A Context-Oriented Approach for the Adaptation of Service Compositions

Karima Belgharbi
Laboratory LIRE, University Constantine 2
Constantine, 25000, Algeria
Belgharbi_k@yahoo.fr

Mahmoud Boufaida
Laboratory LIRE, University Constantine 2
Constantine, 25000, Algeria
mboufaida@umc.edu.dz

ABSTRACT
The environments of pervasive computing are open and dynamic. In order to ensure the adaptation of the service composition and the dynamic handling of the context in such environments, we suggest a new approach based on the taking into account information of context to adapt the process of composing services. Although there exist many composition norms that have been suggested, such as BPML and BPEL4WS, these norms are somewhat limited in terms of the dynamic adaptation of the service composition during the execution. In this paper, we present an approach that takes into account at the same time the user's context and the services, with integrating them in the service description language WSDL and in the composition process BPEL. The suggested extensions, known as A-WSDL and A-BPEL, permit us to describe the adaptations capable of bearing the service, the composition of services and the dynamic handling of context.

Keywords
Pervasive computing**,** Adaptive service composition, BPEL, WSDL, Context.

1. INTRODUCTION
Pervasive systems examined by Weiser [25] aim to create a transparent and interoperable environment to ensure the sharing of information and services. In order to overcome pervasive information challenges, the service-oriented aspect of the architectures, which constitute a current trend in the integration of heterogeneous services. These architectures permit the interoperability of service conceived of independently, and the management of open systems, in which they may appear or disappear [3], respond to the opening and the dynamicity of the pervasive environments.

Current research is based on the discovery and the composition of services by using standards such as the Web Service Description Language (WSDL) [4], BPML [1] and BPEL4WS [2], and so on. However, in a pervasive environment due to the diversity of users and the conditions under which they access to services. Other parameters must be considered when making the discovery, such as the type of the device used (PDA, laptop, and so on), user's preferences, users' localization, and so on. All these parameters form a particular context of use [5]. In such a context, the process of making up services satisfying the users' needs must be able to adapt in a dynamic way to context changes. It is within this framework that the present work is undertaken.

Our objective is to suggest a solution for the composition of services in the pervasive environment in order to ensure the flexibility to dynamically generate adaptive systems, permitting the access, the sharing and the exchange of resources from the context information.

To overcome the limitations imposed by technology and the norms that do not tackle the questions related to description of service adaptation aspects, this article aims to provide a conceptual approach for describing and realizing the context adaptation of the service composition processes. In this paper, we suggest some extensions at the level of the service description language WSDL and the business process BPEL on the basis of a context model. These extensions allow us to describe the adaptations capable of sustaining services and their compositions. The suggested extensions, known as A-WSDL (adaptive WSDL) and A-BPEL (Adaptive BPEL). Finally, we validate our approach by implementing a home follow up scenario of a patient.

The rest of this article is structured as follows. Section 2 is devoted to the literature review. Section 3 is devoted to the description of our suggested approach concerning the examination of their context in order to dynamically compose elementary services, in an open and context sensitive environment. In section 4, we illustrate our approach via an example. Section 5 deals with the implementation and evaluation of the suggested approach. Section 6 concludes and presents some perspectives for future research.

2. RELATED WORK
The necessity of adapting a services composition process has been a topic of interest in recent years. Many authors have already confirmed the importance of the adaptability flexibility

ICCASA 2014, October 15-16, Dubai, United Arab Emirates

DOI 10.4108/icst.iccasa.2014.257319

for the service composition. Several research works highlight the introduction of the context in the discovery mechanism and service composition, in the domain of pervasive computing ([6], [7]). [8] has integrated this aspect in the planning mechanism in an intelligent environment, and have indicated that the means to reach a given aim may depend on the context. [9] suggest a system in which the composition is influenced by the context. [10] present an approach for the web service composition based on the use of agents and the context.

Other approaches deal with different aspects of the context adaptation on the life cycle of web services. [11] defined three techniques of adaptation to web services context: content adaptation, adaptation of presentation and functional adaptation of the service. [12] present a generic framework to sustain the development of adaptative and context sensitive web services.

However, contrary to our approach, none of these initiatives have tried to integrate the context in the process of service description and composition, to make face the limits of norms and service-oriented information technologies, which do not permit the description of users' preferences and do not make them to take into account the context, and that services participate to the composition. However, there are many suggestions aiming to extend WSDL and BPEL to other domains, such as extensions for the security [13], the quality of the service [14], the BPEL extensions placed for the sub-processes [15], BPEL extension to address human interactions (BPEL4People) [16] and BPEL extensions for the versions [17].

