Context Information Refinement for Pervasive Medical Systems

Dawit Bekele  
*Internet Society, African Regional Bureau*  
Addis Ababa, Ethiopia  
Email: bekele@isoc.org

Birhanu Eshete, Adolfo Villafiorita, Komminist Weldemariam  
*Center for Information Technology*  
(fbk-irst) Trento, Italy  
Email: (eshete,adolfo,sisai)@fbk.eu

Abstract—Emerging technologies like mobile and wireless communication are offering promising opportunities to enable mobile healthcare delivery to citizens. But, enhancing the quality of service of such systems demands systematic improvement of existing service infrastructure.

In this paper, we describe a context information refinement architecture proposed to address the shortcomings of existing works in relation to context information refinement in pervasive medical systems. The shortcomings are lack of adequate consideration for: quality parameters of context information, relevance of context information and particular requirements of the pervasive healthcare domain. The proposed architecture facilitates and coordinates the refinement of context information starting from acquisition of context information up until the refined context information is delivered to the target application in a pervasive medical system.

We developed a prototype that implements the core components of the proposed architecture and evaluated it with real-life pervasive healthcare scenario and proved the validity of the architecture using a real-life scenario.

Keywords—Context, Information Refinement, Medical Systems, Pervasive.

I. INTRODUCTION

Pervasive computing uses the increasing number of mobile computing devices available to provide information and services needed by users anywhere, on various devices and at any time [1], [2], [3]. The new trend created by pervasive computing is attracting a large number of researchers from many application domains such as e-Government, e-Commerce and e-Health, to mention few [4], [5], [6]. In particular, it has tremendous applications in the medical field as workflow of medical professionals is characterized by high degree of mobility and they have high demand of getting the right information at the right time and place so as to facilitate their day-to-day activity.

One specific feature of pervasive computing interesting for medical field is context-aware computing whose primary concern is providing information relevant to location, time, activity and environment of a user. However, the management of context information remains a challenge due to its inherent dynamic nature. Such dynamicity is more outstanding for pervasive healthcare systems, characterized by high degree of mobility.

Not surprisingly, due to high degree of mobility and frequent demand for information at the point of decision, healthcare professionals rely on the information they receive from the underlying computing infrastructure to pass each decision about their patients. Therefore, the quality of decision made by medical practitioners is influenced by the quality of context information received from the underlying context-aware pervasive computing facility [7], [8], [9], [10]. In the existing models, the major focus is on how to represent, classify and interpret context information. However, what makes the problem even challenging today is, in our opinion, the refinement of context information, with the right quality and relevance of context information while considering the nature of the domain.

In this paper, we propose context refinement architecture to refine context information in pervasive medical systems ensuring the issues just mentioned are addressed. We do so by introducing a separate service parameter specification to allow physicians put quality constraint on the service while enhancing reasoning of existing models via an additional decision layer and capturing the domain characteristics through domain ontology.

In the next section, we discuss the background pertinent to this work. Section III presents the detail of the proposed architecture. The prototype implementation and demonstration are discussed in Section IV. Finally, we present related work and conclusion in Sections V and VI respectively.

II. BACKGROUND

Context is characterized by range of temporal characteristics, many alternative representations, and high degree of inter-relationship between different context types. Quality of context (QoC) refers to any information that characterizes the worth of information and it has accuracy, probability of correctness, level of trust, resolution and freshness as distinguishing parameters [11], [12].

It is trivial that, before storing context information, it should be modeled. The popular context modeling approaches are based on data structures used for representing and exchanging context information. In light of this, there are six context-modeling techniques discussed in [13], [14], [15], [16]. In this paper, however, we consider only ontology-based model as it offers high and formal expressiveness and possibilities for applying reasoning techniques.
that allows the user to specify the potential determinants of quality and relevance of context information. Moreover, it allows the user to modify or remove existing service parameters if needed. This enables the Service Parameters Manager (SPM), on the context refinement server, to maintain up-to-date list of service parameters specified by the user and based on which the Decision Component (DC) commits its final decision.

The Event Listener (EL) is in charge of continuous listening to incoming notifications from the decision component and forward the notification to the user. It possesses a continuous communication capability such that it binds itself to a particular communication channel (e.g., a port number) so as to immediately accept all incoming messages through the channel. As opposed to interval-based polling [16], [21], it subscribes to list of events and is notified on occurrence of one of the events that it subscribed for. Such a subscription-based approach is selected for this component due to the unpredictable nature of notification arrivals and the frequent change of the underlying context like sensor readings of a patient and current context (e.g., location) of a user.

The Device Tracker (DT) is responsible for periodically identifying and submitting current context information (location and identity) of the device to the Context Acquisition Component (CAC). It also maintains local information stored on the device, by interacting with the on-device persistent record manager. The presence of on-device persistent record manager is of significant contribution to reduce the impact of intermittent wireless connections through caching the most frequently accessed information. Like the EL, it needs to bind itself to a particular channel to which it periodically delivers the current context of the device that in turn indicates the current context of the device owner (user) provided that the device-to-user (ownership) association is defined in advance. As opposed to the EL, it operates in a synchronous manner since it is possible to set a certain minimum interval within which the context of the device changes. For instance, it may be configured to submit change
to current context (e.g., location) of the device every five seconds after analyzing the interval of context change in practice.

