Context-aware Quality Adaptation Using Rich Explicit Constraints in E-health System

Adel Altī
LRSD
University of Setif, Setif - Algeria
+21336632017
altiadeli2002@yahoo.fr

Sébastien Laborie
LIUPPA
University of Pau, Anglet - France
+33559574372
Sebastien.Laborie@iutbayonne.univ-pau.fr

Philippe Roose
LIUPPA
University of Pau, Anglet - France
+33559574348
Philippe.Roose@iutbayonne.univ-pau.fr

Abstract—One of the key aspects of any e-health application is the quality management of urgent situations. Currently, these situations are accessible on a wide variety of embedded sensors. The heterogeneity of such sensors and the diversity of user’s needs require management quality of service and adaptation to different critical situations (e.g., hypoglycemic diabetic coma). Since the last decade, a fair amount of research has been conducted in order to develop adaptation platforms. These platforms generally adapt services in order to comply with dynamic context evolution. However, we have noticed that current adaptation platforms do not fully exploit the semantic benefits for describing the heterogeneous contexts, the adaptation process. In this paper, we propose a model for specifying rich contexts containing explicit constraints expressions with qualitative and quantitative information. Our proposal has the great advantage to offer to users a global flexible adaptation infrastructure exploiting semantic information at multiple levels, i.e., from the design level to the run-time level. To demonstrate the utility of our approach, we propose the design of an ambient system applied to a diabetes case study.

Keywords—e-Health, architecture, context, emergency situation, quality dynamic adaptation.

I. INTRODUCTION

Recently, the World Health Organization emphasized that the future of e-Health depends on the ability to install a culture of mobile communication and quality in e-health services. The e-health services in a mobile environment have a critical role in creating user satisfaction. They are neither a uniform group, nor able to give consistently high service quality. Indeed they are significantly mobile embedded and limited (low bandwidth, power consumption, etc.) device has to firstly prevent interaction and mobility limitation. The heterogeneity of components regarding embedded sensors, CPU power, communication mechanisms (GPRS, WIFI, Bluetooth, etc.), speed of transmission as well as the media variety (sound, video, text and image). The heterogeneity of such applications and the diversity of user needs require management quality of service and adaptation to different urgent situations (e.g. diabetic coma).

Since the last decade, a fair amount of research has been conducted in order to develop adaptation platforms [6][10][11]. These platforms generally adapt services in order to comply with dynamic context evolution. For that purpose, they generally exploit high-level context models in order to identify situations. Furthermore, when conflictual situations have been identified, they usually search and compose several adaptation components in order to solve these situations. However, we have noticed that (1) current adaptation platforms do not take into account a semantic description of heterogeneous context data (e.g. blood glucose sensor provides blood glucose level data whatever unit of measure). (2) - none of them are portable, i.e., it is not possible to migrate a context model between different platforms without modifying multiple quantitative context values. (3) - none of them are expressing rich constraints. For instance, a user cannot specify the following constraint in his profile: “If it is lunch time, remind me to take some medicines using audio contents”. In other words, current context models do not support some specific desired actions under several conditions.

In this paper, we propose to extend our previous proposal ContextualArchRQMM (Contextual Architecture Requirement Quality MetaModel) [9], by adding adaptation process families to such metamodell. Moreover, we specify rich contexts containing explicit semantic constraints with qualitative and quantitative information, description of context model based on service references. Our approach focuses on description of rich semantic constraints model explicitly and to facilitate the system architecture quality control with the continuous evolution of its context. The major contributions are to integrate the rich context models to facilitate the management of e-health mobile applications at run-time level.

The rest of this paper is organized as follows: Section 2 details our rich context models and quality adaptation process, while section 3 presents an overview of the proposed framework. Section 4 presents application scenarios handling our metamodell. Section 5 presents some related works. Finally, Section 6 concludes and presents some future works.

