



HAL
open science

Dynamic Services Adaptation to the User's Context

Bouchra Soukkarieh, Florence Sèdes

► **To cite this version:**

Bouchra Soukkarieh, Florence Sèdes. Dynamic Services Adaptation to the User's Context. 4th International Conference on Internet and Web Applications and Services (ICIW 2009), May 2009, Venice/Mestre, Italy. pp.223-228, 10.1109/ICIW.2009.39 . hal-03765452

HAL Id: hal-03765452

<https://hal.science/hal-03765452>

Submitted on 31 Aug 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Dynamic Services Adaptation to the User's Context

Bouchra Soukkaïeh, Florence Sèdes
Paul Sabatier University, IRIT
Toulouse, France
{sokarieh, [sedes](mailto:sedes@irit.fr)}@irit.fr

Abstract— The diversity of terminals used by the user to access resources (Personal Digital Assistant, mobile phone, etc.) using several types of networks (wireless, local, etc.) generates a growing need to adapt services dynamically to the user's context. In this article, we present our architecture that aims at adapting content and presentation of services to the user's context. We realize the content adaptation based on a new data modeling methodology. This methodology aims to take into account the different structures associated with the same data allowing a data representation in several viewpoints. The presentation adaptation is based on an automatic generation process of the interfaces service code. The context in our architecture is presented by a generic model for the user and the service. The adaptation process is detailed based on an e-learning scenario.

Keywords— component; adaptation; content; presentation; context; service.

I. INTRODUCTION

Nowadays, the variety of devices used by the user to access resources (PC, Personal Digital Assistant, etc.) using several types of networks (wireless, local, etc.) create an increasing need for services that adapt dynamically to the user's context. So, to ensure the usability of the service, it is necessary to advance in the design phase of this service by finding ways to realize the adaptation to the user's context. This context should contain information on the user's environment (device, network, location, etc.) and his profile (preferences, name, etc.).

Researchers in the adaptation area consider, therefore, the context as a main parameter. However, they did not agree yet on a definition of the context concept which is generic and pragmatic. But among the different existing definitions, the definition of Dey is widely accepted: context is any information that can be used to characterize the situation of an entity. An entity is a person, a place or an object that may be relevant to the interaction between the user and the application, including user and application themselves [2].

Several works address the problem of providing the user with an adaptive service from different viewpoints: terminal, network, etc.

(i) The objective of Cheverest [10] and Brut [14] is to propose a service which returns to the user relevant information to his context where the context is presented by user location and his preferences.

(ii) CAMELEON [5], MONTAGE [13] and SOMA [18] are platforms interested in applying of mobile agents in order to personalize services. The context, in these platforms, is characterized by the terminal capacity.

(iii) VESPER (Virtual Home Environment for Service Personalization and Roaming Users) [3] and CESURE [15] projects realize the modeling and the operation of service applications to the mobile user. The context is often represented either by the network resources, or by the terminal capacity.

In spite of the importance of the context notion, we note that the majority of previous works have a limited usage of this notion, such as Cheverest that considers only the user location. Nowadays, several systems are interested only in the user location or in aspects related to the use of mobile devices as CAMELEON.

In addition, researchers in this domain work to save multiple versions of the service and then send to the user a version pertinent to his context in order to achieve the service adaptation to the context changes:

(i) [16] proposed an architecture that aims to realize dynamic services adaptation according to the user's context. In this work, each service is presented by a set of components. The context concerns only the user's needs. Profiles, which describe not only the user's context but also the context of each component containing in the service, are used by an "Adapter" to verify the compatibility between the user profile and the profile of each component of the service. When the Adapter detects the incompatibility between the profiles, it starts an adaptation process by adding, removing or replacing components of the service.

(ii) [17] presented an architecture that aims to dynamically adapt services to the user's context where each service is composed of a set of interconnected components. The services adaptation is realized at the access time of the service and at the time of his execution on the terminal. This adaptation aims to select and assemble the service components depending on the context. The adaptation of the service at the time of his execution consists of dynamically changing the service composition following the changes of the environment by adding, removing or replacing service components.

(iii) [20] proposed a generic architecture in order to provide adaptable services to mobile users. This architecture addresses simultaneously the logic service adaptation and the multimedia content adaptation. A database of multimedia

data and another database of services and/or services components are generated by the service provider. These two databases contain several versions of service data and components in order to select the relevant version of the service to user's context.

