural interaction and provide proactive services (without explicit user intervention). A clear and complete definition of context is the cornerstone of a context-aware systems. Several researchers have proposed definitions of context, some of which were based on enumerating contextual information (localization, nearby people, time, date, etc.) like those proposed by [12, 13, 14]. Others were based on providing more formal definitions in order to abstract the term, like the one proposed by Dey [15]. Most of these definitions were specific to a particular domain, such as human-computer interaction and localization systems. In our previous work [16, 17], we have made a survey of existing definitions of context and proposed a service-oriented definition of context for a pervasive and ubiquitous computing environments as follows: *"Any information that triggers a service or changes the quality (form or mode) of a service if its value changes."* This definition is sufficiently abstract and helps to limit the set of contextual information. We believe that this definition is more expressive, because it is simple, complete and covers all aspects of context. The definition was based on the concept of service because such one plays a crucial role in the operation of a pervasive and ubiquitous computing system and can be easily adopted in the context of smart spaces.

The aim of context modeling is to provide an abstraction of context information from technical details of context sensing which allows an easy context management and more flexible sharing among devices and appliances. As mentioned early in this paper, there has been a significant amount of effort in the field of context modeling especially ontology based approaches however less of efforts has been applied on logic based modeling in spite of the high level of formality and expressiveness of logic. Before we tackle the modeling process we should identify context elements in a smart office which is a basic step in such process. The proposed approach for context modeling consists of identifying context elements and then proceed to its modeling using a generic logic predicate.

Based on the adopted definition of context, the process starts (first step) by specifying for each equipment, appliance or device of the smart office the set of services that can be provided. In addition, for each service we should specify also the set of information which their change of values will trigger the service (Table I). The unique information that trigger all of the office appliances and devices is the worker's presence which can have two main values: present or absent. Worker can be present in the office either working at desk with PC or without PC. Worker can be absent temporary for a break time, a meeting, etc. or absent.

TABLE I. SERVICES TRIGGERING INFORMATION

Device or appliance	Service	Triggering information
Window blinds	Lighting	Worker's presence
Light	Lighting	Worker's presence
Heater	Heating	Worker's presence
Air conditioner	Air cooling and conditioning	Worker's presence
Phone	Communication	Worker's presence
Computer desktop	Work assistance	Worker's presence

The second step consists of specifying for each service the set of forms through which the services can be provided. We should also specify for each form of service the set of information which their change of values will change the form of a service. The set of these forms are fixed according to how the office should operate (Table II).

TABLE II. SERVICES FORM CHANGING INFORMATION

Device or appliance	Service's forms	Form changing information
Window blinds	Closed, mostly closed, half opened, mostly opened, totally opened	Worker's presence, outdoor light, worker's preferences
Light	Off, on low, average, on high	Worker's presence, outdoor light, worker's preferences
Heater	Off, on preferred heat, on other	Outdoor temperature, worker's preference
Air conditioner	Off, on preferred cool, on other	Outdoor temperature, worker's preference
Phone	Default mode, divert to answer machine, divert to mobile phone, divert to SMS indication on worker's mobile phone	Incoming call, Worker's presence, worker's schedule, date
Computer desktop	Shut down, sleep, hibernate, On	Worker's presence, seated worker, date

The third step consists of making the union of the two previous sets to get the final list of contextual information. This information will compose the global context and in our case will be composed of the following elements with their possible values (Table III).

TABLE III. CONTEXT ELEMENTS AND THEIR POSSIBLE VALUES

Context elements	Possible values
Worker's presence	Absent, temporary absent, present
Seated worker	Working at desk with PC, working at desk without PC, No
Date	Work day, week-end
Outdoor light	Dark, low, average, high
Outdoor temperature	Low, average, high
Incoming call	Yes, no
Worker's preferences	To be fixed according to the worker
Worker's schedule	Daily tasks, meeting, temporary exit, vacation

B. *Context Modeling*

Our context model will be based on first order logic by using a first order logic predicate with three arguments for each context element. The context predicate is defined as follows:

Context_element (<entity>, <state/value>, <time>)

The entity refers to any context element which can be worker, temperature, window blind, etc. Depending on the entity type, he state/value refers either worker and in this case it will take the state argument or other context entity and in this case it will take the value argument because a worker has states not values. The last argument of the predicate refers to time either a specific moment or a time interval (morning, afternoon, etc.). Using this predicate, we can express a large variety of basic context or even complex contexts using a compound statements of the context_element predicate. The time is an essential argument of the context_element predicate because all of the context elements vary according to the time in a fixed space (office).

