Management of User Preferences and Conflicts to Ubiquitous Applications Adaptation

Salima Bourougaa-Tria[1], Hassina Seridi-Bouchelaghem[2] and Farid Mokhati[3]
[1]*Tebessa University, LAMIS Laboratory, Tebessa, Algeria*
[2]*Annaba University, LabGED Laboratory, Annaba, Algeria*
[3] *Oum el bouaghi university, LAMIS Laboratory, Oum El Bouaghi, Algeria*

Keywords: Context of Use, User Profile, Web-based Information System, Nomadic Environment, Ontology, User Preferences, Conflicts, Ubiquitous Computing, Web Service.

Abstract: Ubiquitous computing is a recent research domain that has attracted interest from several researchers. Its main objective is allowing users to access data anytime, anywhere, in particular using Mobile Devices (MD).Applications in this domain are sensitive to the context and must be able to perceive it to adapt their behaviours to this context, taking into consideration data that deals with the context of use and user preferences. Data about context of use provides in particular a conditions description (temporal, spatial, hardware, etc.) under which the user accesses Information Systems (IS). Data about user preferences aims at expressing what the user would like to obtain from the system considering different aspects (activities display, etc.) to meet nomadic users' expectations. In this paper, we attempt to answer this problem by proposing a novel approach allowing essentially: (1) representing the semantic context and nomadic user's preferences through a novel proposed ontology, (2)resolving conflicts that may arise between user preferences and, (3) adapting such applications to the context of use and user's profile by adapting the user's request.

1 INTRODUCTION

Currently, Web users access to a large mass of invarious data situations through distinct devices, to have answers to their requests that are usually very numerous, from multiple sources of information (heterogeneous and remote). Such answers are not all equally interesting and relevant, and they do not answer all the user's wishes, which may decrease the user satisfaction. This complexity is increased if the user is nomadic (user who frequently changes localization) and appealed SIW (System Information on the Web), anywhere and anytime via mobile devices (PDAs, phones, laptops) because the change of localization, for example, causes a change in working conditions and consequently a change in the general context of use. What incited developers to integrate these mobiles devices into their applications, giving rise to new information systems called pervasive or ubiquitous.

In this case, these applications must taking into account the user's situation called contextual situation. This latter includes the context of use as well as information on its profile. To adapt all applications behaviours to a situation for returning to the user a relevant response from the content point of view and time, it is the subjacent idea of ubiquitous computing, where applications are sensitive to the context (context-aware applications).

In this context, ensuring access by nomadic users to information Systems through various devices and the adaptation of responses to nomadic users profile and context of use are two bound problems. Nevertheless, tree aspects can be approached by the following questions:

- How to model the context of use and the nomadic user profile?
- How to resolve conflicts that may arise between users' preferences?
- How to adapt the context-aware application behaviour to satisfy the needs of these mobile users?

In order to answer these questions, we propose an approach which includes:

- Ontology to modelling the context of use and the user's preferences based on a new definition of the

Bourougaa-Tria S., Seridi-Bouchelaghem H. and Mokhati F..
Management of User Preferences and Conflicts to Ubiquitous Applications Adaptation.
DOI: 10.5220/0004010802710276
In *Proceedings of the 14th International Conference on Enterprise Information Systems* (ICEIS-2012), pages 271-276
ISBN: 978-989-8565-11-2

context which separates the application data from the contextual data.

- The conflicts resolution solution.
- A Web Service based architecture to ensure the dynamic functional adaptation of Web service-based context-aware applications.

The remainder of this paper is organized as follows. In Section 2, we gave a brief overview of major related works. We present the architecture in section 3. We detail the static part of our approach in Section 4. Section 5 details the dynamic part of the system. We explain the implementation of our approach in section 6.We discuss in section 7, our actual research, draw some conclusions and give some future work directions.

2 RELATED WORKS

We distinguish four categories of context modelling approaches: The first consists in storing the context by using key-value pairs (attribute, value) or by using a set of triplets. The category of the model-oriented approaches includes: Markup Scheme Models, graphical Models and Object Oriented Models. The third category represents the context by a logic-based model. The last category models the context by using ontologies, the most referred modelling are: CoBrA-ONT by (Chen, 2003), SOUPA by (Chen, 2004), and ontologies of PUMAS a framework based on the agents proposed by (Carrillo, 2007).

