

A PERVASIVE SYSTEM ARCHITECTURE FOR SMART ENVIRONMENTS

Imen Ismail and Faouzi Moussa

CRISTAL Laboratory, National School of Computer Sciences,
Manouba University, Tunis

imen_ismail@yahoo.fr faouzimoussa@gmail.com

ABSTRACT

The adaptation paradigm is a basic defining feature for pervasive computing systems. Adaptation systems must work efficiently in smart environments while providing suitable information relevant to the user-system interaction. The key objective is to deduce the information needed to fulfil the current activity according to the context of use. However, as the context changes needed information changes. Therefore relying on fixed operational models would be inappropriate. Hence, the proposition of RADEM, a flexible pervasive system architecture that has been designed to meet the aforementioned challenges, is suggested. This would provide dynamic and realtime functional models of the user-system interaction. The paper explores also some key issues related to pervasive computing systems. An analysis of related work is being conducted in order to provide key criteria and requirements for such intelligent systems.

KEYWORDS

Pervasive Systems, RADEM Architecture, Smart Hospital, Current Context of Use, Adaptation

1. INTRODUCTION

Due to the emergence of studies on pervasive computing, the capacity to build systems that adapt the delivered information and knowledge to meet the requirements of users in smart environments has attracted quite a bit of attention [1][2]. The primary objective is to improve the interaction user-system quality in such smart environments [3][4]. This technology induces an increasing interest from studies relating to mobile user interface, user-system interaction and ambient intelligence [5][6]. The realtime adaptation of the provided information to suit needs, preferences and distinct behaviours of different users still constitutes the challenge and is still not adequately addressed in these systems. The challenge is to know what information is most relevant for the user, when it will be need and how the information should be displayed on the devices interface [7]. Delivering content in such dynamic environments may require more operational flexibility to provide data accordingly; a static method of operation is certainly inadequate. This complexity arises when the provided data must fit properly with specific and individual features of each user. To address these issues, architecture (RADEM) suitable for continually updating the user's changing requirements is proposed. This indented architecture performs a dynamic and realtime adaptation while maintaining maximum flexibility. It takes into account the current context of use and dynamically creates a proper functional model and thus deduces the plausible information expected to fulfil the user's requirement according to that model.

The second section of this article discusses some literature reviews and lays out the requirements that need to be respect in a pervasive system functioning. The remainder of the paper is structured

as follows: section 3 gives an overview of the proposed architecture. Section 4 gives further details on the adaptation process. The implementation of the case study is described in section 5. In the last section, the paper provides a short conclusion and overview of future work.

2. RELATED WORK

The beginning researches on pervasive systems have focused on the contextual data management and context modelling. That is why the contextual data processing layer was highlighted in the initial related work. Researchers have been deeply interested in the optimal arrangement for managing and modelling this layer [8][9][10][11]. Thus, the first systems were largely involved the interfacing between the contextual data management and the pervasive applications. The adaptation's concept in these systems has particularly concerned the adaptation to contextual situations. Nevertheless, in these solutions the adaptation strategies are not detailed or they are completely neglected. For example, the main goal of CIS¹ architecture [9] is to collect, to model and to provide the contextual information. The SOCAM² architecture [10] was especially proposed to convert the physical spaces, thus contextual information can be converted into a semantic space and can be shared between the context-aware services. Moreover, the key component in the CoBrA³ architecture [11] is responsible for managing and processing the contextual information while maintaining the contextual model. However, a number of recent architectures [12][13][14] regard as a major priority the adaptation stage in such architecture, given with specific adaptation strategies.

2.1 Requirement analysis of the adaptation systems

After careful review of definitions referring to pervasive computing [15][16][17][18], three key factors which should govern the adaptation process have been deduced. First, the adaptation of the provided information must be at realtime. In fact, according to his context of use and in addition to meeting his preferences, the user needs realtime information. Secondly, the current activity should be addressed and evaluated in the adaptation process. The aim is to continuously improve the quality of adaptation while providing users with the suitable and necessary information to effectively carry out his current activity. Finally, we must pursue a user-centred adaptation process. Thus, the adaptation must be specific to the user's profile, his subjective preferences, his individual and intellectual characteristics, etc.

The following table (Table.1) describes the meeting of these criteria in some related works. First, we note that these criteria are not well-referred in the first architectures. However, current research is sign that a meeting of these issues is beginning to emerge and realize how important they are for improving the system's capacity to meet essential requirements. With regard to the realtime factor, notwithstanding its obvious importance, it has been neglected. Whereas, for the CAAMDTA⁴ platform this factor was not explicitly mentioned, but we have implicitly deduced it from the operation nature of the system. Similarly, introducing user's preferences in the adaptation process is not highlighted in most of studied work. Nevertheless, the Context-Provider component of the SECAS⁵ platform is responsible for managing contextual data relating to a user's profile and his history, but it's not actually be considered to conduct the adaptation process. The CAAMDTA platform provided a module responsible for deducing high-level data relevant for the physiological status of users and according to their movements.