Another work related to our suggestion is [18] that proposes to add extensions at the level of WSDL and BPEL languages on the basis of the business event aspect. This suggestion is based on the principle of combining the service oriented architecture (SOA) and the event driven architecture (EDA).

Even these studies are different from each other, they defend the evolution of systems towards context-adaptative systems, either by integrating the context in the WSDL service description, or by inserting the context in the BPEL process. We think that the approaches are actually complementary, and that such an adaptation cannot be truly reached unless there is a combination between these two approaches. In our view, only an adaptation mechanism for service composition may be based at the same time on the integration of the adaptation at the level of service description invoked in the composition process using WSDL extensions, and on the integration of the adaptation in the composition process using BPEL extensions.

3. AN APPROACH FOR A FLEXIBLE COMPOSITION OF SERVICES IN A PERVASIVE ENVIRONMENT

In this work, we are mainly concerned with the adaptation of services composition to context, with regards to the dynamicity of service availability and context information in a pervasive environment. To reach this aim, we suggest to extend the classical service-oriented architecture (SOA) with the notion of context. Our contribution consists of adding different adaptation-devoted components to this architecture. Figure 1 presents the principles of our architecture, which are described in a preceding work, where we have demonstrated the importance of using contextual information and its benefit to the service composition, as well as the importance of taking into account the matching between service candidates during the composition process [19]. In this paper, we present our case study and the experiment that validates the suggested system.

We start by providing the definition and the context modeling approach that would be taken into account during the composition process and the selection of services as a response to a user's request. To solve this adaptation problem of the service composition, we particularly use the concept of context. Our approach consists of presenting a context model expressing the adaptation criteria in the service description. This model is meant to be integrated in the service description by using WSDL and in the specification of the service composition. In our case, we use the BPEL to describe this composition.

The approach followed in our process of adapting the service composition includes two parts.

1. The first part treats the WSDL language, which describes different services. This part is devoted to the description of different types of data necessary to the definition of the role of services involved in the composition process. Thus, this description permits to handle heterogeneous context information sources by making a service act as a producer and/ or a context consumer.

2. The second part concerns the description of the process of orchestrating different activities, which appear in a BPEL description.

These languages are based on the elements of context to describe the service composition. These elements are essential for the description of the behavior of the service composition within the framework of service composition adaptation.

To reach our approach, we describe the way we model the pervasive environment which is characterized by dynamicity and heterogeneity (of the physical environment), context-aware and users' mobility. [23] describe the components of the pervasive information systems (PIS) as a services space. In our work, we also consider the pervasive environment as a services space inside which the users will evolve. In this space, the services execute a set of behavior and can interact with other services.

Definition 1 (Pervasive Environment): A pervasive environment *Ep* consists of a set of services S characterized by a context *Cx*, a set of relations *R* with other services, and it performs the behavior *B*.

$Ep = \langle S, Cx, B, R\rangle$ in which:

-S: Is a set of involved services;

-*Cx*: The context for the proper functioning of the service;

-*B*: The play of behavior performed by the services;

- *R:* The links between the services (subscription relations).

In what follows, we briefly describe the first steps by presenting first a definition of factors which affect the composition of services (i.e., its context).

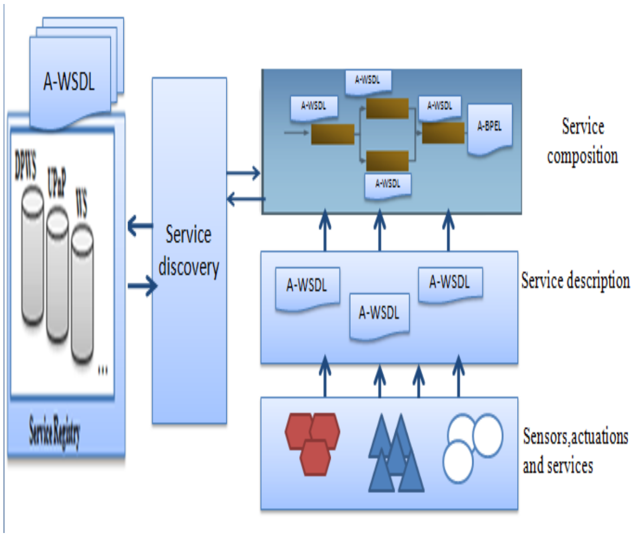

Figure 1. Adaptation of the service architecture.

