Virtual Remote Nursing System

Mehran Najafi
Computing and Software Dep.
McMaster University
Hamilton, Canada
najafm@mcmaster.ca

Shima Aghtar
Computing and Software Dep.
McMaster University
Hamilton, Canada
aghtars@mcmaster.ca

Kamran Sartipi
Computing and Software Dep.
McMaster University
Hamilton, Canada
sartipi@mcmaster.ca

Categories and Subject Descriptors
I.2.11 [**Computing Methodologies**]: Distributed Artificial Intelligence—*intelligent agents*; J.3 [**Computer Applications**]: Life And Medical Science—*health, medical information system*; D.2.11 [**Software**]: Software Architecture—*service oriented architecture (SOA), domain-specific architecture*

General Terms
Design

Keywords
Virtual Nurse, eHealth, Generic Software Agent, Task, Medical Guideline, SOA, Web Service

ABSTRACT
New demands for better public health services has enhanced the sate of electronic health (eHealth) systems towards concepts such as patient-centric, ease of use, smarter interactions, and accuracy of decisions. While eHealth services aim to provide continuous medical and health services for public and professionals, limitation of access to experts such as nursing is still a challenge. In this paper, we propose a new framework, namely Virtual Remote Nursing (VRN) that provides a virtual nurse agent installed on the client's personal computer or smart phone to control the client's health condition continuously. In this approach, medical practitioners can assign different tasks to a virtual nurse using a generic task definition mechanism, where a task is defined as a combination of medical workflow, operational guidelines, and associated data. A VRN is controlled by the practitioners who decide on the patient's treatment. Therefore, a VRN acts as a personalized and full-time nurse for its client that performs the practitioner's tasks on the client's health information. Such patient information can be obtained from a Personal Health Record (PHR) system such as Google Health or Microsoft Health Vault. We have developed a prototype system that enables traditional client applications and healthcare provider systems to collaborate using a VRN system. Finally, through a case study, we demonstrate how diabetic patients can take advantages of this system.

1. INTRODUCTION
Healthcare is the main cause of governmental expenditures around the world, and healthcare systems are under pressure to lower costs and improve outcomes. eHealth technologies are presented to address the healthcare problems using Electronic Medical Record (EMR) systems, Patient Health Record (PHR) systems, Clinical Decision Support Systems (CDSS), and telemedicine applications.

Smart Internet, also known as user-centric Internet, encourage individuals to use PHR systems to track their health situations. PHR systems such as Google health or Microsoft health vault are growing fast in a way that in the near future, every person is likely to have a PHR account. Similarly, the usage of mobile medical devices is becoming increasingly important and popular. Furthermore, the current medical devices that are integrated with the PHR systems enable updates on the patient's health status continuously and automatically. Therefore, a high volume of medical information that must be interpreted by healthcare professionals is unavoidable. However, the healthcare domain suffers from the lack of enough professionals, such as physicians and nurses, to process all this data which could limit the PHR applications.

A software agent is a piece of software that acts on behalf of an agency to serve a user. Software agents have been applied in different domains to model roles and practices such as financial adviser in banking, fraud detector in insurance, and trip planner in entertainment. However, agent computing has not been widely adopted in the healthcare domain. Software agents tend to be autonomous while in the healthcare domain decision making is a critical task that must be performed under a health professional control.

We propose an approach to simulate the functionality of nurses remotely, hence we call it Virtual Remote Nursing (VRN) system. In this approach a generic software agent, as virtual Nurse (vNurse), is installed on the client's personal computer or smart phone. Each medical practitioner can define a different task for this generic agent. In other word, the medical practitioner tells the virtual nurse *what to*