¹ Context Information Service

² Service-Oriented Context-Aware Middleware

³ Context Broker Architecture

⁴ User-Adaptive and Context-Aware Architecture for Mobile and Desktop Training Applications

⁵ Simple Environment for Context-aware Systems

2.2 Discussion

By analysing related work, we note that the contextual data management was the most highlighted layer considered a priority feature of a pervasive system. Then, the adaptation layer was progressively becoming crucial in such systems, even going to mapping contextual data functionality to the adaptation layer (case of CAASS platform). The literature survey has allowed us to disclose the fixed based-model that conducts the whole functioning of reviewed systems. On the basis of this background, we propose a pervasive system while addressing these aforementioned problems. As a result, investing architecture with a high sense of flexibility that enable a realtime and dynamic adaptation is an extremely important part of our work.

Table 1. Meeting of identified criteria in some related works

	Current Activity Consideration	Dynamic and Flexible functioning	User's Preferences consideration	Realtime Adaptation	Remarks on adaptation's mechanism
Context Information Service (1999)	No	No	No	No	The "monitor" component looks for the sensor that can provide adequate information.
Context Toolkit (2000)	No	No	No	No	Context adaptation to the application layer's requests.
HyCon (2003)	No	No	No	No	Physical Adaptation
CoBrA (2004)	No	No	Confidentiality's policy	No	Reasoning on contextual data
JCAF (2005)	No	No	No	No	The « Service client » ensures the adaptation but merely an adaptation to context.
SOCAM (2005)	General activities but not the current one.	No	No	No	The "context-aware services" change their function according to the context change.
SECAS (2006)	General activities but not the current one.	No	Yes	No	Content Adaptation Presentation Adaptation
CAAMDTA (2008)	Yes	No	Yes	Yes	Content Adaptation Presentation Adaptation
CAASS (2011)	No	No	No	No	Content Adaptation

3. OVERVIEW OF THE PROPOSED ARCHITECTURE

The RADEM system, which stands for Realtime Adaptation based on Dynamic and Evolutionary Models, comprises three functional phases that roughly constitute the adaptation process. This section provides an overview of these phases and describes the structures of its components. But first it's important to elucidate the motivations and goal settings of the intended architecture.

3.1 Motivations and goals of the proposed architecture

The goal of our work is to build a standards-based architecture that provides flexibility and scalability of pervasive systems in smart environments. There is a need to dynamically create models and enable them to be supported in such environments. For that reason, models used for the user-system interaction should be built with context-awareness capabilities, so that they can properly adapt to the changing context of a moving user. Pervasive systems must adapt the information it provides by implicitly deriving the user's requirements from his context of use, whereas, the context tends to vary at runtime in smart environments. The goal is to enhance the adaptation system's operation while improving models at runtime. Owing to the fact that fixed models cannot handle the high amount of information in such environments, the models should progress and change its structure to better match the user's requirements. Consequently, propose an efficient functioning constructed on the basis of dynamic models to cater to the runtime requirements of each mobile user is the primary challenge of RADEM architecture.

3.2 Functional Description of the RADEM Architecture

The functional description of the proposed architecture, given in figure 1, consists of three dependently functional modules: the knowledge repository module, the current context of use and the adaptation engine that is the main module. The retrieval mechanism is built on a realtime and dynamic user activity modelling. Oftentimes, the user is involved in a variety of activities over the course of his work. These can be routine activities, unexpected activities; they can be also activities that change according to working conditions, etc. In any case, the activity solicits specific information in order to be accomplished. Thus, the models should be built in proportion as runtime evolving of the pervasive system as well as according to available information. The following describes the conceptual model of the architecture contemplated to model the user-system interaction at real-time and according to the current context of use.