3.1 Context Modeling

The notion of context is largely used in different domains of informatics, but today there is no agreement on a unique definition. The context, according to Dey and Abowd, is any information which may be used to characterize the situation of an entity, where an entity is a person, a place or an object which is considered as relevant for the interaction between a user and an interface [20]. In the literature, the context is defined according to the objective of the suggested approach. Within the framework of our approach, the objective pursued is the use of the context for the discovery and the composition of context sensitive services.

In the pervasive environment, the context is ensured from different resources, such as the sensors which can produce capture errors. In fact, it is necessary to have an abstract description of this information. This makes the context modeling an important stage in the service composition process. Our context model corresponds to an ontology to define the different elements which influence the selection of services involved in the composition process. Figure 2 represents a meta-model of contexts inspired from works carried out in this domain, and that we are trying to improve in the future.

In the present work, the context is presented by three types of information: the user's context (U-Context), the service context ((S-Context) and the environment context (E-Context).

1) U-Context: this type of context contains information concerning the user. It is made up of two parts:

- The user's static context (his profile, interests, preferences).
- The user's dynamic context (his localization, used device, role, and so on).

2) S-Context: this context gathers information related to the use, the behavior and the execution of a given service such as the features offered by the service, type of devices supporting the execution of this service, characteristics of the environment where to apply this service, and so on.

3) E-Context: this context gathers information related to the user's environment and the service (the date, the localization, and so on).

In our work, we use the notion of the contextual situation which can be defined by the values associated to elements of the context. The passage from one situation to another may change the behavior of the composition process, by selecting new services related to a given situation.

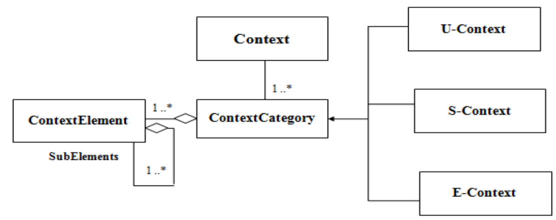

Figure 2. Meta model of context.

3.2 Integration of the Adaptation in the WSDL Standard

Our work enables service-oriented architectures to withstand the process of searching and making available services adapted to the user's context in order that a given service is able to respond to the user's context. Existing studies in the domain of context-aware present a common view on the separation of context acquisition from context use. In fact, according to [11] it is very important to encapsulate context acquisition mechanisms in components that offer a standard communication interface. Today, there is no standard context acquisition model from a variety of sources. Each context-aware system suggests its own acquisition and context collection model. In order to handle the heterogeneity of the context information sources, a context manager is necessary in the pervasive environment which bears the responsibility of collecting, classifying and stocking contextual data, and so on. In order to make the access to contexts resources more flexible, it is necessary to structure the context managers and organize them in a way that makes each service possess its proper manager. We have suggested that the context sources such as the sensors be encapsulated by the services. Thus, each service can manage information related to its context. To be able to ensure the access and sharing of this information, the integration of the context in the description of the service presents many advantages in order to permits a given service to produce and / or consume the context. To achieve this solution, we have been inspired by the approach [18] who has suggested a solution based on the principle of combining between the service oriented architecture (SOA) and the event driven architecture (EDA). They suggest specific extensions to WSDL and BPEL. In order to permits to services to act as producers or consumers of events, which is equivalent to our approach, a service may, consequently, be a context source or a context consumer.

The approach [18] does not treat the adaptation and the management of context at the level of service description and thus objective interpretation is different. Contrary to this approach which makes use of the business event, defines the event as " a significant change of state'' [26]. For example, the purchase of a car changes the state of “to sale” to the state “sold”, and the event purchase triggers the invoicing activity. However, in our work consider the event as any information which may be used to characterize the user’s situation, a service, a device and the environment.

In front of the change in values of context parameters, we have also set the triggering of the context which presents this change to contextual situations. Meanwhile, the service must also preserve its capacity to expose operations. Consequently, we have decided to add new constructions to WSDL.

The adaptation, which represents the reaction to context change, is based on the related context data collection and evaluation. The evaluation mechanisms must contain interfaces for the activation. To declare the context, in the same way as [18] we have introduced the concept of <awsdl: context>. The context declaration is <interface> in the document WSDL. For the <awsdl: context>, we have to specify the name of the context.

To activate the support for a service to act as a context source or a context consumer, we have added the expression <awsdl: ContextSource> and <awsdl: ContextReceiver> constructed at the service interface (portType). The <awsdl: ContextSource> declares that a given service is a context producer. The <awsdl: ContextReciever> declares that a given service is context consumer.