In (Belhanafi, 2006), we find a synthesis on the characteristics of the context modelling approaches and this let us deduce from it that in spite of the principal disadvantage of the ontology approaches which is the ontology's complexity execution and the reasoning weight on their facts and their entities. They are most expressive and most promising for context description in an environment sensitive to the context. This is our motivation to choosing ontology in context modelling in this work.

Those works have considerably forwarded the domain by proposing novel strategies for context modelling. However, they omit some important aspects which can be summarized as follows:

- None of existing ontologies of context modelling separate between the context data and the applications data. According to (Chaari, 2006), this separation is very necessary for a reliable modelling of context.
- The user's preferences management considered only by PUMAS (Carrillo, 2007). Although, it represents a very important point to satisfy the user and to return to him answer adapted to its context.
- The conflicts' resolution is considered only by PUMAS (Carrillo, 2007). It defined some conflicts and presents their corresponding solutions. But this approach does not solve this problem, because it has not considered all conflicts which can arise during the user's preferences checking.

3 CONTEXT-AWARE MODELLING AND ADAPTING APPROACH ARCHITECTURE

The approach presented in this paper is focused on three important activities:

- The modelling of the context of use and the users' preferences based on a new definition of the context which separates the application data from the contextual data.
- The resolving of conflicts.
- The dynamic functional adaptation of web service-based context-aware applications.

Consequently, our approach consists of two parts (Figure 1):

- **A Static Part:** focuses on the contextual information modelling of users and especially the preferences and conflicts management which can arise between the user's preferences during their check.
- **A Dynamic Part:** assures the functional dynamic adaptation of these context-sensitive applications to the various user's contextual situations and aims to better respond its needs.

According to figure 1, the application adaptation to context of use and the user's profile passes by 16 steps which will be explained in the following sections.

4 THE STATIC PART: MODELING CONTEXT AND CONFLICT MANAGEMENT

In this section, we present the static part of our approach modeling the context and managing conflict, which accedes to application in the ubiquitous computing field, where applications are sensitive to the context: "context-aware applications". We opted for a semantic representation of the context of use and the

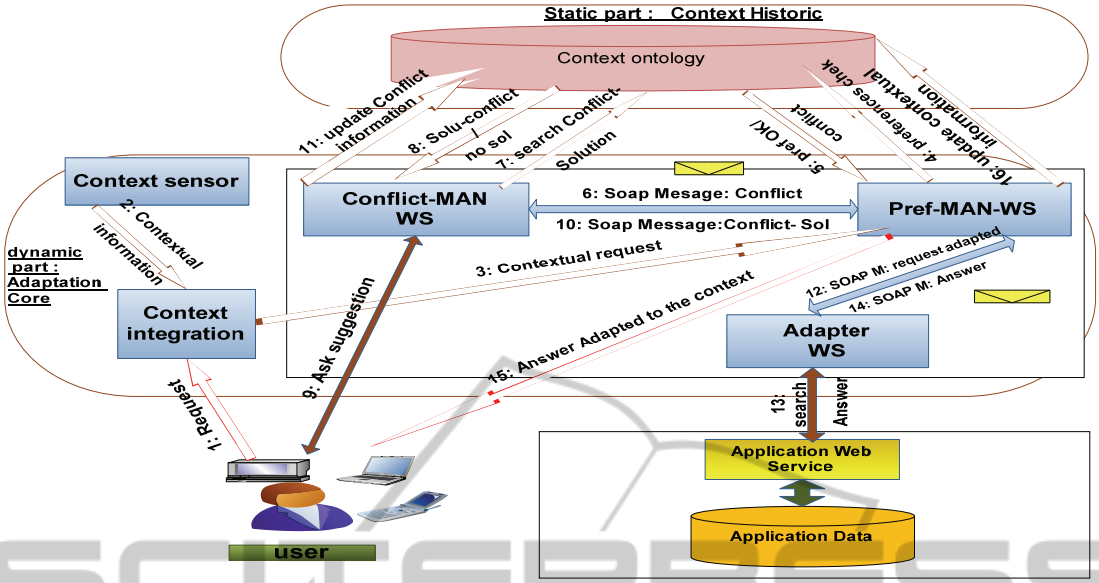

Figure 1: Architecture of our approach.

profile of user, which involves the use of ontologies. Because this representation focuses on relationship between information content, allowing inferences of other information, in order to adapt information to this context of use and user profile. We built light general context ontology to represent contextual information. For the ontology developing, we followed the ontology methodology construction (Kassel, 2002) following three steps: conceptualization, ontologization and operationalization.