Some example of context_element predicate includes the following:

- The worker is working at desktop with PC in the afternoon
  Context_element (Seated worker, working at desktop with PC, 1:00 PM <= T<= 6:00 PM)

- The temperature is high in the office at 10:00 AM
  Context_element (Temperature, high, 1:00 AM)

- The light is very low in the office in the morning *Context_element (Indoor Light, very low, 5:00 AM <= T <= 11:59 AM)*

- It is sunny outside the office at 3:00 PM *Context_element (Outdoor Light, high, 3:00 PM)*

- The worker is in a meeting from 2:00 PM to 3:30 PM *Context_element (worker's presence, Temporary absent, 2:00 PM <= T <=3:30 PM)*

Complex contexts can also be modeled based on the context_element predicate by performing compound predicates using logic operations such as conjunction, disjunction and negation. For example, the following context:

The worker is working at desktop without PC in the morning office hours and the temperature is very low inside the office in the morning office hours and dark sky outside the office all day and there is no incoming call for the worker all the morning office hours.

Can be expressed as follows:

Context_element (Worker's presence, present, 8:00 AM <= T<= 11:59 AM) ∧ Context_element (Seated worker, working at desktop without PC, 8:00 AM <= T<= 11:59 AM) ∧ Context_element (Temperature, very low, 8:00 AM <= T <= 11:59 AM) ∧ Context_element (Outdoor Light, Dark, 8:00 AM <= T<= 6:00 PM) ∧ ¬ Context_element (Phone, incoming call, 8:00 AM <= T<= 11:59 AM).

Existential quantifier (there exists) and universal quantifier (for all) can also be used to describe some context situation when one or more of the context_element predicate is a variable. For example to describe the statement "whatever the value of light outside the office during working hours", we can write the following predicate:

∀ X context_element (Outdoor light, X, 8:00 AM <= T<= 6:00 PM)

To express that there is a period of time where the temperature inside the office is low, we can write:

∃ X context_element (temperature, low, X)

IV. CONTEXT-AWARE SERVICES ADAPTATION

The main objective of building smart offices is to render appliances and devices of the office operating proactively with minimum worker's intervention keeping him focusing on his main work tasks. Based on the previous definition and modeling of context which use predicate logic, we propose a context-aware services adaptation using a rule based system. The central element of the adaptation task is the worker which can be in one of the following states:

- The worker is absent in a vacation, week-end or out of office hours (e.g. at night)
- The worker is present at his office but not working on his desk
- The worker is present at his office and working on his desk
- The worker is present at his office and working on his desk with his desktop computer
- The worker is temporary absent from his office either for lunch break, to attend a work meeting or temporary exit related to his work.

For each of the above worker's state, the appliances and devices should adapt their provided services according the current context, worker's preferences and daily schedule. The adaptation task is a mapping from the current context, worker's preferences and daily schedule to a configuration vector of appliances and devices. We have used the PROLOG programming language for the implementation of the adaptation rules and the SWI-Prolog as a programming environment. In the following we present some extracts of the database adaptation rules used with brief descriptions.

The worker is absent in a vacation, week-end or out of office hours (e.g. at night)

In this case, all appliances and equipment shout be shut down to save energy and safety of the office.

(Worker's presence, Absent, 6:00 PM <= T<= 8:00 AM) V ∀ X (Date, Week-end, X) V ∀ Y (Worker's schedule, vacation, Y) ⇒ ∀ Z (Window blinds, totally closed, Z) ∧ (Light, Off, Z) ∧ (Heater, Off, Z) ∧ (Air conditioner, Off, Z) ∧ (Phone, Divert to answer machine, Z) ∧ (Desktop computer, Shut down, Z).

The worker is present at his office and working on his desk

One possible context in this case is the following:

∀ T [(Worker's presence, Present, T) ∧ (Seated worker, Yes, T) (Date, Work day, T) ∧ (Schedule, Daily tasks, T) ∧ (Outdoor light, High, T) ∧ (Outdoor temperature, High, T) ∧ (Incoming call, No, T) ⇒ (Window blinds, Mostly closed, T) ∧ (Light, Off, T) ∧ (Heater, Off, T) ∧ (Air conditioner, On preferred cool, T) ∧ (Phone, Default incoming call indication, T) ∧ (Desktop computer, Sleep, T)].

The worker is temporary absent from his office either for lunch break, to attend a work meeting or temporary exit related to his work.