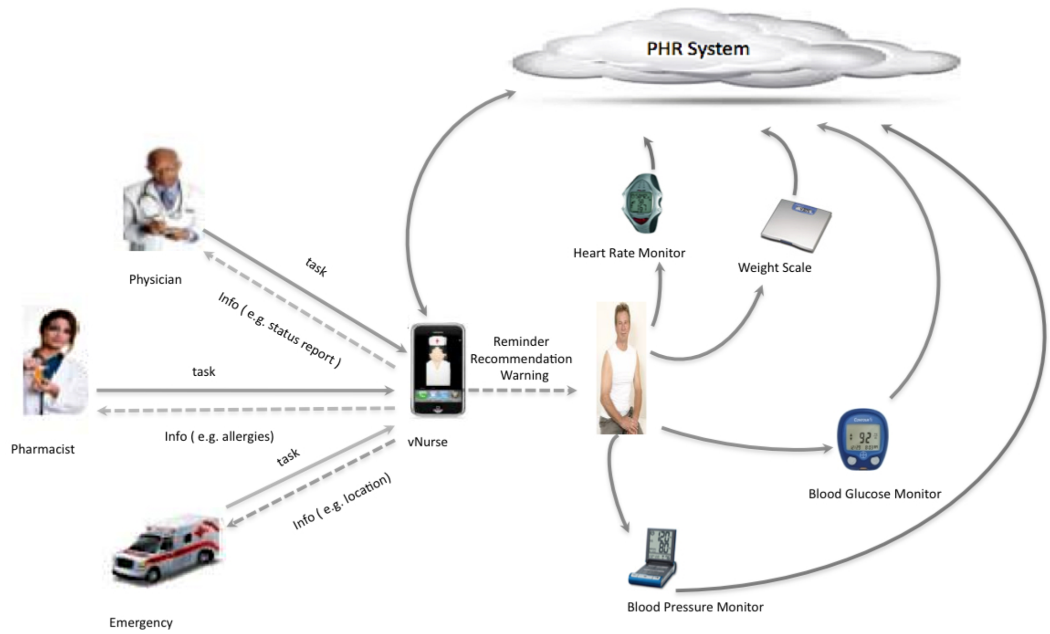


Figure 1: The high level view of the Virtual Remote Nursing.

do and *how to do it* for each assigned task and then the virtual nurse will perform the task. Consequently, the medical practitioner is the decision maker and the virtual nurse is the agent that performs the task requested by the medical practitioner, while it has full-time access to the client's health information from the PHR system. A physician can ask the vNurse to make a report indicating a patient's health status or giving the appropriate warnings, recommendations, and reminders to the patient in defined situations. Figure 1 represents a high-level view of this system.

The VRN system reduces the healthcare costs by lowering the number of visiting patients and avoiding unnecessary hospitalization. As the result, the patient wait time will decrease. On the other hand, the virtual nurse provides information about the patient's health condition to the medical practitioners at a time and with a proper content that could save the time. Moreover, since medical practitioners have access to patient's health information, patients will be diagnosed faster and their treatment will start sooner. The prototype system is currently operating with short messages which are suitable to be applied on smart phone devices. Therefore, patients can always carry their nurses if they install the virtual nurse on their cellphones.

The remaining part of this paper is organized as follows. Section 2 proposes the virtual remote nursing system. Section 3 describes the developed prototype system. Section 4 discusses the case study. Section 5 elaborates on the related work, and finally, Section 6 concludes the paper.

2. VIRTUAL REMOTE NURSING

The proposed Virtual Remote Nursing (VRN) system introduces the virtual Nurse (vNurse) that is a generic software agent installed on the client's machine which would be either a mobile device or a personal computer. A medical practitioner uses a healthcare provider system to define tasks for vNurse based on a generic task definition approach. Each task can be considered as a function that is applied on the client's PHR and context to provide medical information for the practitioner or client. vNurse has access to the client's PHR information (through a PHR system) as well as the client's context, such as current location and time (through local applications), and performs the assigned tasks . In this section, we propose a generic task definition approach as well as the required architecture for the VRN system.

2.1 Generic Task Definition

Each medical guideline represents a pattern in the healthcare domain that corresponds to a decision or an action. An actual nurse is asked to look for these patterns and perform the corresponding actions. This means that every assigned task to the nurse can be modeled by a sequence of guideline steps, called medical workflow. Therefore, we propose to define a task for vNurse with the following components.

$$Task =< Schedule, Model, Knowledge, Data >$$

- *Task Schedule* determines the execution time for the task.
- *Task Model* specifies different steps of a task by a medical workflow.
- *Task Knowledge* provides the corresponding medical guidelines for each step of the medical workflow.
- *Task Data* is the data that is consumed by the provided medical guidelines during the task execution. This data is provided by different sources such as PHR system, medical practitioner, and local applications.