The conceptualization consists of identifying in a corpus the knowledge to determinate domain (Farquhar, 2000).In the sensitivity to the context domain, the corpus contains only definitions. Researches in the context-awareness domain have not yet led to a generic and pragmatic definition of context. So, several definitions for the context were advanced (Pittarello, 2005) and (Chaari, 2006). The definitions issued so far are very abstract or very specific to a particular domain, making the formalization of the context very difficult. The (Dey, 2001) definition is widely accepted as a "good" definition. According to (Chaari, 2006), definition does not help in separating the contextual data from the application data, and the core of the application should be designed in a context independent way. This separation according to (Chaari, 2006) is very important, before beginning the design of an application sensitive to the context. A data defined as contextual in a field can be a data application in another field. Separation between the contextual data and the application data is also important for the modelling of the context. Consequently, the determination of the most descriptive concepts of information which constitutes the context, we chose the separation of the contextual data of the application data according to the definition of (Chaari, 2006) of the context, because it seems to us relevant and generic.

According to our corpus, we can divide the concepts of the context into two parts: the concepts which represent the context of use of a user and the concepts which represent the user profile.

The **"context of use"** in our approach presents the set of data which allows indicating the situation of the user when it connects to the ubiquitous application. It is represented by the following concepts: The user; the session; the used mobile device (MD) and location of the user.

The **"user profile"** is present by a set of preferences of user. We detail these concepts in the following sections.

4.1 User Preferences: Definition and Types

We define two types of preferences:

4.1.1 Activity Preferences

Describe how the user plans to perform its activities in the system. We define this type of preferences as follows: During his first contact with our system, the user can define the contents of each of his preferredactivities.I.e. The user can define from the

beginning when he asks the activity "A" what implies automatically the contents: C1, C2 ....Etc.

Activité (A) →contents (C1, C2,…….etc.)

Therefore, the preferences of activities are represented as follows:

Activity_Preference (A, {content}, {associated_ Activities}).

{Content}: is a list of the contents defined by the user from his first contact with our system .*A*: is the Activity which the user wishes to carry out in the system. *{Associated_ Activities}* is a list of the associated activities which the user wants to execute if he asks the activity *A*.

4.1.2 Display Preferences

Display Preferences describe how the user wants the information to be displayed on his MD (for example, the user only wants information in text format). At every activity is associated a Display preference. It is represented as follows:

Display _Preference (format, characteristics)

Format which can take the value: "video" ,"text", "image", "sound". Each format is based on a set of characteristics.

The user profile is represented by the Activities preferences and Display preferences.

4.2 Conflict: Definition and Types

By conflict we refer to problems which can arise during the verification of user preferences. For example, if a user demands activities which are not suitable for access rights, he can receive nothing. For these problems (conflicts), we offer some suggestions to solve them. At every type of conflict is associated a solution. It is represented as follows:

Conflict (Type, Solution, Suggestion)

- *Type*; represent the conflict which can arise.
- *Solution;* allows defining how to take action to resolve the conflict that occurred.
- *Suggestion*: represents the proposal of the user in cases where the system cannot find a solution to the conflict that occurred.

Our approach manages five conflicts which can be arising between the user preferences during the check of these last ones.

Figure 2 illustrates the context ontology called 'ContoLogy' that we proposed using UML class diagram.