II. ENHANCING EXPRESSIVENESS IN CONTEXT MODELS AND ADAPTATION PROCESSES

The general structure of Context-aware Quality – Model Driven Architecture (CQ-MDA) is presented in Fig. 1 [9]. We consider the full software development cycle within MDA, i.e. from formulation of needs up to the code generation. The proposed structure consists in five levels representing CIM,
PIM, Contextual Platform Independent Model (CPIM), Contextual Platform Specific Model (CPSM), and code. Each level is decomposed into three parts: the left part represents architectural artifacts and context concepts; the right part represents quality model and measurements done for these artifacts while the center part represents requirements. Two ways of using our metamodel are possible:

- The first one assumes that the software architecture quality metamodel is used for evaluating an architecture model. The architecture model is tested and validated with the semantic constraints defined by the metamodel. If the verified architecture model gets bad marks then the design process can be stopped or it can go back to the previous stage either to change requirements or to elaborate a better model.

- The second one, using software architecture quality metamodel considers the case when the metamodel is used for selecting the best architectural model from different choices. In this case the values of a metric are used to classify the models. A metric formula gives a mark for the architecture model. The values of the metric function are used to classify the models and to choose the suitable one and we select a first model if we have the same value. After that, the selected architectural model is evaluated by the OCL constraints to remove any quality semantic violation.

A. Semantic context modelling

A Context is a class that models the context information. In our design, the type Context is further distinguished into two subtypes: Atomic Context, Composite Context. Atomic contexts are low level contexts that mainly contain static value (e.g. screen size) or dynamic (e.g. battery level, preferred language, network bandwidth, blood glucose, GPS coordinates etc.). In contrast, composite contexts aggregate multiple contexts, either atomic or composite.

To solve the challenge of heterogeneous and dynamic context information, our context model refers to a set of services, from which context information is retrieved, like the user location service, the blood glucose service, the battery power service. These services can be categorized into facets in order to better retrieve them. We distinguish three facets: the context facet, the document facet and the hardware facet. Our context model refers to services, we can migrate the context from platforms to platforms by preserving the constraints. Another advantage of this context model is that we do not need to update the model at any time, only specific services may be called for providing the only needed information.

As it can be seen in Fig. 2, our design models the situation of contextual adaptation where several explicit constraints with complex sets of services can be automatically executed. In order to enhance our context model expressiveness, we develop a framework that bridges the gap between quantitative and qualitative information specified in context models. Adding qualitative terms inside context model is not straightforward. Indeed, a qualitative term, like “Low”, may be applied on several context aspects, such as the device battery level, the blood glucose level, the bandwidth. Hence, the meaning of a term can be completely different depending on the context used. Moreover, even in a specific context, for several domains, a term may have different interpretations. For instance, the term “Low” in the context of the battery level corresponds to different quantitative values if we consider a smartphone or a laptop. As shown in [3], this multi-level approach for bridging the gap between qualitative and quantitative information described in profiles is generic. Fig. 3 shows a context UML meta-model. A DescriptiveWord corresponds to a qualitative term (Level 1). Each qualitative term is attached to one or more Context (Level 2). Each Context can be associated to one or more domain or subdomains (Level 3 and i). Finally, a Constraint associates quantitative values to the qualitative term depending on the context used (Level n).

B. A new situation similarity measure

In order to integrate other profile descriptions, we define semantic relationships between each pair of context concepts, such as EXACT, SUBSUMES (i.e., the concept is specific/generic as another one), NEAREST-NEIGHBOUR (i.e., the concept is nearest neighbour of another) and FAIL (i.e., the concept does not match any concepts). When the situation identification is not possible, the identification can be done by replacing one or more context concepts by replacing it by another one (or another set of concepts) within the same facet or domain. Semantic relationships may be useful for the reaching a precision of situation identification.
There are several possible ways to identify situation [6] [8], and a proper identifying generic technique still has to be defined. Computing similarities is also quite difficult when integrating heterogeneous user profiles.