We note that precedents works aim to save multiple versions of service components and/or data in order to realize the adaptation process. This method leads to the saturation of the system. So, we propose in this article an architecture that aims to treat this problem and to propose some solutions in order to reduce the system overload.

Indeed, researches distinguish between two types of adaptation.

(i) Logic adaptation: Several researchers focus on the logic adaptation of services such as work [7][4]; in this case the service is represented by a set of components; the adaptation aims at changing assembly of these components by adding or replacing a component.

(ii) Content adaptation: Other researchers [12] and [9] concentrate on the content adaptation; it is a set of adaptability forms that chooses changes of the content provided by the service depending on the context. A typical form of the content adaptation is changing the encoding of a multimedia stream or changing the presentation of a service depending on the context.

In our work we focus on content adaptation of a service. We consider that a service is a combination of data and operations. So, the adaptation of the content of a service is the adaptation of services data to the user's context. To achieve this adaptation, we propose an architecture that performs the content adaptation based on a new methodology of data modeling. This methodology does not impose a data structure in priori but personalizes data representations according to usages and contexts changes. The context in our architecture is presented by a generic model for the user and service.

In this article we present a scenario that illustrates different adaptation situations. Then, in Section 3 we present our architecture. In Section 4, we explain the adaptation process of the content and the presentation of service introduced in our architecture by detailing the methodology of data modeling that is the basic of our research. In Section 5, we expose the context modeling in our work. In Section 6, we illustrate our architecture basing on the scenario introduced in Section 2. We conclude, in Section 7, by presenting some perspectives.

II. SCENARIO

Step1. Antony wants to improve his English level, but first he must clarify his current level. By using the Web browser of his PC, he connects to a service allowing evaluating the English level through a test of 3 exercises which include questions written by an English teacher. Antony desires that the test questions are presented in text mode.

Step2. Suddenly, Antony remembers that he has an appointment after 10 minutes with his friends in the downtown. Having a mobile phone, he can continue his testing. He turns on his phone and asks to recover the service

that he uses on his PC hoping that the test questions are presented by video mode.

The previous scenario shows the need to adapt a service to changes in the user environment and his preferences. To adapt a service, it requires the presentation adaptation and the content adaptation of this service. In the next paragraph, we present our architecture that allows the user to restore an adapted service.

III. ARCHITECTURE FOR PROVIDING AN ADAPTIVE SERVICE

In this Section, we present an architecture that manages the dynamic service adaptation to the user's context (Figure 1). This architecture is composed of five elements: the **user**, the **provider**, **Services Management (SM)**, the **Context Management (CM)** and the **Adaptation Management (AM)**.

A service is a combination of data and operations made available to users through the Internet. The user can interact with the service by their interfaces. Each service has two interfaces input and output that contain their data. This service is available on the Web by a **provider**. In order to adapt this service dynamically to the user's context, we adapt their data based on a new methodology of data modeling proposed in our team. This methodology is based on the consideration of various structures associated with data which allow data representation according to several viewpoints. So, no structure is imposed in priori to data but a personalization of representation to various usages and context is induced (cf. Section IV.A).

The service provider must then store specific and generic structures of his service data "**Data and SGS**". It must also provide the context of his service and store it in a database "**Context of the Service (CS)**" (cf. Section V).

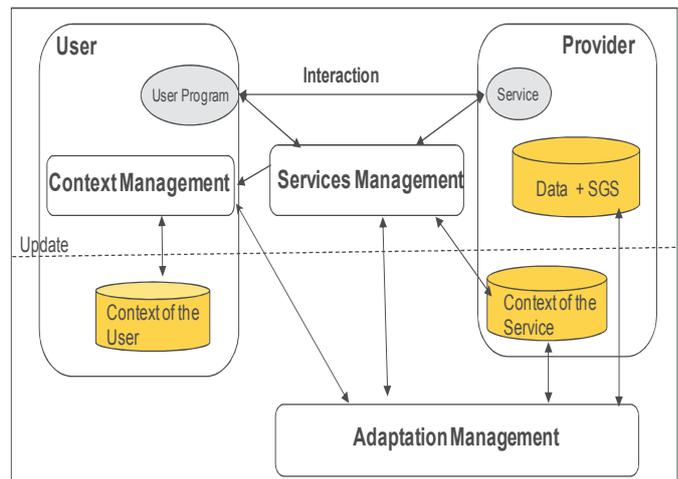


Figure 1. Architecture for managing an adaptive service to the user's context

When the **user** connects to a service, the **Context Management** component captures the user's context and stores it in a database "**Context of the User (CU)**". The user's context is retrieved by explicit and implicit ways (cf.