B. Components on the Refinement Server

The SPM is in charge of maintaining up-to-date list of service parameters to be immediately consumed by the DC. It needs to communicate with the SPSI to accept the list of parameters specified by the user and the DC to provide the list of up-to-date service parameters that enable the DC to commit delivery of context information. Moreover, insertion of new, modification and removal of existing service parameter records that could be initiated on the SPSI are handled by this component for which it needs to interact with the Service Parameters Database, in which list of service parameters is structured into manageable parameter attributes that can be represented in a relational database.

The Reasoning and Decision Engine (RDE) is the central component of the architecture that communicates and cooperates with all other components. It performs ontology supported, rule-based reasoning to pass final decision on whether or not to deliver the currently ready context information to the target application on the client mobile device. In particular, the Interpretation Component (IC) does the rule-based reasoning while the final decision of whether or not to deliver the context information is carried out by the DC. It communicates with

- the DT for periodically receiving the current context (location and identity) of the device and sensor supplied data through the CAC;
- the Electronic Medical Record (EMR) to retrieve the necessary medical history to support the interpretation by the IC;
- the Context knowledge base for accessing context history and reasoning rules through the IC;
- the SPM to get access to the recent list of service parameters through the DC; and
- the EL that accepts final notifications decided to be delivered to a target application on the client device through the DC.

The IC goes through the following steps: First, it periodically accepts current context (location and identity) and sensor information through the CAC on the context refinement server.

Second, it determines whether new context knowledge can be derived from the accepted context information. For example, if it receives context information that shows a physician is located in room 205, it is possible to determine whether new context knowledge can be inferred or not by analyzing the physician-to-patient association from the context ontology or the EMR. Furthermore, if one of the sensors reports a context information in a form (e.g., Patient ID, Blood Pressure, Heart Beat, Body Temperature, Sensing Time), the IC can determine whether new context knowledge can be derived or not by analyzing the service parameters-to-patients association in the service parameters database. If it is not possible to infer new context knowledge from the current context, it stores the information to the context knowledge base because this information can be used for historically reasoning in the future.

Third, it applies reasoning rules to derive new context knowledge that it hands over to the DC. The reasoning rules can be ontology-based or user-defined reasoning rules which associate specific actions to a combination of various conditions. While deriving new context knowledge, it may need to access the EMR to enrich the semantics of the derived context information which in turn enhances the usability of the context information for decision-making.

Finally, it hands over the newly derived context information to the DC.

The DC is responsible for accepting context information prepared by the IC and correlates the information it receives with the recent service parameters by consulting the SPM. Based on the result of correlation, it commits the final decision of whether or not to deliver the information to the client device. If delivery is decided, it immediately sends notification to the specific device through the EL on the client mobile device. Eventually, it appends the notification information to the context knowledge base so that the context knowledge base is stocked with richer set of context over time. Otherwise, it waits for the next handover of information from the CAC.

The CAC performs raw data acquisition and aggregation of data it collects from the various potential sources (e.g., sensor, device tracker) of context and eventually delivers to the IC. It also prepares the collected data in a format consumable by the IC. The introduction of the CAC allows to ensure clear separation of role between data collection and data interpretation to reduce dependency of one on the other. Such separation of role simplifies the task of changing or enhancing one of the components without significantly affecting the other. Furthermore, it creates insulation between the complexity and diversity of low-level context sensing and high-level context interpretation, an advantage for the IC to use ready-made context information and relieved of low-level data sensing.

IV. Prototype Implementation and Demonstration

In this section, we present the experimental setup used to demonstrate the validity of the proposed architecture and show the use of the architecture with real-life scenario.

A. Experiment

The prototype system is implemented as an application with the client-side on a Personal Digital Assistant (PDA) and the server-side on a personal computer. The two sides
communicate over a Wi-Fi (Wireless Fidelity) LAN. On the PDA, the SPSI is implemented as a J2ME [22], [23] MIDlet while the EL and the DT are implemented as J2ME threads which bind themselves to a certain port to communicate with the services on the server over HTTP. On the context refinement server side, the SPM, the RDE, the CAC and the Sensor are implemented as Java Servlets. The context knowledge base and the context ontology are implemented as RDF [24] schema and OWL ontology documents respectively. The EMR and the Service Parameters Database are implemented as MySQL relational databases.

The following real-life, experimental scenario is used to demonstrate the prototype implementation of the architecture.