Several facet services composing a situation are generally connected semantically. Our similarity measure extends the properties of $Sim$ defined in [6]. It is formalized as follows:

$$Sim(Q, S) = \frac{a}{a+b} = a \frac{\sum_{i=1}^{n} w_i \times sima(Q_i, S_i)}{a \sum_{i=1}^{n} w_i + b \sum_{i=1}^{n} w_{a_i}}$$ (1)

Where “$a$” is the set of common concepts of $Q$ (current situation of user) and $S_i$ (profile constraint) for the same domain and “$b$” is the set of concepts of $Q$ and not existing in $S_i$ sima is the atomic similarity between each context concept of situation $Q$. $S$. It is defined as function that maps two concepts to the interval $[0,1]$.

C. Semantic adaptation process modelling

Performing adaptation processes at minimum time with minimum resources are always the most important purposes of users. Times included in this case are divided in to three groups: necessary time to analyze semantic constraints, processing length of time and decision making and finally time of adapted media transfert anticipating these times based on included criteria’s in any adaptation stage and their performance is possible.

We can model the adaptation process as a (possibly infinite) set of finite adaptation paths, where each path defines a sequence of adaptation services in which the first element in the path is the service of the current context, and the final element is a desired target service. Links between successive nodes in a path are associated with transitions that are selected from a set of multimedia adaptation services. Our metamodel proposes to define adaptation family to capture a non-predefined number of adaptation paths having close adaptation services (see Fig.4). For example at image family which includes services offering services of the same nature (i.e. image adaptation services) but only differs by their adaptability to the context (e.g. adaptation time, transfert time).

Our adaptation decision process uses a QoS function in order to select a quality adaptation path. From Eq. (1), we can observe that QoS metric is a ratio of weighted sum of benefits to that of costs of adaptation services. The QoS function is used for selecting the best adaptation path that has a higher ratio Benefice/Cost. In this case the values of a quality formula are used for classifying the relevant adaptation paths that have potential benefit. The evaluation of the adaptation paths having the same mark of benefice metric and differs only by their adaptability cost to the context.
closeBayonneCity analyzes and builds the high-level context model low-level context information from distributed sensors, adaptation process and context-aware meta-models. The comparison of adaptation path scores allows us to select the best path.

III. THE FRAMEWORK

Fig. 5 illustrates the architecture of our framework on adaptation process and context-aware meta-models.

Context model building. The Context Collector collects low-level context information from distributed sensors, analyzes and builds the high-level context model (Table 1).

<table>
<thead>
<tr>
<th>Sensor</th>
<th>Contexts</th>
<th>Context values</th>
<th>Interpretation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Glucose Sensor</td>
<td>Glucose Level (mg/dl)</td>
<td>50</td>
<td>VeryLowGlucoseLevel</td>
</tr>
<tr>
<td>Weight sensor</td>
<td>Weight (calorie)</td>
<td>256</td>
<td>VeryHighWeightLevel</td>
</tr>
<tr>
<td>GPS sensor</td>
<td>Longitude</td>
<td>43°29′42.01&quot;</td>
<td>CloseBayonneCity</td>
</tr>
<tr>
<td></td>
<td>Latitude</td>
<td>5°28′19.000&quot;</td>
<td>InsideHomeLocation</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2013-07-28 09:30pm</td>
<td>BeforeDinnerPeriod</td>
</tr>
</tbody>
</table>

Situation matching algorithm. Each situation consists of two or more conditions. The conditions are joined with the AND operator. An example of situation rule is as follows:

IF VeryLowGlucoseLevel AND VeryHighWeightLevel AND closeBayonneCity AND BeforeDinnerPeriod THEN UrgentDiabeticSituation

The Inference engine (Algorithm 1) aims at matching the current situation with each constraint defined in the user’s profile. It takes a list of current situation’s concepts as input in order to calculate atomic similarity sima of each pair of concepts (of situation and constraint). If the match is not a Fail, it returns the overall score Sim of each constraints and appends the advertisement to the result set. Finally the result set is returned in ranked way. In so doing, we select the higher matching measure.