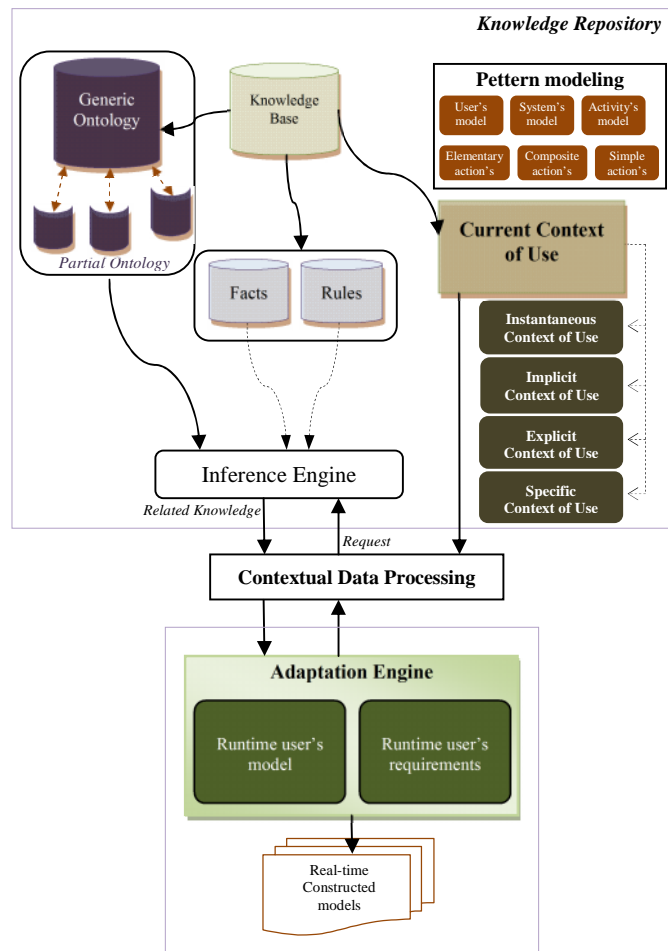


Figure 1. Overall Conceptual Model of RADEM System

3.2.1 The Current Context of Use Component

This module is responsible for classifying the current context of use. It represents an interface between the RADEM architecture and the smart environment. The purpose of such a classification [19] is to provide a high degree of comprehension so that particular required information will be consistently and easily found. This categorization can be examined to determine the requirements that are imposed on data management and can help to derive semantically relevant information. It can be viewed as the current representation of the smart environment, i.e. the knowledge about the environment. Its components are: “Specific context of use”, “Instantaneous context of use”, “Implicit context of use” and “Explicit context of use”. The specific context of use is a part of the current knowledge that determines which of the previously analysed data will be required for a specific situation. It can be viewed as information on request. Explicit data are required information where a conscious action by a user enables an intervention, whereas implicit data are information without any direct user intervention. Instantaneous data are information inferred instantly from the user’s undertaken activity.

3.2.2 The Adaptation's Engine Component

The RADEM architecture operation is mainly based on the adaptation engine module. It can be considered the core architecture for building dynamic models. It's responsible for assembling and providing the realtime and suitable models of the user. This is performed by the runtime user's model developer sub-module. Deducting the relevant user's requirements from the generated model represents the primary role. Upon the user start-up his activity and during his interaction with the ubiquitous environment, the system has to be based on the appropriate models to deduce the needed information and assemble the user interface on the fly. The adaptation engine also provides the basis of selecting and loading the appropriate model for the given user according to his current activity while considering the current context of use. Changing models as required, interpreting models at runtime and improving models based on adaptation strategies are other purposes of the adaptation engine module.

3.2.3 Knowledge repository Component

The Knowledge repository Component module provides a knowledge storing space. It organizes and categorizes various data including as much detail about the users, i.e. their preferences, their interests, their activities as well as any data that characterizes a user, essentially the user's histories and his past interaction with the system. The content of this repository will be increasingly enriched while memorizing and learning the user's interactions and behaviours. In the proposed architecture, the contextual data are expressed by ontology written in owl (Ontology Web Language) [20]. The system maintains a generic ontology. It can be shared by all users or rather the users who can have access to only part of this global ontology. This part, so-called partial view ontology, is closely related to the users' profiles. Thus, merely the required information, notably information related to the current activity will be provided. Moreover, to ensure accurate and updated data of the generic ontology, the system should be aware of the realtime variation of the contextual data and therefore makes necessary updates. The system should periodically synchronize the updated information between the generic ontology and the partial one.

Pattern Modelling Sub-Component

The Pattern Modelling sub-component is an abstraction of a high-level representation of the system especially the users and their interactions. The Petri-nets technology [21] was used as formal modelling to represent the users and their interactions. Thus, this module includes abstract models based on the formalism of Petri-nets modelling. The adaptation engine should generate a concrete and instant model of the user's activity or rather change the state of the current model according to the activity's change while relying on these formal abstractions. The generated model can be seen as the instances of well-defined metamodels and they are considered a collection of elementary actions (sub-models) with some properties. On the basis of these metamodel structures the adaptation engine must provide a Petri-nets basic model of the current activity with the minimum set of elementary actions. During the user-system interaction, the model is improved by inserting other elementary actions based upon existing rules.

Control Structures and Rules sub-component

This element encompasses rules and strategies that control the realtime model creation. An activity is a set of elementary actions arranged for typical compositions such as sequential, parallel, choice, iteration, etc. [23]. Developing the overall model of the user's activity is improved by the operational compositions of elementary action models and upon a well-defined

composition's rules. These rules perform, among others, the order and conditional execution of the action in the model.