The adaptation is carried out in a given situation thanks to the phase pertaining to the evaluation of contextual information. This evaluation is based on the filtering mechanism to detect the appearance or disappearance of services. To filter the entering context, we have included the concept <awsdl: filter> by using the expressing language, XPath. To solve the adaptation problem, it is logical to trigger a situation during the reception of a service. To trigger a contextual situation, we have added a <awsdl: SituationTrigger>.

Before directly interacting with the context information producing services, the consumers asks, first of all, a directory service which we suppose to be present in the pervasive environment (see Figure 3). This directory service must characterize the producers of context information in a general way. Searching specifies the context information desired by the consumers. This is carried out as a second step thanks to a given protocol.

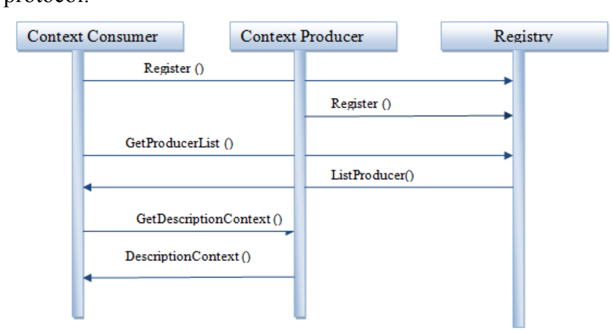

Figure 3. Example of interactions between context consumer, context producer and directory service.

3.3 Adaptation to Change of Context in a BPEL Process

In order to solve the problem of the adaptation to the service context raised in a composition process, we have relied on norms of specification of the composition process to suggest a conceptual model permitting the enrichment of the composition standard by the notion of context. In our case, we have been using BPEL. We have added extensions related to the dynamic handling of the context adaptation. The suggested extension is known as A-BPEL (Adaptive BPEL).

BPEL is an executable language to specify the business processes as service orchestrations. BPEL supports the origin of orchestrations based on the invocation of operations. We suggest extensions which would add support for the context-based service orchestrations.

To achieve the adaptation of services to the context during the execution of the composition process, for each time there is a contextual situation. The composition engine verifies the triggered situation and produces services adaptive to this situation by setting off adaptation rules to produce a series of adaptive services which might have been to the execution engine. Figure 4 presents the adaptation principle of the service composition process. Our adaptation policy rests on the choice of one or many services adaptive to the context. This choice is made with the help of a decision taken at the moment of the execution. This decision is based on the correspondence between the user’s context and the service context. An example of a composition process based on the context is presented in figure 5.

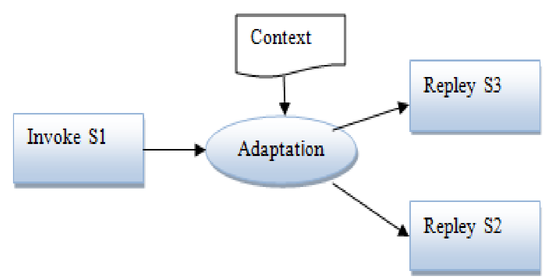

Figure 4. The adaptation principle based on the context.

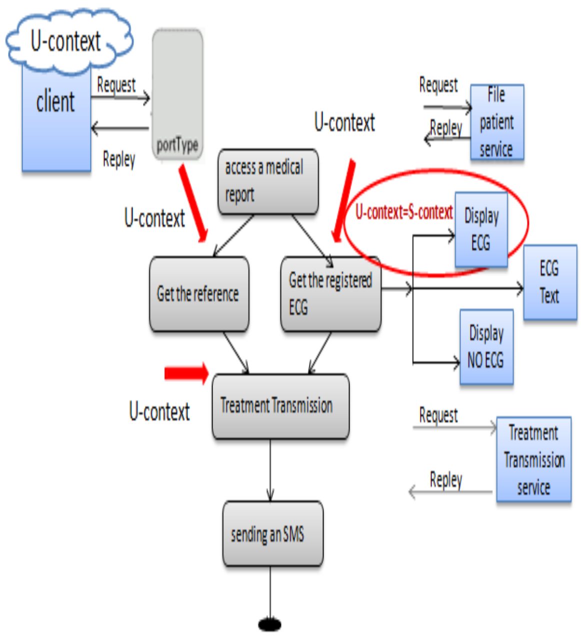

Figure 5. Context-based composition process.

To activate the orchestrations of context-based services, we have implemented new BPEL activities We have used the activity <flow> to select a series of services in parallel which respond to the user's context, and the activity < empty> , and so on, by adding activities which set off the adaptation policy, i.e., the adaptation rules.

To trigger a BPEL contextual situation, we introduce a new activity <abpel: triggerContext>. To trigger the context, we specify the binding partner, port type, and source of context in the same manner as [18].