Section V). The user may at any time change his preferences and context. Thus, CU is updated by the *Context Management component*.

The interaction between the user and the service is not direct but is realized by *Services Management component*. When the user starts interacting with the service, *Services Management component* verifies the compatibility between the user's context and the service context. If the two contexts are not compatible, Services Management component triggers an adaptation process by the *Adaptation Management component*. This later retrieves the context of user and service in order to trigger a loop of the service adaptation.

The service is updated by the Adaptation Management component after each adaptation process. Services Management component download the adapted service to the current context of user.

IV. ADAPTATION MANAGEMENT (AM)

As we have mentioned, the Adaptation Management component is responsible for managing the adaptation process of services to the user's context. It adapts the service content and the presentation of them interfaces to the user's context.

A. Content adaptation

The content adaptation aims to change the properties of the data presented to the concerned user. It can be illustrated by changing the data type to the terminal capacities, to the network capacities or to user preferences. For example, transforming a text into a speech synthesis.

The service content adaptation; i.e. adapt the data service to the terminal capabilities, to the network capabilities and/or to the user preferences.

Indeed, in our team, [8] worked to find solutions in order to rebuild data dynamically replying to the user's needs. To achieve this goal, they created a data warehouse. This warehouse organizes and classifies data and enriched it by the metadata that is considered like any other data that we can display and use in order to adapt the data dynamically to the users needs.

The proposal of [8] replays to users needs using a desktop computer. But since the users are becoming mobile, we note that the proposal of [8] is insufficient to restore the user with the response the most relevant to his context that concern the type of device and network etc.

Other work in our team [11] based on the work of [8] focuses on storage, handling and usage of data. In this work, the author proposed a new methodology which aims to take into account the different structures associated with the same data allowing a data representation in several viewpoints.

To ensure the generation of adapted service content, we focus on the methodology proposed in previous work. Indeed, the primary interest of this methodology is to not impose a data structure in prior, but personalizes data representations to various usages and contexts. The second interest concerns the system; various structures of the same data are saved but data are saved only once, then this methodology diminishes the system overload.

In this methodology the author proposes two structures: generic structure and specific structure.

The specific structure determines the content elements of specific views. The specific view notion is presented as a set of nodes where every node can be an element, an attribute or a metadata. For example, a student file can have "first-name", "last-name" and "address" elements. The links between these nodes are established by relationships. Each relationship is typed: composition, synchronization, etc.

Each specific structure is attached to a generic structure. This latter materializes a global structure that covers a set of sub-structures. These sub-structures are represented by generic views. So, each generic view designates a class of specific views (for more detail sees Figure 2).

So in our work, the service provider must store not only the contents of his service, but also specific and generic structures of this content in a database (data + SGS). When the user begins the interaction with the service, the Services Management component verifies the change of context. In order to adapt the content of the service to these changes, the Adaptation Management component exploits (SGS) the generic and specific structure that is compatible with the current context. It generates the content of these structures, and then starts the presentation adaptation.

B. Presentation adaptation

The presentation adaptation follows directly the content adaptation in order to prepare adapted service interfaces that will present to the user adapted data.

The presentation adaptation, in our case, applies to the service interfaces: i.e. transformations lie at the display in its interfaces. These transformations follow the terminal characteristics used by the user, his network and his preferences.

In order to realize this adaptation, the Adaptation Management component provides with an automatic generation process to the service interfaces code (input/output) for multi-user and multi-terminal environment.

This component (AM) retrieves the user's context in order to specify the characteristics that concern the presentation. The AM retrieves also the content of the service after the content adaptation. Based on the automatic generation process of the service interfaces code, it generates new interfaces of the service. Finally, the AM sends the adaptation result to the Service Management which presents to the user the adapted service.

V. CONTEXT MODELING

In this field, researchers did not agree yet on a definition of the context concept which is generic and pragmatic. Indeed, proposed definitions are very specific to a particular domain. These definitions make the formalization of context difficult due to a lack of precision or of generality.

A generalized notion of context has been proposed by Dey. But Chaari [19] considers that this definition may be a source of conflict because it does not help in separating the contextual data from the application data. To provide more precision in Dey's definition, Chaari proposed the following

definition “the set of the external parameters that can influence the behavior of the application by defining new views on its data and its available services. These parameters may be dynamic and may change during the execution”.