Quality adaptation path decision. As soon as an adaptation event is received (e.g., situation change and/or user context changes), the Adaptation Planner and Decisioner will be able to evaluate adaptation path quality (benefice, cost) among a sub-family (or a family) that corresponds to a current situation and meets user preferences (expressed as constraints) and resource constraint (CPU power, battery level, available bandwith, etc.). For example, suppose the following explicit constraint:

$$W_{\text{benefice}} \times \sum_{s \in \text{path}} \text{Benefice} (s)$$

$$W_{\text{cost}} \times \sum_{s \in \text{path}} \text{Cost} (s)$$

$$QoS (\text{path}) = \frac{W_{\text{benefice}} \times \sum_{s \in \text{path}} \text{Benefice} (s)}{W_{\text{cost}} \times \sum_{s \in \text{path}} \text{Cost} (s)}$$

Where $W_{\text{benefice}}$ and $W_{\text{cost}}$ are weights associated respectively to the mark of benefice and cost. These weights cannot be set directly by the user. A user may specify in advance that the execution time is not a constraint and that he prefers a hi-quality video, and these weights are then computed automatically during the profile creation. All these weights are set between [0, 1]. The comparison of adaptation path scores allows us to select the best path.

Algorithm 1: Situation Matching Algorithm

Input: Profile [], profile Status[, two contexts C1 and C2 / C: S and C: Q
Output: overall score Sim (Q,S)

Step1: Get each constraint the atomic similarity value of each common context attribute (C, C).
Step2: Get each constraint an overall similarity value defined in [1]
Step3: Return the higher matching score with situation matching relation.

IF UrgentDiabeticSituation AND LowBatteryLevel exclude videos AND include image instead. This component translates this qualitative constraint into a quantitative constraint, e.g., if my battery is less than 15%, exclude videos and include image instead for any mobile devices. From a user context model and a current situation if we confirm that we must exclude videos and include images (because the current battery level is less than 15%), the adaptation process should start by evaluating and selecting an adaptation path form video to image subfamily (video family) under a valid device codec. Of course, the adaptation is customizable and parametrizable according to different device of users (Tablet or Smartphone). For instance, it may select an adaptation path that produces rapidly an adapted content with Samsung Smartphone under its implicit constraints (device resolution).

Call and run adaptation services. The Multimedia Server runs a sequence of transformation actions on the original media and provides an adapted media for each professional caregivers and alerts.
IV. A DIABETES CASE STUDY

Our validation is based on the following case study. User A is an elderly patient who has been managing diabetes. Recently, he joined a persuasive telehealth trial study. User A has been provided devices (glucose meter, weight scale, video camera and GPS) to influence him to promote healthy behaviors. The profile information is utilized in aware to identify urgent situations. For example, the situation of "hypoglycemic diabetic coma" is deduced by referencing the average of glucose level and weight in the profile (Table 2).

<table>
<thead>
<tr>
<th>Condition</th>
<th>Alert situation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Glucose 'very low'</td>
<td>hypoglycemic diabetic coma</td>
</tr>
<tr>
<td>weight is 'very High'</td>
<td></td>
</tr>
<tr>
<td>Location is 'Inside_Home'</td>
<td></td>
</tr>
<tr>
<td>Period is 'before Dinner'</td>
<td></td>
</tr>
<tr>
<td>Situation is diabetic coma</td>
<td>Alarm</td>
</tr>
<tr>
<td>Walk is 'less'</td>
<td></td>
</tr>
</tbody>
</table>

To make better informed choices, help User A manage their situations in more practical way by User B (e.g. doctor), which has a Smartphone; video should be delivered in quality and in period no longer than one minute from their request.