To catch (receive) a situation, we define the <abpel: catchContext > activity.

To trigger the adaptation policy, we specify the activity <abpel: TriggerAdaptationRule>.

To compare between any context (request) and the contexts (offers), we define the activity <abpel: MatcherContext>.

To activate the adaptation to the context, some researchers in this domain must answer some fundamental questions: Where to adapt? When to adapt it? How to adapt it? And what to adapt? , and so on. The question is how to describe the adaptation technique used? The most known techniques are [11]: the reflexivity and contract adaptation and the aspect weaving adaptation.

In our work, we also consider that an adaptation following an adaptation policy is necessary to define the rules according to which the composition process will react to adapt its behavior to changes of its context of use. These adaptation rules are based on the Event-Condition-Action (ECA) paradigm, which is inspired from the active data bases [21]: the event (E) describes the context changes, leading us to a situation identified by the conditions (C) to trigger an adaptation (A).

Definition 2 (ECA rule): A ECA R rule is defined as a triplet (E, C, A).

E is a finite set of contexts.
C is a finite set of conditions.
A is a finite set of adaptations.

We model a rule by a triplet (E, C, A). Figure 6 gives a simple example of an adaptation rule where the « event » part expresses the contextual situation: « the terminal does not withstand the ECG graphs. The action associated to this situation is « block access to the service ».

Rule ID
When context: device.PDA.
If PDA=No accepted ECG.
Do service. NO display ECG.

Figure 6. Example of an adaptation rule.

According to [22], we distinguish two types of adaptation rules depending on whether they respond to a change in context. This distinction gives raise to inter versus intra- context adaptation rules.

- Intra-context adaptation rules. The intra-context adaptation rule defines the context of use in which this rule is active. It carries out local transformations on a given service.

  In Ci when event if condition do action.

- Inter-context adaptation rules. The event that triggers the rule is here a change in context. We can, thus, precise that the inter-context adaptation rule makes transformations on the BPEL composition process. It carries out a set of modifications. For instance, it substitutes a given service by another. These adaptation rules are performed by the adaptation manager.

  When Ci->Cj if condition do action.

We have opted for the service technology to implement the service adapters. For every service, the adaptation manager instantiates an adaptor to ensure its adaptation. The syntax of the A-BPEL can be seen in Figure 7.

```
< process>
        <receive  partnerLinks = "partnerLinks"
                    Porttype="Port2"
                    Operation="Op2"
                    Variable=requestS1"
                    Name="receiveS1"
            </receive >
        <invoke   partnerLinks = "partnerLinks"
                    Porttype="PortS1"
                    Operation="Op1"
                    InputVariable=requestS2"
                    Name="receiveS2"
            < / invoke    >
          <sequence>
          <adaptation>
            < triggerContext >
              <context> . . . </ context>
              < MatcherContext >
                <partnerLinks > . . . </ partnerLinks>
                 <variables > . . . </ varibles>
                <sequence > . . . </ sequence>
               < / MatcherContext>
              <adaptation Policy>
                <partnerLinks > . . . </ partnerLinks>
                 <variables > . . . </ varibles>
                <sequence > . . . </ sequence>
              </adaptation Policy>
            </ triggerContext >
          </adaptation>
          <invoke/>
                 . . .
           </sequence>
</process>
```

Figure 7. The syntaxis of the A-BPEL.

4. CASE STUDY

To demonstrate how our work can contribute to the improvement of the services composition by adding a dynamic adaptation, we use a short scenario on the consultation of a patient's record to prescribe a treatment. Our motivating scenario concerns Mohamed, a patient aged 40 years and is being taken care of in a cardiothoracic service at Benbadis Hospital in Constantine. Mohamed connects to a mobile monitor ECG (PEM) [24], which is used to detect and manage his cardiac condition. An electrodiagram (ECG) is a test which registrates the heart's cardiac activity. The mobile monitor (PEM) detects the cardiac problems and sends an alarm to a medical center if necessary. This monitor PEM can identify types of alarm: serious and minor alarms. In the case of a minor alarm, the system selects the service "doctor" close to Mohammed's present location. When a doctor is assigned to Mohammed, he consults his medical report and checks up the ECG registered reference. Then, he diagnoses the case and prescribes an appropriate treatment for his case. In the case of a serious alarm, the emergency hospital localizes the patient in order to trigger other services.