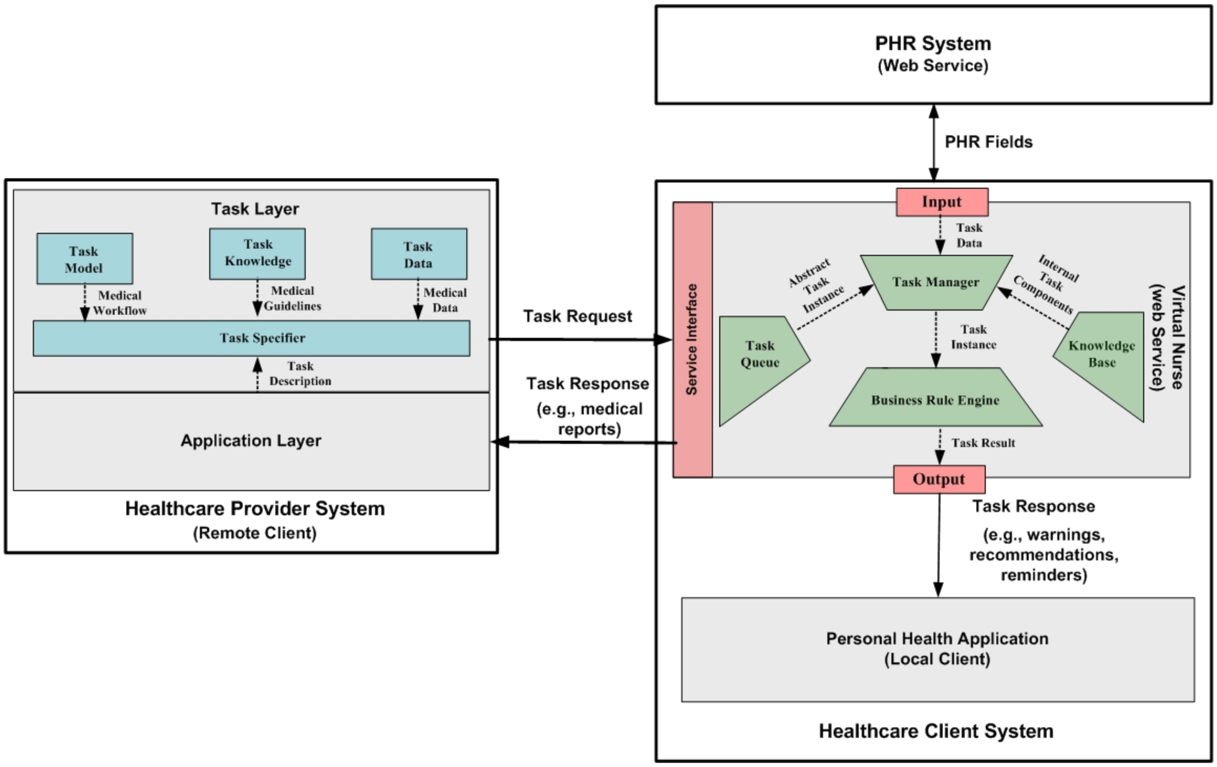


Figure 2: Proposed architecture. Each medical practitioner uses the healthcare provider system to define a task request message that will be executed by the virtual nurse (vNurse).

As a simple example, a physician assigns a task to vNurse to report the general health status of a patient every week (i.e., task schedule). To perform this task, patient's vNurse needs to follow a medical workflow consisting of two steps: first, checking the patient's blood pressure and second, checking the patient's body temperature (i.e., task model). Each step corresponds to a set of medical guidelines that define the patient's status for each range of blood pressure or body temperature, which can be low, normal, or high (i.e., task knowledge). Finally, the obtained information should be reported with a specific medical format (i.e., task data).

2.2 Architecture

In order to allow a medical practitioner to define a task to be performed by the vNurse, an architecture is provided in Figure 2 which consists of the following components.

Personal Health Record System (PHR). This system (e.g., Microsoft Health Vault or Google Health) collects authentic health information from the patients. A growing number of health devices such as heartbeat rate monitor, glucometer, and blood pressure monitor can upload their collected data directly into the PHR systems. Moreover, the vNurse uses the PHR Application Programming Interfaces (APIs) to obtain its client health information.

Healthcare Provider System. The application layer represents a typical Electronic Medical Record (EMR) system that enables a medical practitioner to access to patient's medical information and profiles. Based on this information, the practitioner can describe a task for the virtual nurse to control the patient's status during the treatment. The task description is sent to the task layer, where the task specifier retrieves the required task components from the corresponding modules to formally define the described task. Moreover, the application layer can receive and display task results from vNurses, such as patient's health reports.