In this case, the worker is out of his office for a limited period of time within the office hours period either for lunch break (generally between 11:59 AM and 1:00 PM) or his daily schedule indicates that he has a meeting or has to leave his office for an external job. One possible context in this case is the following:

$\forall$ *T(([(Schedule, Meeting, T) V (Schedule, Temporary exit, T)] V (Worker's presence, Absent, 11:59 AM <= T <= 1:00 PM))* $\wedge$ *(Outdoor light, Average, T)* $\wedge$ *(Outdoor temperature, Low, T)* $\wedge$ *(Incoming call, Yes, T)* $\Rightarrow$ *(Window blinds, Half-opened, T)* $\wedge$ *(Light, Off, T)* $\wedge$ *(Heater, On preferred heat, T)* $\wedge$ *(Air conditioner, Off, T)* $\wedge$ *(Phone, Divert to mobile phone, T)* $\wedge$ *(Desktop computer, Hibernate, T)).*

V. CONCLUSION

An office worker spends a considerable time in configuring and setting equipment, appliances and devices of his office. In this paper we have presented a logic based approach for context-aware services adaptation in an exemplary office in order to render it more intelligent which improve worker's performance and comfort. Firstly, we have made a description of the content of an exemplary office. Secondly, we have identified the main components of context inside the office. Thirdly, we have modeled this context using a simple logic predicate. Finally, we have proposed a rule base system for context-aware services adaptation. Our future work consists of expanding our approach to offices with more than one worker.

ACKNOWLEDGMENT

This work is supported by the King Abdulaziz City for Science and Technology (KACST), KSA under Grant AC-35-95.

REFERENCES

[1] T. Strang and C. Linnhoff-Popien "A Context Modeling survey", In the first International Workshop on Advanced context modeling, Reasoning and management, UbiComp 2004

[2] C. Bettini, O. Brdiczka, K. Henricksen, J. Indulska, D. Nicklas, A. Ranganathan, D. Riboni, A survey of context modelling and reasoning techniques, Journal Pervasive and Mobile Computing, Vol.6(2), pp. 161-180, 2010

[3] C. Perera, A. Zaslavsk, P. Christen, and D. Georgakopoulos, Context Aware Computing for The Internet of Things: A Survey, IEEE communications surveys & tutorials, VOL. 16, NO. 1, pp. 414-254, 2014

[4] J. Mc Carthy, and S. Buvac. 1994. Formalizing context (expanded notes). . Coll. « STAN-CSTN-94-13 ». Stanford University.

[5] A. Ranganathan and H. Roy Campbell, An infrastructure for context-awareness based on first order logic, pervasive and ubiquitous computing, Springer vol. 7, pp. 353-364, 2003

[6] M. Roman, C.K. Hess, R Cerqueira, A. Ranganathan, Campbell RH and Nahrstedt K (2002) Gaia: a middleware infrastructure to enable active spaces. IEEE Perv Comput 10-12:74–83

[7] T. Gu, H. K. Pung, D. Q. Zhang, A service-oriented middleware for building context-aware services, Journal of Network and Computer Applications, elsevier, vol. 28, pp.1–18, 2005

[8] J. G. Nalepa and S. Bobek, Rule-Based Solution for Context-Aware Reasoning on Mobile Devices, Computer Science and Information Systems 11(1):171–193, 2014

[9] C. Le Gal, Smart Environments: Technology, Protocols and Applications, Wiley, 2005

[10] I. Marsa-Maestre, E. de la Hoz, B. Alarcos, J.R. Velasco (2006) A hierarchical, agent-based approach to security in smart offices. In: Proceedings of the First International Conference on Ubiquitous Computing (ICUC-2006)

[11] I. Marsa-Maestre, M. A. Lopez-Carmona, J.R. Velasco, A. Navarro (2008) Mobile agents for service personalization in smart environments. Journal of Networks (JNW) 3(5):30–41

[12] S. Schilit and M. Theimer "Disseminating Active Map Information to Mobile Hosts", IEEE Nenvork, 8(5):22-32. 1994

[13] P.J. Brown, J.D. Bovey and X. Chen "Context-aware Applications: From the Aboratory to the Marketplace" IEEE Personal Communications, 4(5):58-64. 1997

[14] N. Ryan., J. Pascoe and D. Morse "Enhanced Reality Fieldwork:the Context -Aware Archeological Assistant" Computer Applications in Archeology. 1997

[15] A.K. Dey "Understanding and Using Context" Journal of Personal and ubiquitous computing, Vol. 5, pp. 4-7. 200 1

[16] M. Miraoui, C. Tadj, A service Oriented Definition of Context for Pervasive Computing, in Proceedings of the 16th International Conference on Computing, Mexico city, Mexico, Nov. 2007.2007.

[17] M. Miraoui., C. Tadj, and C.B. Amar, Context Modeling and ContextAware Service Adaptation for Pervasive Computing Systems. International Journal of Computer and Information Science and Engineering, 2008. 2(3): p. 148-157.