To illustrate our suggestion, we present two contextual situations C1 and C2. C1 presents a situation where the user is a doctor localized in his office and uses a standard PC to consult the patient's medical report after having been sent an emergency alarm to prescribe a treatment. In the context C2, the doctor consults the same medical report at home by using a PDA. In both situations, the user interacts with the same process. However, the composition does not behave in the same way in both cases. The changes in the values of context parameters lead to changes in the service request: the locality (office and home) modifies the available list of services, the type of device used offers the services which bear its characteristics, for example to compress the document, substitute one service by another, and so on).

In the present paper, we have not dealt with the used adaptation rules, but we relied on some adaptation rules often found in the adaptation approaches such as substituting a service.

Figure 8 presents our scenario for the consultation of the medical report with a standard PC.

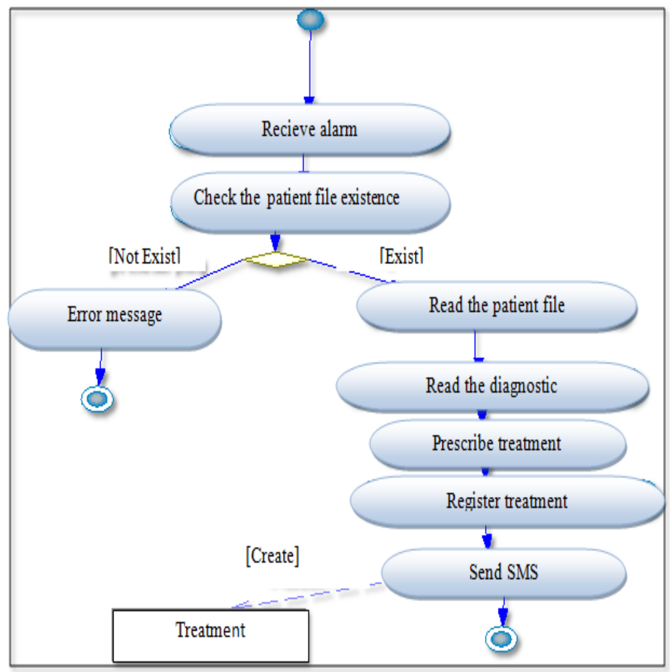

Figure 8. Consultation of the medical report by a standard PC.

Figure 9 presents a scenario adapted to the situation where the consultation of the medical report is done by a doctor with his PDA.

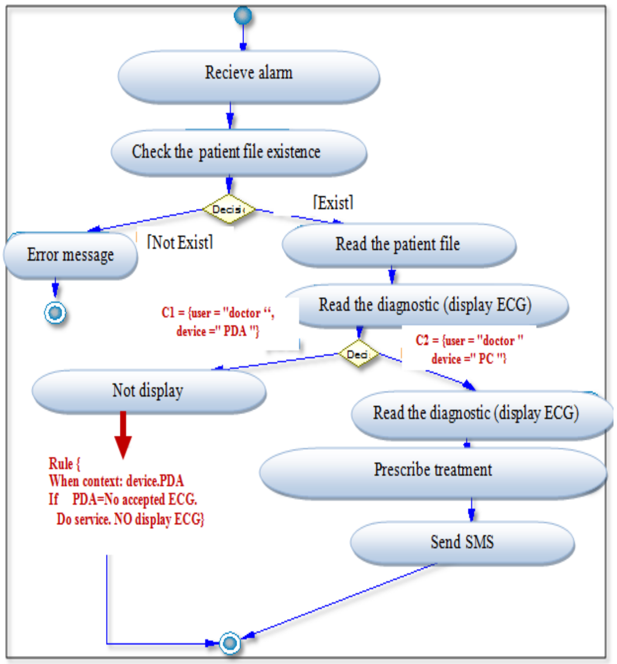

Figure 9. Consultation of the medical report with a PDA.

5. IMPLEMENTATION

To implement our approach, different development tools have been offered to us among which J2EE that we have opted for as a platform and the «java» as a programming language, due to many material and software constraints. Throughout the development of our prototype, we have been using the Apache Tomcat server. The publication of the register of our Web services has been established in a relational database to facilitate the treatment of data. Our system of dynamic composition of services that we have realized offers to users the right to access in order to ensure security. Figure 10 presents the authentication interface to gain access to our system.

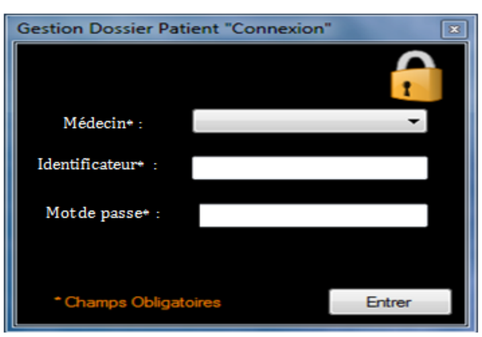

Figure 10. Authentication Interface.