Personal Health Application. This component represents a typical personal health application that is enhanced to receive task results generated by vNurse. Task results are given to this application in the form of messages including medical recommendations, warnings, reminders, etc. This application can simply display the messages or use other Human Computer Interaction (HCI) approaches to inform the client.

Virtual Nurse. The proposed virtual nurse is modeled as a software agent and wrapped into a web service to be called by different healthcare provider systems. Thanks to mobile devices and their application platforms, even smart phones can become a server and provide web services. Therefore, vNurse has a service interface that takes a task request and sends the task response to the healthcare provider (remote client) or healthcare client (local client). In order to apply medical guidelines, vNurse considers each medical guideline as a business rule, including condition and action parts. Similarly, vNurse considers medical workflows as business pro-

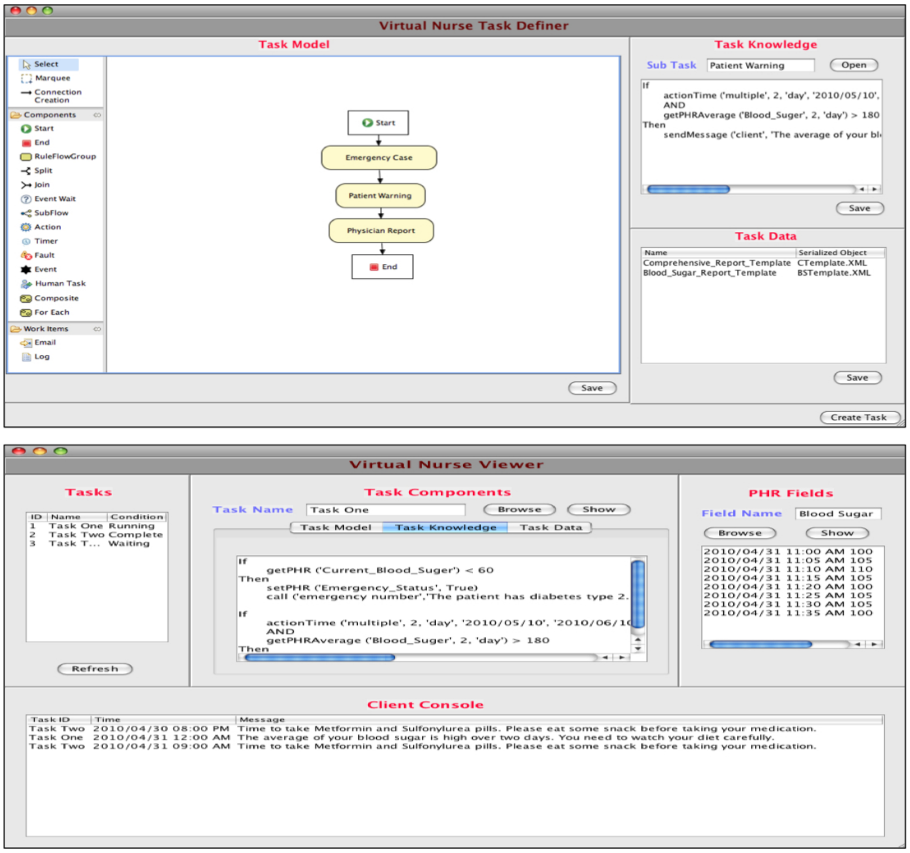


Figure 3: Snapshots of the prototype Virtual Nurse Task Definer (top) and Virtual Nurse Viewer (bottom) Version 1.0 that are running on the case study described in Section 5.

cess models and task data as business objects. The agent components are discussed below.

- *Input* receives client's health information from the PHR system.
- *Output* sends task results (e.g., recommendations, warnings, reminders, etc) to the personal health application to be delivered to the client.
- *Task Queue* stores tasks that are waiting to be selected for execution. Initially, tasks are stored into the task queue based on First-In First-Out (FIFO) strategy and their priority.
- *Knowledge Base* stores received task knowledge to relieve the service caller from sending them each time.
- *Business Rule Engine* executes a task instance, by applying business rules and performing business actions on the business objects.
- *Task Manager* controls the entire life cycle of a task instance that is divided into three phases as follows.
  1. *Selection phase*: selects a task for execution from the task queue based on it's order and schedule.
  2. *Instantiation phase*: i) creates an abstract business process based on the received task components; and ii) instantiates a task instance from the abstract business process using relevant PHR fields and internal functions stored into the knowledge base.
  3. *Execution phase*: passes the task instance to the business rule engine to be executed. The task response is sent to the client (via output) or healthcare providers (via interface).