In a certain hospital, a physician is in charge of following up multiple patients sleeping in different rooms. In a morning of a certain day, the physician looks at his to-do list on his PDA and realizes that his first duty is to make a ward-round of his patients. He walks to room 101 and eventually approaches the gate of the room. Right after he gets in, his PDA notifies him with the history of patient P1. He uses the notification information to decide the type of measure to take concerning patient P1 and leaves the room after some time. Then after, he walks to room 201 to visit patient P2. As soon as he reaches the doorstep of the room, his PDA alerts him again. This time, the patient history displayed on the screen is that of patient P2. He observed the status of the patient and left the room to get back to his office. While he is walking to his office, his PDA alerts him once again. He was wondering if he was around a room of the patients. However, this time what is displayed on the screen of his PDA indicates that blood pressure and temperature of patient P3 in room 103 is out of the normal range that he specified an hour ago. Hence, he quickly goes to room 103 so as to give emergency treatment to patient P3.

B. Demonstration

For evaluating the implemented features, we maintained an EMR of Physicians and multiple patients. The patients are also assigned rooms and a physician to take care of them. The SPSI on the PDA is used to specify constraints on the services using constraints on vital signs, location, time of the day and special parameters like when the physician is on critical tasks like surgery.

After specifying the parameters, the context refinement server notifies the physician either when he/she is at (near depending the distance accuracy) the room of his/her patients. Figure 2 shows automatic notifications from the context refinement server indicating that the physician is at room 201 at a certain time. The notification displays recent vital signs and diagnosis history of the patient sleeping in that particular room. Such a notification assists the physician to make informed decision for each patient while minimizing the effort of the physician to retrieve patient history.

The other kind of notification provided by the context refinement server is when the sensor report indicates abnormal vital signs of a patient. In such a case, the physician is notified irrespective of his/her current location so as to take urgent measures. Figure 3 shows the notification message for a patient alerting the physician when the temperature and heartbeat of his patient is out of the normal range by the physician.

V. RELATED WORK

From the early 1990’s, context information management in pervasive systems has been subject of several research works (see, e.g., [8], [11], [9], [25], [26], [27], [28], [29], [17]). These works vary in aspects like the software architecture they follow, the principal issues they address, the domain of application and the models they utilize for context modeling and reasoning. In this section, four of the related works are presented with an emphasis on their primary goals, the issues addressed, the gaps we have identified and the comparison of each to our work.

In [9], a context-aware application in hospital environment is presented. It demonstrates entrance of a nurse to an active zone, context-aware bed, context-aware pill container and context-aware electronic patient record. It also details how the context information assists in decision-making in the daily activities of the clinicians. Furthermore, critical design principles, one of which is the quality of context information, are listed to be crucial in the course of providing context-aware pervasive computing facility in hospitals. This work is similar to our approach in that the context-aware infrastructure is specific to healthcare environments and
the distinctive characteristics of pervasive medical systems, such as mobility, are well considered in the design of the infrastructure. The context representation model is an object-oriented model that suffers from expressiveness power that hinders context reasoning. Our approach, however, uses ontology-based context representation that is known of its expressive power to facilitate context reasoning.

In [11], the goal is to formalize issues with implications on quality of context information. In particular, the important quality parameters are described. The relationship and interdependence among QoC, QoD (Quality of Device) and QoS (Quality of Service) are also mentioned stressing that these three issues are different but inter-dependent. Finally, the paper formulates a role model for context-aware services with emphasis on the importance of refinement before a context-aware service attempts to deliver context information. The proposed model is different from our work in two aspects. Firstly, the model is domain neutral while ours is domain specific. Secondly, the role of the quality parameters formulated is not indicated in the model while we have considered the QoC parameters in the context refinement process.

In [25], modeling, storage and retrieval issues of context information are explained in relation to context identification and inference. In the presented architecture, the context management sub-system is one separate component along with context sensors and context actuators and is composed of context inference engine, context base, context pattern and context pattern mining sub-component. Even if ontology-based context modeling is not utilized for the context management sub-system, it is recommended for enhancing the context inference.

In [8], core to the service is the context ontology to model a healthcare domain. The context management service targets at correlating predefined context information with current situation of actors (like patients, physicians and nurses). This work resembles our work in that it utilizes ontology for context representation in pervasive healthcare even though the ontology is not utilized for reasoning.

VI. CONCLUSION AND FUTURE WORK

Delivering services with acceptable quality is critical in healthcare systems running over pervasive infrastructure. In this paper, we described a context information refinement architecture proposed to address the shortcomings of existing context management models for pervasive medical systems with regards to context information refinement.

We introduced the SPIS and SPM used to explicitly put constraints on quality parameters of context information. The relevance of context information is ensured by the DC indicating the preference of the user. The dynamic nature of pervasive medical systems is addressed by formulating a pervasive healthcare ontology. We also conducted an encouraging initial demonstration of how the architecture facilitates refinement of context information by considering the quality and relevance parameters specified by medical professionals and the particular characteristics of the pervasive medical domain.

Although delivering healthcare services with better quality of services needs integration of several services and institutions, we believe such an enhancement of existing service infrastructures, combined with other similar efforts, can ensure better healthcare service delivery to the society. Though important, we have not considered privacy and security aspect of context information and we leave it as future work. Components like the device tracker and the sensor were partly simulated due to lack of adequate devices. Hence, enhancing the prototype implementation and evaluation of its performance with large number of users is also left as a future work.

REFERENCES