The interface for handling the medical file uses context information specific to the user such as the patient's profile (first name, last name, address, blood group, and so on) (see figure 11).

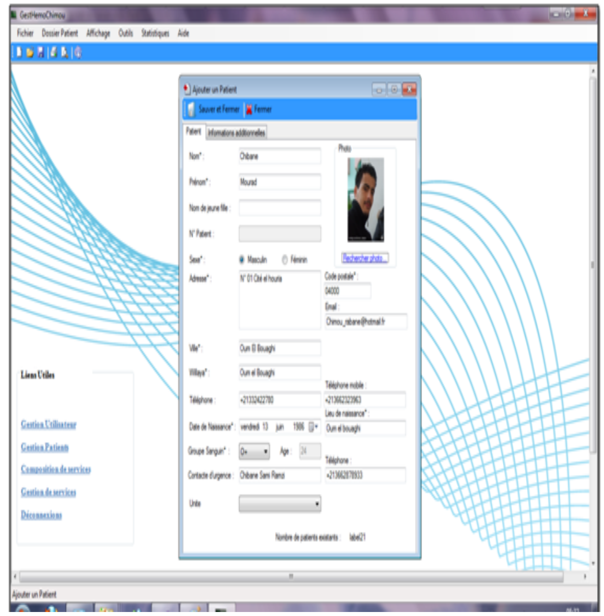

Figure 11. Interfaces corresponding to the handling of the medical file.

We have undertaken experiments to validate our approach of context sensitive composition approach. These experiments have been carried out on two different contextual situations by using the following parameters (user's role, type of terminal used, localization of the user). Consequently, we have selected a set of adaptative services for each situation in order to respond to the contextual situation characterized by the following parameters C1 = {user =" doctor ", device =" PDA ", location = " hospital "}. The service composition will be adapted according to the doctor's profile. We have, thus, searched for services capable of supporting the capacities of a Smartphone (for example, we compress data in small chunks). Figure 12 illustrates the result of the search for the best service which might adapt to this situation.

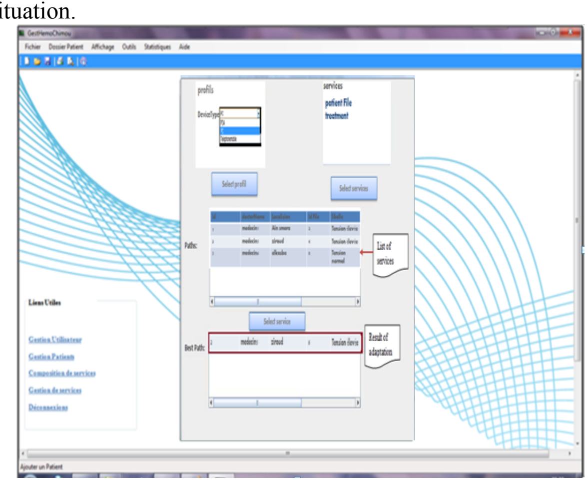

Figure 12. The search result corresponds to a given situation.

6. CONCLUSION

The adaptability and the flexibility are difficult questions in the pervasive environment. In this respect, the main contribution of our approach is the dynamic orchestration adaptive to the user's context and the service context. To make this approach realizable, we have used the service-oriented architecture in a way our solution allows services to act as producers and / or context consumers, in order to handle well heterogeneous context information sources. This architecture permits to the orchestrations of context-aware services. We have suggested extensions specific to WSDL and BPEL. With the WSDL extension, we have introduced context management and adaptability management.

The BPEL extension permits to the user to include the handling of the adaptability in order to create dynamically adaptable composition processes. We have also presented the manner through which the users may define their own context to trigger the adaptation of the composition process. With a simple scenario, we have demonstrated how our approach may be used to manage information of the context of a composition process and to adapt dynamically. The present approach may be improved by enriching our context model through the treatment of the semantic adaptation. On the other hand, we intend to test our approach with a more complex scenario and to evaluate it with a semantic base containing a broader range of service descriptions.

REFERENCES

[1] Arina, A. Business Process Modeling Language- BPML 1.0, 2002.

[2] Andrews, T. Business Process Execution Language for Web Services 1.1, May 2003.

[3] Al Kukhun, D. 2012. Steph Towards Adaptive Situation and Context-aware access: a contribution to the extension of access control mechanisms within pervasive information systems. Doctoral thesis, l'Université Toulouse III - Paul Sabatier.