We think that this definition is more useful for the context exploitation in applications. Therefore, we will use this definition to define the structure of our context model.

Figure 2 presents the general structure of our context model. For each user session we associate a profile "Context-Profile" which describes the four elements of the context "Context-Element". Each "Context-Element" may contain a "Context-Sub-Element". Each "Context-Sub-Element" may contain another "Context-Sub-Element" or a "Context-Attribute". The contextual situation is defined by the values associated with Context-Sub-Element of the context. The modification of one of these values corresponds to a transition to another contextual situation. Each Sub-Element of the context can be static if it does not change in a user session or dynamic if its value can vary in the same user session. That's why we associate for each Sub-element, an attribute "Type" to differentiate dynamic elements from static ones. This differentiation is very important for the adaptation process.

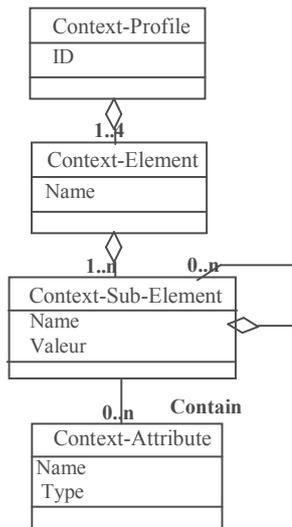


Figure 2. General structure of the context

In order to return to the user the list of the most services relevant to his context, we must take into account not only the user's context but also the services context. In our architecture, we define the service context by the four elements identified before to the user.

To represent the contextual information, the majority of researchers used the format CC/PP (Composite Capability/Preference Profiles) [6]. CC/PP is a decomposable, uniform and extensible standard but we did not use it to represent our context because of its lack of structured functionalities where its two leveled strict hierarchy is not appropriate to capture structures of complex profiles.

We use the CSCP (Comprehensive Structured Context Profiles) format to overcome the structure deficits of CC/PP.

CSCP provide a multileveled structure which allows the representation of all the types of contextual information [1].

So, we stock all the contextual information in an XML document based on this CSCP model. Each element of this document presents a context element.

Based on the general structure of our context model presented in Figure 2, we present the user's context by four elements and the service context by three. The main difference between these two contexts (user, services) is there profile.

The user's context is composed of four elements:

1. **User characteristics** which are composed of static characteristics (langue, first name, etc) and of evolutive characteristics defined by his environment (his localization, time, etc) and his preferences,
2. **Device characteristics** which are presented by hardware characteristics (device type, screen size, etc.) and software characteristics (operating system, version, etc.).
3. **Network characteristics** which expose information about type network, bandwidth, etc.
4. **Session characteristics** which present the user connection to the system (duration, date, etc).

The service context is composed of three elements:

1. **Service characteristics** which are presented by the service name, description (location, time, language, etc.), Pre-Condition (free memory, etc.) and QoS (bandwidth, response time, etc.),
2. **Device characteristics** (hardware and software characteristics)
3. **Network characteristics** (type, bandwidth, etc.)

VI. ILLUSTRATION OF OUR ARCHITECTURE BY THE SCENARIO

Going back to the first of our scenario (Section II), supposing that the service input interface contains three exercises; each one includes questions presented in video mode. We note that Antony desires that the questions are in text mode. In this case, Services Management component (SM) discovers the incompatibility between the service context and the user's context by consulting the two contexts. The SM triggers the adaptation process by connecting to the Adaptation Management component which will searches in the database (data + SGS) for the generic and specific structure which is compatible with the Antony context.

In our example Antony is a student. The AM must select among the generic views stored in the database, the view corresponding to the representative of the Antony context. So we must choose the student view (Figure 3).

Then, AM selects the specific view that matches the generic view selected (Figure 4) taking into account the user's context and his preferences. Antony wants questions are in text mode. So, AM selects the specific view (a) presented in Figure 4. After selecting the specific view of the data, the AM generates the content which corresponds with this specific view. The adapted content is sent to Services Management by AM.