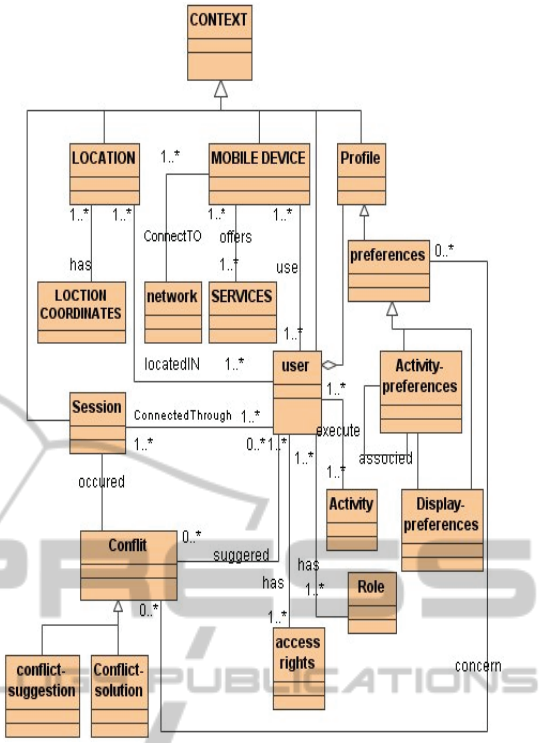

Figure 2: UML class diagram of the ontology of profile "ContoLogy".

5 THE DYNAMIC PART: ADAPTING UBIQUITOUS APPLICATIONS

In this section, we present the dynamic part of our approach for adapting the ubiquitous applications to the user's context and the user profile. This part assure the functional dynamic adaptation of these applications sensitive to the context of use and the profile of user, in the various contextual situations of a user has at the end to meet the expectations better of this last one.

This adaptation is assured by the adaptation of the user request to the context of use and the user profile. The context of use of a user witch accedes to a ubiquitous application, in addition to being composed of multiples aspects, is very variable and in constant evolution, which returns the operation of adaptation of the application in question a difficult task. In order to ensure this process of adaptation and to be able to change the behaviour of such application sensitive to the context of use, we propose:

- Using Web Services (Ponge, 2004) during the development of this type of application.

- Using in the dynamic part of our approach, a module composed of three web services, and two modules for detecting the context: Context integration and context sensor.

The dynamic part is composed of three Web Services as follows: Preferences Manager WService, Conflicts manager WService and WService Adapter, Context integration and context sensor. These modules assure:

- The adaptation of the user request to the context;
- The resolution of conflicts;
- Returning an answer adapted to the context to the user.

5.1 Preferences Manager WService

This Wservice is charged of the management of the preferences. It checks conformity between the access rights of the user, the characteristics of his used MD and the specified preferences.

5.2 Conflicts Manager WService

This WService has as role to manage the conflicts that may arise between user preferences. The conflicts are managed by our approach according to the following sequence diagram (See figure 3). It ensures:

- The resolution of conflicts;
- Sends the solution of conflicts;
- The storage of information on the conflicts which have occurred.

5.3 Adapter WService

It represents the principal WService of our architecture. It executes the flowing steps to obtain the answer to the request of user:

- Access to the Web Services of the application.
- Research on the WSDL of these latter, in order to extract WServices with their interfaces, their operations and the number of interfaces specific to each Web Service
- Select the WS which answer better the request of the user.
- Reformulate and sends the adapted answer to the context of use.

5.4 Context Sensor

This module is responsible for capture the context of the user at the time of a connection, namely: localization, MD, session. Then sends these contextual information to the module "Context integration". It is composed of two S/modules: 1) *Logical context sensor*: a set of interfaces used by the user to enter his context. 2) *The physical context sensor*: a set of physical dispositive used to capture the context of the use.

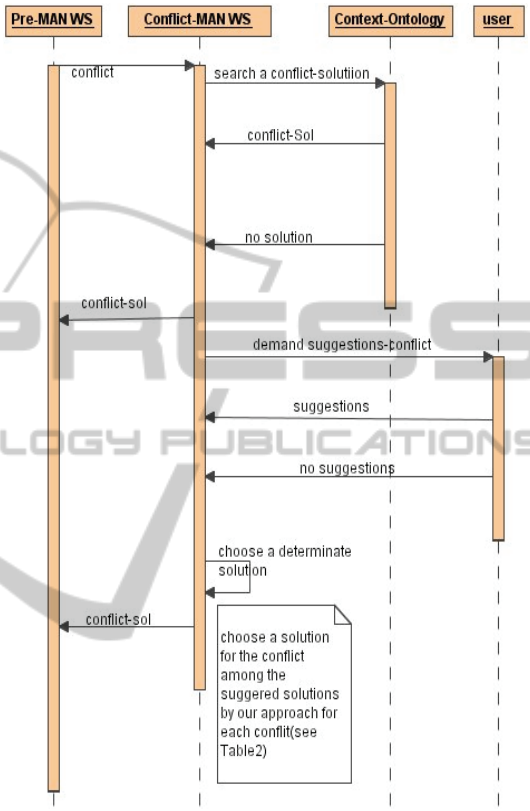

Figure 3: conflict sequence diagram.