To sum up, it should be apparent from the presented architecture that the system could have a suitable representation of the user in a particular situation in realtime. Hence, it can deduce the suitable model or build a realtime model according to the analysis of the user-system interaction and the current context of use. It can adaptively offer user information concerning what he is doing. A case study, aimed at illustrating the process described above, will be presented in the section 5 while highlighting the dynamic aspect of the proposed architecture.

4. THE ADAPTATION ENGINE FUNCTIONING

The adaptation engine is considered the reasoning part of the system. It continually supervises the update of the models as the user changes activities or as the current context of use changes. This supervision will be under the direct control of the user's interaction with the environment. Responding accurately to the user's requirements notably needed to fulfil the current activity, is the primary objective of the adaptation engine. However, the user's activity does not strictly follow a plan, but is very situation dependent. As activities may change according to working and contextual conditions, the main event that triggers and monitors the adaptation engine functioning is the current activity event. Figure 2 describes the architectural building blocks and specifies the relationship between these blocks.

The current context of use block is the principal intermediate between the adaptation engine and the environment. This component actively manages contextual data in an organized and strategic manner. It includes, among others, the context of use, the current activity and information related to specific interests and preferences of each user. In the light of this information, the adaptation engine monitors the construction of the appropriate model of the user's interaction. The dynamic composition of the model is closely related to particular user's characteristics. For certain activities the associated model, respectively the provided information, may differ from one user to another. Moreover, for particular activities the system generates different models according to several features, e.g. the user's preferences, individual interests and intellectual level. The current contexts of use as well as the corresponding activity are periodically checked. Whenever a change occurs and recognized by the adaptation engine, it is likely to result in a dynamic update of the activity model. This is followed by an automatic update of the related data including the dialogue history. The model update is either to ameliorate the current model structure or re-building an appropriate model. Past interaction and other knowledge sources represented in the knowledge models manage this dynamic composition too. Through this process, the user's requirement submodule maintains realtime data closely related to the current user's model. These data are implicitly dependent on current activity and the widget identification module retrieves the most suitable widget to meet the need. Finally, the interface generator creates the concrete interface for the target platform.

The widget is originally defined as a graphical representation displayed on the user interface. These objects will be dynamically identified and assembled on the fly under the adaptation engine control. This part supervises the realtime reorganization and layout of the required and available widgets. It's in synchronous relationship with the widget's identification module. The aim is to build the most fitting interface layout.

Contextual data in smart environments include a large and heterogeneous amount of information: the user's identity and the identities of people around him, his interests, his intellectual level, his preferences, his profession, his social status, the devices, the season, the date, the time, the light level, the room temperature, etc. Contextual parameters cannot be limited. It is necessary to

categorize this information to better match the various contextual parameters with varied requirements. The obtained categories are called "contextual parameters". Three main entities which characterize a pervasive and smart environment are identified [24][25][26][27]: places, people and objects. The *places* mean the geographic location, *persons* mean individuals or groups of people and *objects* describe the physical features or software components. To describe these entities, different taxonomies have been proposed. However, four main categories are identified by Dey [7] have been retained:

- Identity: represents an explicit identifier that characterizes an entity. It must be one-way defined in the application domain.
- Location: represents the geographical data of the entity (e.g. position and orientation) as well as spatial relations to other entities.
- Time: shows the date and time, e.g.: hours of work.
- Status: represent the different features that can be perceived by a user.