A. Examples of scenarios

A first simple example is made of a user B which wants to share the elder's short video captured about 15 minutes using a Samsung Smartphone. There is atomic adaptation service that can fit the device constraint (e.g. a low screen resolution of the Samsung Smartphone = 200 x 200 is inferior to the document pictures resolution 400 x 400). The planner is used as a basis for finding adaptation services that matches the Samsung Smartphone context (i.e. the less screen size). Two re-sizing services: a black and white picture and a coloured picture. To limit the search results when browsing the elder's short video on the Samsung Smartphone, only the first one is returned that matches the Samsung Smartphone context (i.e. the less screen size) and the quality (i.e. coloured picture). A second scenario is when the adapted document is ready to be executed. User B receives a phone call to join his colleagues at the work. The analyzer has noticed a problem of bandwith, and the profile is considered almost 30 adaptation services. Each service is offered by five providers with different qualities in a small environment deployed on 3 heterogeneous platforms (Table 3).

B. Experiments results and discussion

In order to validate the feasibility of our approach, we have considered almost 30 adaptation services. Each service is offered by five providers with different qualities in a small environment deployed on 3 heterogeneous platforms (Table 3).

TABLE III. EVALUATION CONFIGURATIONS.

1. Android Smartphone connected Wi-Fi with RAM 512 MB processor (0.8 GHz)
2. Android 3.2 Tablet with 1GB of RAM a double heart Tegra 2 processor (1GHz)
3. Windows 7 laptop with 8GB of RAM and i5-quadruple heart processor (2GHz)

In Fig. 7, we compare the response time (adaptation time and transfert time) of elder's short video on three platforms (same context model). We find that the response time is four times faster in the configuration#3 using our adaptation strategy. These experiments allow us to consider the response time on three configurations with/without adaptation. We find our approach has a good performance in adapting the elder's short video using adaptation process families. Based on these experiments given in this section our approach improves the accuracy in comparison with techniques without adaptation. However, the platform seems to have impact on adaptation time. Thus, we want to continue, the evaluation of our quality adaptation technique on other platforms in order to define an optimized strategy.
V. RELATED WORKS

The first related areas of research are smart home projects that have been proposed for monitoring of the elderly people at home [4] [5]. However, these works only have kept the focus under the old people welfare. Robles and Kim [12] concluded in their review that a main challenge of installing a smart home system is balancing the complexity of the system against its usability. In [7] a framework for assisting elders at home called ANGEHLA was proposed. That work has presented a solution based in a middleware that integrates a network of sensors and actuators such as: Cameras, Microphones, GPS, and Infrared etc. There is a central Human Interaction that monitors the elderly people and manages the members for aid situations. In emergency cases as fall, the elder will send a SOS message through a wearable device (RFID card) to the center. Then, the center gathers and sends members (unknown people are available) that are the closest to the elderly who needs help.

The second related area of research are some works involving adaptations of component-based applications refer to the capacity of a system to adapt to the evolving needs of the users and the context by exploiting knowledge on its configuration and the characteristics of QoS of its constitutive components. The adaptation based planning [2] and [1] is one of the adaptation approaches of component-based applications. Relevant work to our model is reported in [6] and [3]. In [6], Anagnostopoulos et al. introduces imprecise knowledge reasoning and decision making for determining and reasoning about the current situation. This work concentrate only on context-aware system architecture, while our proposal i) provide multi-types model supporting, ii)- provide descriptive information and explicit constraints expressions, and iii)- provide quality context management in different mobile platforms. Compared to other work [3], our approach reasons about adaptation benefice/cost, deals with satisfying different user preferences, allowing users to best use context resources and restricting the of finite sets of adaptation paths that corresponds to situation identified from device sensing.

VI. CONCLUSIONS AND PERSPECTIVES

In this paper, we propose to extend our previous proposal ContextualArchRQMM (Contextual ARCHitecture Requirement Quality MetaModel) [9], by adding adaptation process families and explicit constraints. At run-time, our framework copes with the challenges posed by the highly dynamic nature of mobile systems through continuous monitoring, identification of situation and calculation of the most suitable adaptation path. If a better path is found, the framework adapts at run-time the software, potentially via intelligent adaptation and mobility. We presented a diabetes case study to illustrate the applicability of the proposed approach. The experiment shows that our approach outperforms adaptation ratio and time response. In the future, we want to continue, the evaluation of our quality adaptation technique on other platforms in order to define an optimized strategy.

VII. REFERENCES