3. PROTOTYPE SYSTEM

To evaluate the effectiveness and feasibility of our model, we developed a prototype of the proposed virtual remote nursing system, including the vNurse agent and the required APIs to design task requests and receive task results. This prototype, namely *VRN version 1.0*, is implemented based on J2EE 1.5 technologies and Apache Tomcat 6.0 application server. vNurse v1.0 uses Drool v5.0 business rule engine to run business processes, apply business rules, and execute business actions. Moreover, vNurse v1.0 connects with Google Health as its PHR system by calling Google APIs.

Each business rule is defined by conditions and actions: *conditions* are expressed by First Order Logic and *actions* are defined by Java statements or functions. Business process

Task Model

Task Knowledge

Emergency case	Condition I	getPHR ("Current_Blood_Sugar") < 60
	Action I	setPHR ("Emergency_Status", True); call ("911", "The patient has diabetes type 2. There is a risk of hypoglycemia. Her address is" + getPersonalInfo ("Current_Location"));
Patient Warning	Condition I	actionTime ("multiple", 2, "day", "2010/05/10", "2010/06/10", "12:00 AM") = True AND getPHRAverage ("Blood_Sugar", 2, "day") > 180
	Action I	sendMessage (getPersonalInfo ("Email_Address"), "The average of your blood sugar gone high over last two days. You need to watch your diet carefully");
Physician Report	Condition I	actionTime ("single", "2010/06/10", "6:00 PM") = True
	Action I	String Report = makeReport (Comprehensive_Report_Template); sendMessage ("physician @email.com", Report);
	Condition II	actionTime ("multiple", 1, "week", "2010/05/10", "2010/06/10", "12:00 AM") = True
	Action II	String Report = makeReport (Blood_Sugar_Report_Template); sendMessage ("physician @email.com", Report); sendMessage (getPersonalInfo ("Email_Address"), "The average of your blood sugar gone high over last week. You need to consult with your physician");

Task Data

Comprehensive_Report_Template (provided by physician)
Blood_Sugar_Report_Template (provided by physician)
Patient's PHR Information (provided by PHR system)
Patient's Personal Information (provided by local applications)

Figure 4: Task One. Sherry's physician assigns this task to vNurse to control her blood sugar level continuously.

models are converted to XML format, business rules are encoded by PMML standard version 3[15], and business objects are serialized to form the task request message. VRN v1.0 is provided by two Java packages, namely the *TaskDefiner* and the *vNurse*, used as follows.

- **TaskDefiner** package: provides graphical APIs and widgets for a medical practitioner to define tasks and send them to a virtual nurse.
- **vNurse** package: provides APIs to generate virtual nurse and connect it with local and remote clients.

Based on these two packages, we also developed vNurse Task Definer v1.0 (Figure 3 (top)) to generate task request messages and vNurse Viewer v1.0 (Figure 3 (bottom)), to control the virtual nurse execution. Finally, it should be mentioned that different healthcare providers and client systems can be compatible with VRN system using the provided APIs.

4. CASE STUDY

In this section, we demonstrate applications of the Virtual Remote Nursing system through a case study that demonstrates the involvement of three healthcare providers in using the VRN v1.0 to assign their tasks to the vNurse.

4.1 Scenario

Sherry is an elderly woman with type 2 diabetes. She has a vNurse installed on her smart phone. She also uses a blood glucose sensor that collects her blood sugar level continuously and sends them to her account in a PHR system, such as Google health.

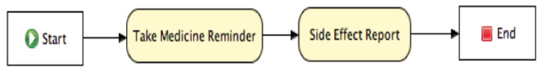
During her last visit the physician prescribes her Metformin and the Glimepiride and also advises her to go on a diet. The physician wants to ensure that Sherry stays on her diet until her next regular monthly appointment. Meanwhile, the physician is interested to know how well the prescribed medications work during this period. However, if Sherry's blood sugar level stays high even for a week, she will need to be visited sooner than her scheduled appointment. As with other diabetic patients taking medications, Sherry is at the risk of hypoglycemia (i.e., low blood sugar level) that could result in losing consciousness. Consequently, the doctor wants to make sure she receives emergency medical services at the earliest time, in the case of extreme hypoglycemia.