[4] W3C, Web Services Description Language (WSDL) Version 2.0, W3C Recommendation, 2007.

[5] Bouyakoub, F, M Belkhir. 2013. A Context-Aware Web Service Discovery Based on a Quantitative Similarity Measure. International Journal of Modern Education & Computer Science, Vol. 5 Issue 8, 27-42.

[6] MacCaull, W and Rabbi, F. 2012. NOVA Workflow: A Workflow Management Tool Targeting Health Services. Delivery, in FHIES'11, ser. LNCS, vol. 7151. Berlin, Heidelberg: Springer- Verlag, 75–92.

[7] Heorhi, R. 2012. Service Composition in Dynamic Environments: From Theory to Practice. Doctoral thesis, Fondazione Bruno Kessler, University of Trento.

[8] Saif, U. Pham, H. Paluska, J, M. Waterman, J. and Ward, S. 2003. A case for Goal-oriented Programming Semantic*s*. In Workshop on System Support for Ubiquitous Computing, 5th International Conference on Ubiquitous Computing, Seattle, WA, USA.

[9] Vukovic, M and Robinson, P. 2004. Adaptive, Planning-based, Web Service Composition for Context awareness. In 2nd International Conference on Pervasive Computing, Vienna, Austria, April.

[10] Maamar, Z. Mostefaoui, S, K and, Yahyaoui, H. 2005. Toward an Agent-Based and Context-Oriented Approach for Web Services Composition, IEEE Transactions on Knowledge and Data Engineering 17, 686-697.

[11] Chaari**,**T.2007.Adaptation d'applications pervasives dans de s environnements multi-contextes**.** Doctoral thesis. Order No. 07-ISAL-0058 INSA de Lyon.

[12] Keidl, M and Kemper, A. 2004. Towards Context-Aware Adaptable Web Services, In Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters, May 19-21, New York, NY, USA, doi:10.1145/science.1013367.1013378.

[13] Adams, C. Boeyen, S. 2002. UDDI and WSDL extensions for web service: a Security Framework, in Proceedings of the ACM Workshop on XML Security, 30-35.

[14] D'Ambrogio, A. 2006. A model-driven WSDL extension for describing the QoS of web services, in Proceedings of the IEEE International Conference on Web Services, 789-796.

[15] Kloppmann, M. 2005. WS-BPEL extension for sub-processes – BPEL-SPE, in A Joint White Paper by IBM,September,2005.DOI=http://download.boulder.ibm.com/ibmdl/pub/software/dw/webservices/ws-bpelsubproc/ ws-bpelsubproc.pdf.

[16] Agrawal, A. 2007. WS-BPEL Extension for People (BPEL4People), Version 1.0, 2007. DOI=http://download.boulder.ibm.com/ibmdl/pub/software /dw/specs/wsbpel4people/ BPEL4People_v1.pdf.

[17] Juric, M, B. Sasa, A. Rozman, I. 2009. WS-BPEL extensions for versioning, Information and Software Technology 51 (8) 1261-1274.

[18] Juric, M, B. 2010. WSDL and BPEL extensions for Event Driven Architecture, Information and Software Technology 52(10), 1023-1043.

[19] Belgharbi, K. Boufaida, M. 2014. Flexible Composition of Services and Adaptation to Context in a Pervasive Environment. In international conference on artificial intelligence and information technology. Ouargla, Algeria. 45-60.

[20] Dey, A, K. Salber, D. 2001. A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications. Human-Computer Interaction (HCI) Journal, 16(2-4), 97-166.

[21] Bounaas, F.1995. Using ECA Rules for Object and Schema Evolution in an Object-oriented System. In Technology of Object-Oriented Languages and Systems.

[22] Canneau, V. Calvary, G. Demunieux, R. 2007. Métamodéle de Règle d'adaptation pour la Plasticité des Interfaces Homme-Machine. In proceeding of the 19th International Conference of the Association Francophone d'Interaction Homme-Machine. 91-98.

[23] Le Grand, B. Kirsch Pinheiro, M. Souveyet, C. Najar, S. 2013. Systèmes d'Information Pervasifs et Espaces de Services: Définition d'un cadre conceptuel. 9èmes journées francophones mobilité et ubiquité, Nancy (France).

[24] EPI-MEDICS: a project of the European Commission DOI=http://epi-medics.insa-lyon.fr.

[25] Weiser, M. 1991. The Computer for the 21st Century. Scientific American, September, 1991.

[26] Mani Chandy, K. 2006. Event-Driven Applications: Costs, Benefits and Design Approaches. California Institute of Technology.