5.5 Context Integration

This module ensures:

- Receive the initial request of user.
- Reformulates the initial request of the user by adding the contextual information
- Sends the contextual request.

6 IMPLEMENTATION

In the implementation phase, we followed the following steps: Firstly, we created the context ontology "Contology", using the editor PROTEGE2000 (Protégé, 2000) As a result of this step we obtained the OWL File that will be used by SPARQL queries. In step2, we developed a web

service-based application for managing a travel agency and Online reservation. This application will be used by our system for the adapting to the context of use and the profile of user.

7 CONCLUSIONS AND FUTURE WORKS

The ubiquitous computing focuses on the use of two essential notions: user profile and context of use, in order to satisfy better the demands of the nomadic users. Furthermore, a reliable modelling of such two notions and an adaptation of the application behaviour to them are two required processes. In this paper, we presented a novel approach that allows, on one hand, modelling the context of use and the user profiles using an ontology we proposed, and, on the other hand, resolving the conflicts using some proposed solutions. An architecture illustrating the dynamic adaptation of web service-based ubiquitous applications is also proposed.

As future directions to this work, we envisage to complete the development of our approach by using the context ontology and the developed application for the management of a travel agency and Online reservation.

REFERENCES

Belhanafi N., 2006. Ajout de mécanismes de réactivité au contexte dans les intergiciels pour composants dans le cadre d'utilisateurs nomades. Thèse présentée pour l'obtention du grade de Docteur de l'Institut National des Télécommunications Soutenue le 27 Novembre

Chaari. T and Laforest. F., 2006. Adaptation in Context-Aware Pervasive Information Systems: The SECAS Project, *Journal of Pervasive Computing and Communications*, Vol. 2, no. 2, June 2006. Received: August 2 2005; revised: January 27

Carrillo R. A., 2007.Agents ubiquitaires pour un accès adapté aux systèmes d'information : Le Framework PUMAS. Thèse pour obtenir le grade de docteur de l'université joseph fourier Spécialité : Informatique, préparée au Laboratoire l'Informatique de Grenoble présentée et soutenue publiquement le 5 mars.

Chen, H., Finin, T. and Joshi, A., 2003.Using OWL in a Pervasive Computing Broker. In *Proceedings of Workshop on Ontologies in Open Agent Systems (AAMAS)*.

Chen, H., Perich, F., Finin, T., Joshi, A., 2004. SOUPA: Standard Ontology for Ubiquitous and Pervasive Applications. *International Conference on Mobile and Ubiquitous Systems: Networking and Services*, Boston, 22-25 August

Dey A. K., Abowd. G. D., and Salber.D., 2001. A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications. *Human-computer Interaction*, 16: 97–166.

Farquhar A., 2000. Fikes R. &Rice J., Ontolingua server: a tool for collaborative ontology construction, in *International Journal of Human-Computer studies* (46), pages 707-727.

Kassel G., 2002.OntoSpec : une méthode de spécification semi-informelle d'ontologies, in Actes des journées francophones d'Ingénierie des Connaissances (IC'2002), pages 75-87.

Pittarello, F., 2005. Context-Based Management of Multimedia Documents in 3D Navigational Environments. In: Candan, K. S., Celentano, A. (eds.): *Proceedings of the 11th International Workshop on Advances in Multimedia Information Systems (MIS 2005)* (Sorrento, Italy, September 19-21, 2005), Lecture Notes in Computer Science, vol. 3665, Springer-Verlag, Berlin Heidelberg, pp. 146-162.

Ponge, J., 2004. Comptabilité et Substitution Dynamique des Web Services, Mémoire de fin d'études, Université Blaise Pascal Clermont II, juillet.

Protege 2000 Ontology Editor Home Page, http: //protege.stanford.edu