Sherry visits a pharmacy to obtain her prescribed medications. The pharmacist wants to make sure Sherry does not forget the right dosage, time, and frequency of her medication that are prescribed by her doctor. Moreover, since the prescribed medications could cause side effects, he wants to have Sherry's doctor informed if the medications cause any complications. Finally, in case of an emergency situation, the emergency center should obtain the latest information about Sherry's health condition.

4.2 Virtual Nurse Assigned Tasks

Sherry's physician, pharmacist, and the emergency center define the following tasks for the Sherry's vNurse to fulfil

Task Model



Task Knowledge

Take Medicine Reminder	Condition I	actionTime ("multiple", 1, "day", "2010/05/10", "2010/06/10", "9:00 AM") = True OR actionTime ("multiple", 1, "day", "2010/05/10", "2010/06/10", "8:00 PM") = True
	Action I	sendMessage (getPersonalInfo ("Email_Address"), "Time to take Metformin (500 mg) and Sulfonylurea (5 mg) pills. Please eat some snack before taking your medication");
Side Effect Report	Condition I	containPHR (Current_Status, "Diarrhea") = True
	Action I	sendMessage (getPersonalInfo ("Physician_Email_Address"), "Your patient" + getPersonalInfo ("Name") + "has Diarrhea after taking Metformin");

Task Data

Patient's PHR Information (provided by PHR system)
Patient's Personal Information (provided by local applications)

Figure 5: Task Two. Sherry's pharmacist assigns this task to vNurse to remind her taking her medications and reports the side effects to her physician.

their concerns about her.

Task One. The physician assigns a task to Sherry's vNurse to control her blood sugar level. This task is scheduled to be executed every time Sherry's blood sugar level is updated in the PHR system. This task, as shown in Figure 4, defines the following responsibilities for vNurse.

- In the case of hypoglycemia, vNurse contacts the emergency center.
- Every two days, vNurse checks the average of her blood sugar level. If it is high, Sherry will receive a warning to watch her diet seriously.
- Every week, vNurse checks the average of her blood sugar level. If it is high, vNurse makes a report about Sherry's blood sugar status over the last week and sends it to the physician. Moreover, Sherry will be advised to visit her physician.
- In two weeks, vNurse makes a comprehensive report about Sherry's health condition and sends it to her physician.

Task Two. The pharmacist assigns a task to Sherry's vNurse to handle the medication-related issues. This task, as shown in Figure 5, is scheduled to be executed every hour and defined as follows.

- At medication times, vNurse reminds Sherry to take her medications.
- If these medications cause any side-effects vNurse informs Sherry's physician about it.

Task Three. In a case of emergency, Sherry's vNurse contacts emergency center. Then, the emergency center assigns the following task, as shown in Figure 6, with a high-priority (i.e., immediate execution) to the vNurse.

- If the emergency situation is confirmed, the vNurse calls Sherry's emergency contact to let her/him know about Sherry's condition.
- vNurse sends the required information about Sherry to the emergency center in order to provide better and faster medical services for Sherry when they arrive.
- If Sherry's health condition changes, her vNurse informs the emergency center immediately.

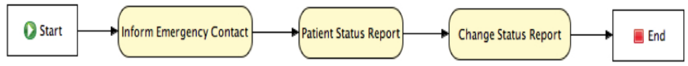
In order to perform these tasks, the virtual nurse must call some functions that are stored in the internal knowledge base of vNurse. These are presented in Table 1.

5. RELATED WORK

User-centric Internet introduces Medicine 2.0 [9] as medical services that works based on the clients' health information maintained by on-line PHR systems. Moreover, Web 2.0 technologies, semantic web, and virtual reality tools facilitate social networking and collaboration between patients and physicians. Also, integration of Medicine 2.0 with health monitoring system [12], telemedicine [20], and sensor network [3] provides more beneficial services. For example, an architecture for monitoring body motion and heart activities through sensor networks is proposed in [14] where the collected data can be sent to the PHR systems. Similarly, [10] introduces a health monitoring system including intelligent reporting and alerts. In this paper, we propose a healthcare monitoring system that is controllable by medical practitioners.