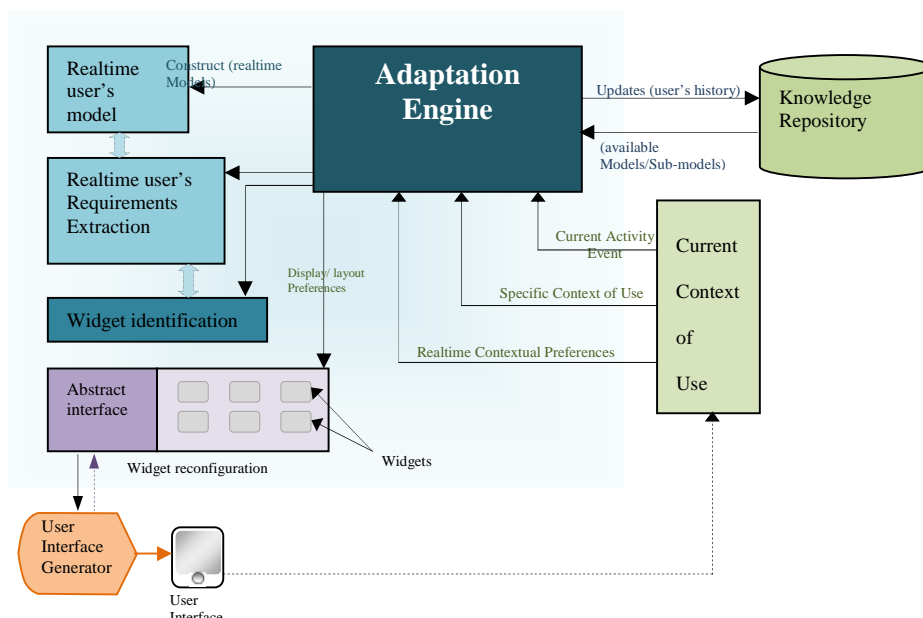


Figure 2. Overall functioning pattern of the Adaptation Engine

However, this classification does not sound quite helpful to identify the importance level of the information designed to be presented to the user. Hence, our proposed categorization aims at highlighting the type of information according to the user's requirements. The contextual data that help the user by giving him useful information is called Informational Parameters (*InfoParam*). Moreover, the user may need some judicious information to help him make a decision in a particular situation or under collaborative activities. This kind of information is called Decisional Parameters (*DeciParam*). Some information may not be interesting for the user, while it should be heavyweight informative. This kind of information is called Alert Parameters (*AlerParam*). Each of these types will be transmitted to the user interface through different forms according to their specific content as described below:

- Alergets: (Alerting-widgets) the content of these widgets is considered warnings to evoke an emergency state and thus request a direct user intervention.
- Infogets: (Informational-widgets) the content of these widgets is closely related to the user's requirements.

- Decigets: (Decisional-widgets) the content of these widgets consists of judicious data that enable smarter decisions.

Within this classification, the user interface content and presentation can give advice by smarter decisions and calls the user's attention by alerting him about a specific situation. This also may lead to improve the preciseness and the conciseness of the provided information. Finally, based on the Petri-nets model of the user activity and taking into account the identified contextual parameters, it will be easily to deduce all widgets needed for a useful user interface [23]. This is simply realized by associating the suitable widgets to each corresponding parameter (Figure 3).

5. IMPLEMENTATION AND RESULTS: CASE OF SMART HOSPITAL

Computer science technology applied to a personalized medicine's benefits greatly from research done in the pervasive computing area. For application and evaluation purposes we have tested the architecture described above with a view to simulate a smart hospital. We have particularly implemented a pervasive application for the realtime supervision of diabetic patients in a smart hospital. The main purpose is to meet the request of several users (nurses, doctors, paramedics, etc.) and return information according to the specific requirements of each user in realtime. A smart hospital can be viewed as a physical space which incorporates an important number of heterogeneous and smart devices creating a growing ubiquitous network (Figure 4). Examples include medical devices equipped with context-awareness capabilities, smart ambulances equipped with sensors, control sensors which are responsible for a realtime-remote-monitoring of each patient's state. These biological sensors periodically control the patient glucose level (GL) and perform other necessary tests. Doctors, nurses and paramedics are the primary users of the pervasive system. They should carry mobile devices with specific features and parameters enabling the interaction with the pervasive computing system.

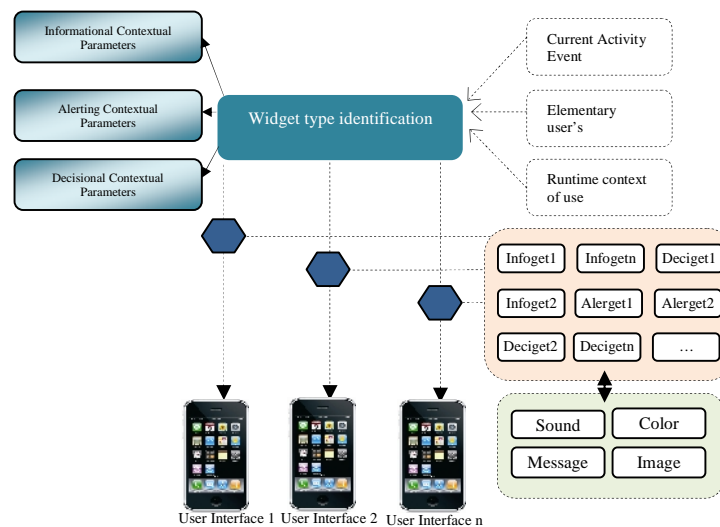


Figure 3. Interface layout identification based on widget type

Some scenarios are below presented to clearly outline the flexibility that characterizes the proposed system's operation. In this example, patients with hypoglycemia are particularly studied. Hypoglycemia is the most common diabetes-related emergencies. It can occur when the diabetic patient administers too much insulin or if he misses a meal. The pervasive system monitors each patient's state while analysing their clinical records and treatments. The glucose

level values are the most important factors that must be followed in the course of a severe hypoglycemia intervention. In the light of those values, the system must verify the evolution of each patient. A medical intervention depends necessarily on a patient’s recorded status. A set of actions that indicate the necessary steps to treat a diabetic with hypoglycemia has to be respected. The Table2 presents briefly these steps and identifies the contextual information needed at each stage.