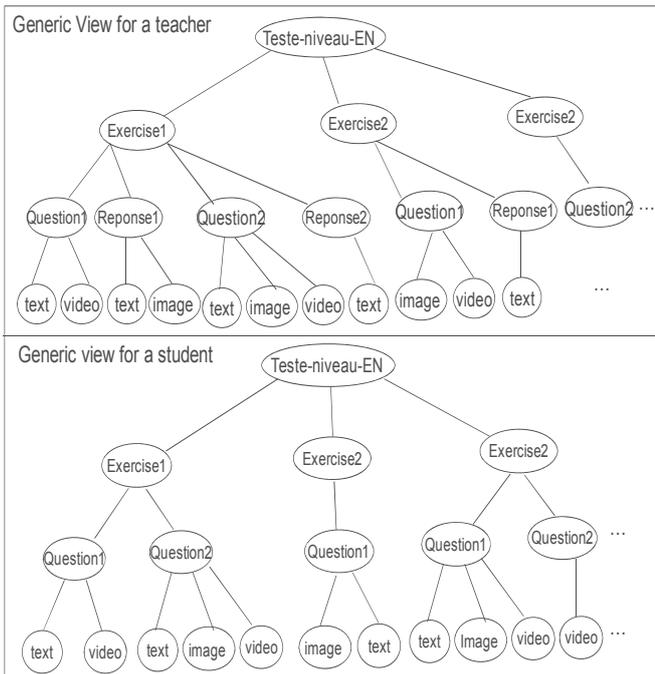


Figure3. Generic structure

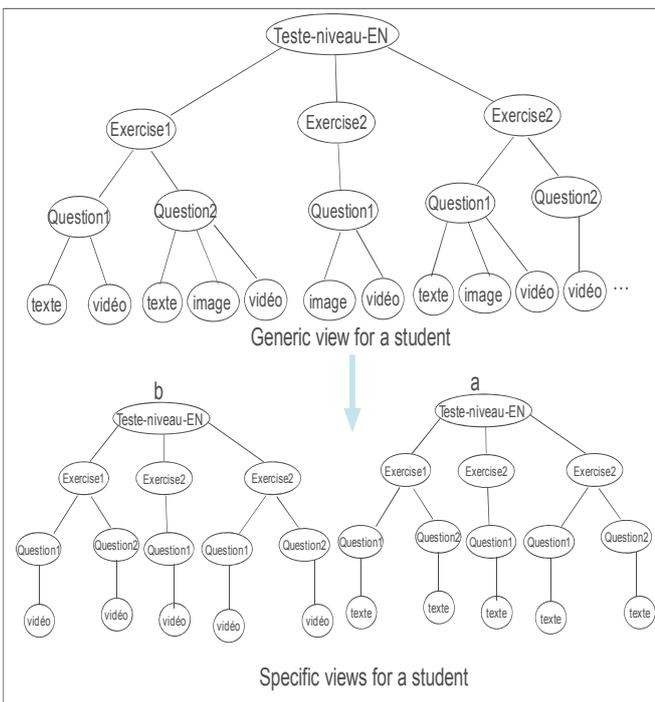


Figure 4. Generic and specific structures of the student

Taking the second step of our scenario, Antony Connected from his mobile phone requests to recover the service that he uses on his PC. This situation requires the recovery of the service execution context. Since the mobile phone has different characteristics from those of a PC, a loop of adaptation processes is triggered by the Adaptation

Management component. This latter recovers from the Context Management component the new user's context that contains the profile of new terminal. The Adaptation Management component determines adaptation actions in order to achieve the presentation and the content adaptation of the service.

Concerning the content adaptation, as we have mentioned, it selects the generic and specific structure from the database SGS and generates the content. In this example Antony would need to view the questions in video mode. The Adaptation Management component then chooses the structure (b) Figure 4. The change of the terminal characteristics aims to change the characteristics of the service interfaces to be compatible with the phone. After adapting the content, AM passes to the presentation adaptation, it generates automatically the service graphic interfaces for the mobile phone environment.

VII. CONCLUSION

In this article we proposed a general architecture that allows adapting automatically a service to changes in the user's context. This architecture aims at adapting the content and the presentation of service to the user's context.

To achieve the content adaptation, we focus on a new methodology proposed in our team. This methodology does not impose a data structure in priori but personalizes data representations according to usages and contexts changes. This methodology also diminishes the system overload because it saves various structures of the same data but data are saved only once.

To realize the presentation adaptation, we proposed an automatic generation process of the service interfaces code.

The context, in this architecture, is presented by a generic model regarding the user and the service. The service context is provided directly by the provider. The user's context is retrieved by explicitly (as user preferences) and implicit (such as device type, network type, etc.) ways.