Since the demand for controlling the patient's health is on the rise and the number of health experts is limited, the IT researchers are pursuing the simulation of some of the health practitioners' activities [18]. For instance, Clinical

Task Model



Task Knowledge

Inform Emergency Contact	Condition I	getPHR ("Emergency_Status") = True
	Action I	call (getPersonalInfo ("Emergency_Contact"), getPersonalInfo ("Name") + "is in emergency status. She is at" + getPersonalInfo ("Current_Location"));
Patient Status Report	Condition I	getPHR ("Emergency_Status") = True
	Action I	String Report = "Patient Blood Pressure=" + getPHR ("Current_Blood_Pressure") + "Patient Blood Tempeture =" + getPHR ("Current_Blood_Tempeture") + "Patient Blood Sugar =" + getPHR ("Current_Blood_Sugar") + "Patinet Allergies =" + getPHR ("Allergies"); sendMessage ("emergency @ email.com", Report);
Status Change Report	Condition I	changePHR("Blood_Sugar") = True
	Action I	String Report = "Patient's Blood Sugar is changed to" + getPHR ("Current_Blood_Sugar");
	Condition II	changePHR("Blood_Pressure") = True
	Action II	String Report = "Patient's Blood Pressure is changed to" + getPHR ("Current_Blood_Pressure");
	Condition III	changePHR("Body_Tempeture") = True
	Action III	String Report = "Patient's Body Tempeture is changed to" + getPHR ("Current_Body_Tempeture");

Task Data

Patient's PHR Information (provided by PHR system)
Patient's Personal Information (provided by local applications)

Figure 6: Task Three. Emergency center assigns this task to vNurse to receive latest information about Sherry's health condition in the case of hypoglycemia.

Decision Support Systems (CDSS) [16] and Electronic Medical Record Systems (EMRs) [1] are introduced. CDSS could be used as a decision maker for assisting physician in diagnosis of diseases like heart failure [19] or for administrating tasks like elective surgery scheduling [8]. EMR could be used as an administrator for creating and maintaining the patients data electronically. Although nursing plays an important role in healthcare domain, its roles have not been modeled efficiently. Only a few approaches have been proposed to model nurse activities: [7] uses patient record modifications; and [2] uses a concept-oriented approach. In this paper, we propose a task-oriented approach to provide an efficient model for nursing.

In the proposed architecture, we model a nurse with a generic software agent. Software agents have been integrated into the healthcare domain such as [4] [11] . Traditional approaches can model a nurse as a mobile agent [6] that can physically travel across a network and perform tasks on different nodes. This method causes several security and privacy vulnerabilities, especially in healthcare. This unsafe mechanism has been resolved in VRN by sending task and knowledge to vNurse, a resident generic agent. Besides, mobile agent architectures based on classic methods (e.g. Mole [5]) suffer from low-efficiency, as they need to send the entire computer program and process. In our architecture, we solved this problem by sending short task messages.

Finally, the proposed architecture extends our previous work in [17] and [13]. The former, introduces a framework for data and knowledge interoperability where it enables knowledge (i.e., medical guideline) to be transferred in association with data (i.e., patient EMR); the later introduces a generic software agent that resides at the client side and customizes web service responses based on client's context.

6. CONCLUSION AND FUTURE WORK

Healthcare domain will embrace smart Internet and mobile technology in near future. Smart Internet assists individuals in accessing to PHR systems; and mobile technology provides inexpensive devices that can efficiently collect medical or contextual information from their clients. Our proposed approach takes advantages of these technologies to present a virtual nurse which enables healthcare professional, as the main decision makers, to control their patients' treatments and conditions. In this paper, we presented new methods that a healthcare professional can define a specific task for a patient's virtual nurse based on the medical workflows, guidelines, and data. The virtual nurse then performs the specified task and returns the task results to the healthcare professional or the client. We are currently working on modeling interactive virtual nurses where an assigned task can be accomplished by the collaboration of the patient and the virtual nurse. Finally, a healthcare provider will be able to assign some simple decision making tasks to the virtual nurse.

7. REFERENCES

[1] R. Agrawal, G. Tyrone, C.Johnson, and J.Kiernan. Enabling the 21st century healthcare information technology revolution. *Commun. ACM*, 50(2):34–42, 2007.

[2] A.Kumar, B. Smith, D. Pisanelli, A. Gangemi, and M. Stefanelli. An ontological framework for the implementation of clinical guidelines in health care organizations. *Studies in Health Technology and Informatics*, (102):95–107, 2004.

Table 1: Virtual Nurse functions stored in the internal knowledge base.