Table 2. Required information according to each processing phase

Steps Nb	Necessary phases to treat a conscious patient	Required Information	Necessary phases to treat an unconscious patient	Required Information
1	Physical measure of the glucose level (GL)	UGL {Usual glucose level of the given patient}	Turn the patient on side if possible	
2	Give patient the suitable medicine if the GR is less than 4 mmol/l	UM {Patient’s specific medicine type, e.g. Sweet jelly}	Call doctor who takes care of the patient and follow his eventual recommendations	Doctor’s details {Name, Unit of service, Phone number PhoneNum, etc}
3	Call doctor who takes care of the patient and follow his eventual recommendations	Doctor’s details {Name, Unit of service, Phone number PhoneNum, etc}	Physical measure of the glucose level (GL)	UGL {Usual glucose level of the given patient}
4	Wait 10 to 15 minutes and make again a glucose measure	The remaining time RT	If GR is less than 4 mmol/l then give Glucagon to the patient	
5	If the GR is still less than 4 mmol/l then repeat the steps 2 and 3 until GR becomes more or equal than 4	{UM}	If the patient’s weight is less than 44lb (20kg): give him the half of the syringe. Else give him the full syringe	Patient weight {WGHT} Glucagon’s quantity {Half_Glu, Full_Clu}
6	Once the glucose level reaches the specified value or more, give the patient a carbohydrate or protein based-product	The product existing stock	If the patient has an insulin pump, disconnect it.	

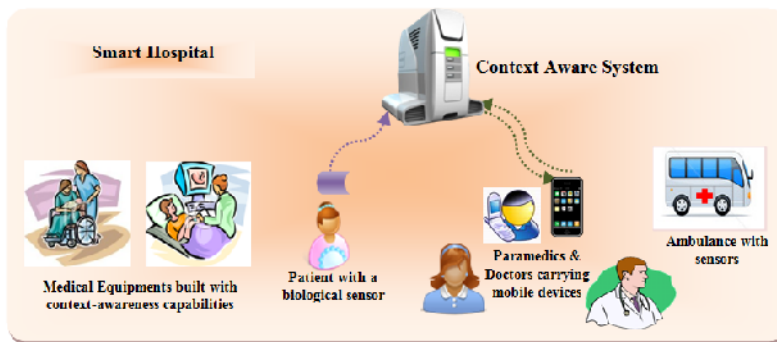


Figure 4. Smart hospital simulation

Initially, the simulation process is started by the detection of an abnormal GL. If during a periodic check, the system detects an abnormal GL of a given patient (e.g. less than 4 mmol/l), it deduces that he is suffering from hypoglycemia. Therefore, an urgent message will be spread throughout the entire hospital service (Figure 5). Once, one of the paramedic proceeds through a diffuse area,

this message will be automatically displayed in his device's interface as an emergency request. The pervasive system should provide all information necessary to offer the best care for the patient (Table2).

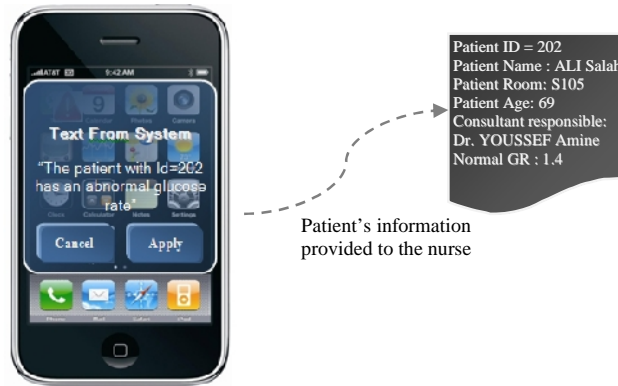


Figure 5. An an emergency message issued by the system

5.1 Adaptation to the context of use

This subsection illustrates how the adapted information must change according to the context of use, for instance the patient glucose level. Suppose that the patient became unconscious while the nurse was caring for him. The current functioning models are not suitable. The system must regenerate the appropriate model or re-adjust the current one in order to best meet the nurse's context of use. Let's take the fifth step according to the first situation (S1: conscious patient) the nurse needs the UM value as required information. If after a short time the patient's status changes to unconscious, the nurse will need as urgent data the patient's weight as well as the glucagon's quantity (Figure 6).



Figure 6. Required data adaptation according to the context change within the same user's conditions

5.2 Adaptation to the current activity

Let's take the required information in the third step of the conscious situation. The pervasive system must provide details about the doctor who takes care of the given patient. The nurse should obtain such details on his interface; the system can even enable the phone number with the objective of facilitating the call (figure 7).



Figure 7. Change of operation according to the current activity

5.3 Adaptation to a specific user

The adaptation has to take into account a particular user having different characteristics. In a heuristic evaluation, an adaptation is efficient only when it works for different kind of users even if they perform the same activity. If we consider the fifth step within unconscious situation, the pervasive system must tailor the information according to the nurse profile. Thus, if the nurse is hematology specialist, merely the patient's weight is required as contextual information. Otherwise, the glucagon's quantity value must be specified after it has been inferred based on the patient's weight (Figure 8).