In the near future, we envisage implanting our architecture. Then, we envisage also improving the process of automatic generation of service interfaces code.

REFERENCES

- [1] A. Held, S. Bouchholz, A. Shill, "Modeling of Context for Pervasive Computing Application", In Proceedings of the 16th world Multiconference on Systemics, Cybernetics and Informatics (SCI). Orlando, FL, USA, 2002.
- [2] A. K. Dey, D. Salber, G.D. Abowd, "A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications", In Human-Computer Interaction Journal 16 (2-4), 2001, pp. 97-166.
- [3] A. Liotta, A. Yew, C. Bohoris, G. Pavlou, "Delivering Service Adaptation with 3G Technology", In 13th IFIP/IEEE International Workshop on Distributed Systems: Operations and Management (DSOM'2002), Montreal, Canada, October 2002, Springer, pp. 108-120.
- [4] B. Marquet, C. Gustave, A. Lefebvre, S. Nemchenko, S. Chassande-Barrioz, "Secured services in a multi-tier architecture", In World Telecommunications Congress, WTC 2002, Paris, September 2002.
- [5] CAMELEON : ACTS CAMELEON project home page. <http://www.comnets.rwthachen.de/cameleon/>.

- [6] CC/PP, W3C, Composite Capability/Preference Profiles (CC/PP): Structure and Vocabularies 1.0. W3C Recommendation 15 January 2004. The last version: <http://www.w3.org/TR/CCPP-struct-vocab/> (Janvier 2007).
- [7] D M. BRAIN, "Concepts for Service adaptation, scalability and QoS handling on mobility enabled networks", IST-1999-10050 project, deliverable 1.2., 31 Mars 2001.
- [8] I. Amous, "Méthodologies de conception d'applications hypermédia - Extension pour la réingénierie des sites Web.", PHD, Paul Sabatier University, December 2002.
- [9] J. G. Kim, S. F. Wang Y and Chang, "Content-Adaptive Utility Based Video Adaptation", In ICME2003.
- [10] K. Cheverest, K. Mitchell, N. Davies, "The role of adaptive hypermedia in a context-aware tourist guide", in Communication of ACM, vol. 45, n° 5, Mars 2002, ACM Press, pp. 47-51.
- [11] K. Djemal, C. Soulé-Dupuy, N. Vallés-Parlangeau, "Formal Modeling of Multistructured Documents", In International Conference on Research Challenge in Information Science (RCIS 2008), Marrakech, Morocco, June 2008, pp. 241-249.
- [12] L. Boszormenyi, H. Hellwagner, H. Kosch, M. Libsle, S. Podlipnig, "Metadata driven adaptation in the ADMITS project", In EURASIP Signal Processing: Image Communication Journal, Vol. 18, No. 8, September 2003, pp. 749-766
- [13] MONTAGE: ACTS MONTAGE project home page. <http://montage.ccrle.nec.de>.
- [14] M. Brut, D. Al Kukhun, F. Sèdes, "Towards Service-Oriented Semantic Annotation and Retrieval within Pervasive E-Learning Systems", In International Journal of Business Intelligence and Data Mining, Inderscience Publishers, November 2008.
- [15] M. C. Pellegrini, O. Potonniée, R. Marvie, S. Jean, M. Riveill, "Cesure, une plate-forme d'applications adaptables et sécurisées pour usagers mobiles", In parallel computers, Evolutions of Platform oriented distributed object, 2000, 12(1) : pp. 113-120.
- [16] M. Cremene, M. Riveill, C. Martel, C. Loghin and C. Miron, "Adaptation dynamique de services", In DECOR'04, Grenoble, France, 2004.
- [17] O. Fouial, "Découverte et fourniture de services adaptatifs dans les environnements mobiles", PHD, Paris, 2004.
- [18] P. Bellavista, A. Corradi, C. Stefanelli, "The ubiquitous provisioning of internet services to portable devices", In Pervasive Computing, IEEE, 1(3): 81-87, July-September 2002.
- [19] T. Chaari, "Adaptation d'applications pervasives dans des environnements multi-contextes", PHD, Lyon, France, 2007.
- [20] Z. Kazi-Aoul, I. Demeure and Moissiniac J. C., "Une architecture générique pour la fourniture de services multimédia adaptables-illustration par un scénario", In Ubimob'04, June 2004.