Name	Parameters	Description
getPHR()	PHR Field	Retrieves the value of the PHR field from the PHR system.
getAveragePHR()	PHR Field Value	Assigns the value to the PHR field in the PHR system.
setPHR()	PHR Field Period	Takes average of the PHR field over the defined period.
changePHR()	PHR Field	Returns true if the PHR field just changed more than its corresponding threshold.
getPersonalInfo()	Field	Returns the requested information about the client.
call()	Phone Number Message	Makes a call to the defined phone number and tells the message. It uses a text2Speach function to convert the message into a voice message.
sendMessage()	Email Address Message	Emails the message to the defined email address.
actionTime()	Frequency Period Start Date End Date Time	Returns true if the current time mateches with the execution time of the action. For example, actionTime (”multiple”, 2, ”day”, Start Date, End Date, Time) says the execution time for the next action is every two days from Strat Date to End Date and at the Time.
makeReport()	Report Template	Makes a report based on the recived template that specifies the required PHR information in a proper format.

[3] F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E.Cayirci. A survey on sensor networkd. *IEEE Communication Magazine*, 40(8):102–114, 2002.

[4] A.Moreno and J.Nealon. *Applications of Software Agent Technology in the Health Care Domain.* Birkhauser (Architectural), 2004.

[5] J. Baumann, F. Hohl, K. Rothermel, M. Schwehm, and M. Strasser. Mole 3.0: a middleware for java-based mobile software agents. In *The International Conference on Distributed Systems Platforms and Open Distributed Processing*, pages 355–370, London, UK, 2009. Springer-Verlag.

[6] P. Braun and W. Rossak. *Mobile Agents: Basic Concepts, Mobility Models, and the Tracy Toolkit.* Morgan Kaufmann Publishers Inc., San Francisco, USA, 2004.

[7] E.Goorman and M.Berg. Modelling nursing activities: Electronic patient records and their discontents. *Nursing Inquiry*, 7(1):3–9, 2000.

[8] J. Everett. A decision support simulation model for the managment of an elective surgery waiting system. *Health Care Manage Science*, 5(2):89–95, 2002.

[9] G. Eysenbach. Medicine 2.0: social networking, collaboration, participation, apomediation, and openness. *Journal of Medical Internet Research*, 10(3), 2008.

[10] H. Garsden, J. Basilakis, B. G. Celler, K. Huynh, N. H. Lovell, and S. Member. A home health monitoring system including intelligent reporting and alerts. pages 3151, 3155, San Francisco, CA, USA, 2004.

[11] K.Haigh, J.Phelps, and C.Geib. An open agent architecture for assisting elder independence. In *The Proceedings of the first international joint conference on Autonomous agents and multiagent systems*, pages 578–586, New York, NY, USA, 2002. ACM.

[12] I. KORHONEN. Health monitoring in the home of the future. *IEEE Engineering in Medicine and Biology*, 22(3):66–73, 2003.

[13] M. Najafi and K.Sartipi. A Framework for Context-Aware Services Using Service Customizer. In *The IEEE International Conference On Advanced Communication Technology.*, volume 2, pages 1339–1344, Phoenix Park, Korea, 2010.

[14] C. Otto, A. Milenkovis, C. Sanders, and E. Jovanov. System architecture of a warieless body area sensor network for ubiquitous health monitoring. *ACM Trans. Program. Lang. Syst.Journal of Mobile Multimedia*, 1(4):307–326, January 2006.

[15] S. Raspl. PMML Version 3.0 - Overview and Status. In *The ACM Workshop on Data Mining Standards, Services and Platforms*, pages 18–22, Philadelphia, USA, 2004.

[16] R.Greenes. *Clinical Decision Support: The Road Ahead.* Academic Press, Inc., Orlando, USA, 2006.

[17] R.Kazemzadeh and K.Sartipi. Interoperability of data and knowledge in distributed health care systems. In *The IEEE International Workshop on Software Technology and Engineering Practice*, pages 230–240, Washington, USA, 2005. IEEE Computer Society.

[18] R.Lenz and M. Reichert. It support for healthcare processes - premises, challenges, perspectives. *Data and Knowledge Engineering*, 61(1):39 – 58, 2007.

[19] S.Leslie, M. Hartswood, C.Meurig, S.McKee, R. Slack, R. Procter, and M.Denvir. Clinical decision support software for management of chronic heart failure: Development and evaluation. *Computers in Biology and Medicine*, 36(5):495 – 506, 2006.

[20] R. Wootton, J. Craig, and V. Patterson. *Introduction to telemedicine.* Royal Society of Medicine, 2006.