(a) Message to a hematologist (b) Message to no-hematologist user

Figure 8. Required Information adaptation according to the user's characteristics

6. CONCLUSION AND FUTURE WORK

The development of pervasive computing systems whose behaviour must adapt to specific users and at realtime is still a challenge task. This paper outlines the architecture of a pervasive system that allows dynamical adaptation towards the challenges of the smart environment. Helping users to cope with their activity in realtime is the primary aim of the proposed approach. For that reason, the internal structure of the system is heavily and principally influenced by the current activity of the user. This can be achieved by introducing the dynamic models, which can be constructed in realtime to interact with the environment. This result is of considerable importance on its own, and is particularly relevant for inferring current needs from these dynamic models as a consequence once the system came to create the most suitable model of the current activity. The system could indeed have a suitable and realtime representation of the user in a particular

situation. It can consequently deduce the suitable model or build a realtime model according to the analysis of the user-system interaction and the current context of use.

A preliminary analysis of existing pervasive systems has been made. We conclude that the state-of-the-art does not provide sufficient adaptability towards number of key factors, especially the realtime aspect; even though this factor is considered the most important constraints of the pervasive systems. In fact, the realtime adaptation of the provided information to match needs, preferences and distinct behaviours of different users still constitutes the challenge and is still not adequately addressed in these systems.

The second part of this paper is concerned with describing a case study that has exemplified the dynamic aspect of the proposed architecture. The main purpose is to meet the request of several users in a smart hospital (nurses, doctors, paramedics, etc.) and return information according to the specific requirements of each user in realtime. Many concepts have already been tested out as prototypes for those purposes. It has been shown from these tests that the system can adaptively offer user information concerning what he is doing.

As future work, two important issues need to be addressed. First, an equally important factor to consider when inferring the required information is the emotional and psychological state of the user. This factor must be further investigated with a view to obtaining outcomes that closely reflect a higher level of reality from users. This criterion can be taken as a perceived critical aspect when it comes to intelligent applications, and it becomes more important in the healthcare arena. Second, future work will address the following research objectives: evaluation and verification method to analyse operating models of pervasive systems must be designed. Moreover, providing accurate and exact information is not always guaranteed. This problem is not well studied although it can be considered a dysfunction aspect of pervasive systems. The aim is to integrate this dysfunction analysis phase in the adaptation process.

REFERENCES

- [1] Susanna Pantsar-Syväniemi, Jarkko Kuusijärvi, Eila Ovaska. Context-Awareness Micro-architecture for Smart Spaces. In *Advances In Grid and Pervasive Computing, Lecture Notes in Computer Science*, 2011, Volume 6646/2011, 148-157.
- [2] Eirini Kaldeli, Ehsan Ullah Warriach, Jaap Bresser, Alexander Lazovik, Marco Aiello. Interoperation, Composition and Simulation of Services at Home. *Service-oriented Computing Lecture Notes in Computer Science*, 2010, Volume 6470/2010, 167-181.
- [3] Dirk Roscher, Grzegorz Lehmann, Veit Schwartze, Marco Blumendorf, Sahin Albayrak. *Dynamic Distribution and Layouting of Model-Based User Interfaces in Smart Environments*. In: Springer Berlin / Heidelberg; 2011
- [4] Michael Quade, Marco Blumendorf, Grzegorz Lehmann, Dirk Roscher and Sahin Albayrak. Evaluating User Interface Adaptations at Runtime by Simulating User Interaction. In: *25th BCS Conference on Human Computer Interaction - HCI2011*; 2011
- [5] Kranz, M., Holleis, P., Schmidt, A.: *Embedded Interaction: Interacting with the Internet of Things*. *Internet Computing, IEEE*, vol.14, no.2, pp.46-53 (2010)
- [6] Tran, M.H., Han, J., Colman, A.: *Social context: Supporting interaction awareness in ubiquitous environments*. In *Mobile and Ubiquitous Systems: Networking & Services, MobiQuitous*, pp 1- 10 (2009)
- [7] Ramón Hervás, , José Bravo. Towards the ubiquitous visualization: Adaptive user-interfaces based on the Semantic Web. In *Interacting with Computers Volume 23, Issue 1, January 2011, Pages 40–56*
- [8] A. Dey, D. Salber, and G. Abowd: "A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications", *Special issue on Context-Aware Computing in the Human-Computer Interaction (HCI) Journal*, vol. 16 (2-4), pp. 97-166, 2001
- [9] J. S. Pascoe, R. J. Loader and V. S. Sunderam. An Election Based Approach to Fault-Tolerant Group MemberShip in Collaborative Environment. In *Proc. of the 25th IEEE Anniversary Annual*

- International Computer Software and Applications Conference (COMPSAC), pages 196-201, Chicago, IL, October 2001. IEEE Press. ISBN: 0-7695-1372-7.
- [10] Gu, T., Pung, H.K., Zhang, D.Q.: A service-oriented middleware for building context-aware services. *Journal of Network and Computer Applications* (2005), Volume: 28, Issue: 1, Publisher: Elsevier, Pages: 1-18
- [11] H Chen, T Finin, A Joshi. "An Intelligent Broker Architecture for Context-Aware Systems". Adjunct Proceedings of Ubicomp, 2004
- [12] Tarek Chaari, Frederique Laforest. Adaptation in Context-Aware Pervasive Information Systems: The SECAS Project. *Journal of Pervasive Computing and Communications*, Vol. 2, No. 2, June 2006
- [13] F. Buttussi, "A user-adaptive and context-aware architecture for mobile and desktop training applications", in Proc. Mobile HCI, 2008
- [14] Mahmoud Hussein, Jun Han, and Alan Colman. An Architecture-based Approach to Context-aware Adaptive Software Systems. Technical Report #C3-516_01, Swinburne University of Technology, Australia. January 2011
- [15] Daqiang Zhanga, Minyi Guoa, Jingyu Zhoua, Dazhou Kangb, Jiannong Caoc. Context reasoning using extended evidence theory in pervasive computing environments. In *Future Generation Computer Systems*. Volume 26, Issue 2, February 2010, Pages 207–216.
- [16] Lima, J.C.D.; Rocha, C.C.; Augustin, I.; Dantas, M.A.R.; , "A Context-Aware Recommendation System to Behavioral Based Authentication in Mobile and Pervasive Environments," *Embedded and Ubiquitous Computing (EUC)*, 2011 IFIP 9th International Conference on , vol., no., pp.312-319, 24-26 Oct. 2011
- [17] Juan Ye, Simon Dobson, Susan McKeever, Situation identification techniques in pervasive computing: A review, *Pervasive and Mobile Computing*, Volume 8, Issue 1, February 2012, Pages 36-66, ISSN 1574-1192, 10.1016/j.pmcj.2011.01.004.
- [18] Rasch, Katharina; Li, Fei; Sehic, Sanjin; Ayani, Rassul; Dustdar, Schahram. Context-driven personalized service discovery in pervasive environments. In *World Wide Web*, Volume: 14, Issue: 4, 2011-07-01. Pp 295-319. Springer Netherlands, Issn: 1386-145X.
- [19] Ismail I. « Contextual Data Management in Ubiquitous Environment ». Taibah University International Conference on Computing and Information Technology. Al-Madinah Al-Munawwarah, Saudi Arabia 12-14 March 2012.
- [20] Owl2 the Web Ontology Language: Structural Specification and Functional-Style Syntax Boris Motik, Peter F. Patel-Schneider, Bijan Parsia, eds. W3C Recommendation, 27 October 2009, <http://www.w3.org/TR/2009/RECowl2-syntax-20091027>. Latest version available at <http://www.w3.org/TR/owl2-syntax/>. Last consultation March 2012.
- [21] F. Moussa, M. Riahi, C. Kolski and M. Moalla. Interpreted Petri Nets used for Human-Machine Dialogue Specification in Process Control: principles and application to the Ergo-Conceptor+ tool. *Integrated Computer-Aided Engineering*, 9, pp. 87-98, 2002.
- [22] Ismail I. and Moussa F. Towards a Runtime Evolutionary Models of User Interface Adaptation in a Ubiquitous Environment. The Fifth International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies: UBICOMM. Lisbon November 2011.
- [23] Ismail I. and Moussa F. User Requirements Deduction in a Pervasive Environment. NGMAST: IEEE International Conference on Next Generation Mobile Application, Services and Technologies. July 2010.
- [24] Schilit, Bill N., Adams, Norman I. and Want, Roy. Context-Aware Computing Applications, in Proceedings of the Workshop on Mobile Computing Systems and Applications, Santa Cruz, CA, December 1994. IEEE Computer Society.
- [25] P.J. Brown, J.D. Bovey, X. Chen. Context-Aware applications: From the Laboratory to the Marketplace. *IEEE Personal Communications*, 1997, Vol. 4, N°5, pp. 58-64.
- [26] N. Ryan, J. Pascoe, and D. Morse, "Enhanced reality fieldwork: the context-aware archaeological assistant", In Gaffney, V. et al. (Eds.) *Computer Applications in Archaeology*, 1997.
- [27] A. Schmidt, K. Aidoo, A. Takaluoma, U. Tuomela, K. Van Laerhoven, and W. Van de Velde, *Advanced Interaction in Context*, 1th International Symposium on Handheld and Ubiquitous Computing (HUC99), Lecture notes in computer science, vol. 1707, Springer, pp.89-101, 1999.