AN AGENT BASED VIRTUAL MEDICAL DEVICES

DUSAN STARCEVIC
ZELJKO OBRENOVIC
The University of Belgrade, Belgrade, Serbia and Montenegro

EMIL JOVANOV
The University of Alabama, 213 EB Huntsville, U.S.A.

Abstract
In this paper we present the telemedical environment based on VMDs implemented with Java mobile agent technology, called aglets. The agent based VMD implementation provides ad-hoc agent interaction, support for mobile agents and different user interface components in the telemedical system. We have developed a VMD agent framework with four types of agents: data agents, processing agents, presentation agents, and monitoring agents. Data agents abstract data source, creating uniform view on different types of data, independent of data acquisition device. Processing agents produce derived data, such us FFT power spectrum, from raw data provided by the data agents. Presentation agents supply user interface components using a variety of user data views. User interface components are based on HTTP, SMS and WAP protocols. Monitoring agents collaborate with data and processing agents providing support for data mining operations, and search for relevant patterns. We have applied VMDs to facilitate distributed EEG analysis.

Keywords: Software agents, telemedicine, distributed systems, perceptual user interfaces

1. INTRODUCTION

One of the main factors that define quality of life is the quality of health care, i.e. quality and availability of hospitals, doctors, dentists and emergency services. To achieve high quality medical care it is necessary to provide prompt, accurate recording, communication, and retrieval of patient data and medical logistics information. Beyond ordinary usage, such information are extremely useful in emergency medicine (trauma care, epileptic attack, heart attack, stroke, etc.), where sometimes they make the difference between life and death. In addition, information availability allows for better planning and scheduling of medical resources.

In conventional metaphor of health care services physical presence and collaboration of physicians, medical stuff, and patient was prerequisite. However, globalization and higher people mobility (business, tourism, etc.), leads to fragmented care delivered at scattered locations. A related problem implicated our highly mobile workforce. Upon changing jobs, people found out that previously existing conditions were not covered on the new health plan. As a consequence, common limitations of healthcare include:

- Lack of patient medical record (PMR);
- Absence of qualified physician;
- Required specialist team;
- Lack of sophisticated medical equipment.

Telemedicine is often viewed as a solution for these new demands on health care. Increased performance of computers, telecommunications and information infrastructures establish a basis for new types of telemedical applications [3]. In addition to improved health care by telepresence, telemedicine can create a new quality by creating virtual collaborative environments, independently of the participants' physical presence. For example, telemedicine provides a way for specialists to electronically travel to under-served regions, thus increasing the size of coverage without actually traveling to these areas. Furthermore, telemedicine can make possible providing health care to patients who either will not or cannot travel to seek specialty care.

Additional factor that drives the evolution of telemedicine is the cost of public and private health care. Historically, the medical business has been a fee-for-service institution. One of the negative results of this model was the spiral increase in associated costs, which increased the problem of paying the bill. To address these problems, health care model is moving toward managed care. Although telemedicine is not necessarily cost-effective in the conventional fee-for-service setting, it has potential to fit well inside the managed-care model. On the other hand, hospitals are no longer able to stand as the independent service they previously were. Small rural hospitals may not continue to exist unless they align themselves with larger tertiary care centers in a telemedicine network, or some other type of partnership. Telemedicine partnership has great potential by helping to keep patients in their local community [4].

In this paper, we present the telemedical environment based on agent technologies. We apply a novel concept of Virtual Medical Devices (VMD), which we have implemented using agents [11]. This VMD implementation is a distributed agent platform, a decentralized system for building telemedical applications by networking local system resources such as various medical devices, medical sensors or medical information systems. We have found that the flexibility of the distributed agent architecture is very well suited for telemedical application domain.

The remaining sections of this paper are organized as follows. Section 2 presents the needed background material and existing solutions. This includes a short overview of the evolution of telemedicine and especially Web based telemedical systems. In Section 3, we present the essence of the proposed solution. This includes the description of architecture with more detail description of the VMD concept, and the discussion of potential benefits of using agents. Proposed agent based VMD framework is presented in

Section 4. In Section 5, we outline implementation details of telemedical EEG application developed within our framework. First experiences and lessons learned are discussed in Section 6. Section 7 concludes the paper.

2. BACKGROUND AND EXISTING SOLUTIONS

In this chapter we describe existing telemedical and web-based telemedical systems.

2.1 TELEMEDICINE

Telemedicine can be broadly defined as the use of telecommunications technologies to provide medical information and services. It involves the practice of delivering health care over a distance using telecommunications equipment as simple as telephones and fax machines or as complex as PCs and full-motion interactive multimedia. Telemedicine uses a combination of some existing technologies from other fields, but it comes with its own additional set of challenges because of the highly complex world of medicine.

From functional view, telemedicine consists of both real-time interactive consults, and batch processing of patient data and information for diagnosis at a later time. Also, due to its distributed nature, telemedicine applications are often used for continuing medical education and patient education in some forms of distance learning. Telemedicine involves the electronic convergence of medical information for the purpose of diagnosis and treatment of patient using PC, telecommunications links, specialized video, audio, and imaging equipment. For instance, *teleEEG* that is developed at University of Calabria, gives the possibility to manage local patient database, including EEG data files. It was based on point-to-point communication using a standard telephone line [10].

There is an explosion of activity and competitors among provides today with regard to cost, quality, and access to care on a regional basis. However, most of these active programs use proprietary and expensive telecommunications networks. As result, reimbursement and the cost of telecommunications were identified as the most important barriers by the 72 active programs in the U.S. [4].

2.2 WEB BASED TELEMEDICINE

Telemedicine significantly changes storage, retrieval, and use of biomedical information in medical information systems. Evolution of medical information systems is greatly influenced by capabilities and price/performance ratio of novel information technology. The Internet and World Wide Web (WWW) as global information infrastructure offer low cost environment for telemedical applications [1]. Contemporary Web infrastructure provides lots of services including middleware, content description languages, directory services, and security frameworks. It seems that at the present state of technology, Web based medical applications represent natural way of creating interactive collaborative environment - Virtual Medical World (VMW).

Web as a platform for telemedicine extremely decreases reimbursement and the cost of telecommunications. Yet, the main problem of Web based telemedicine systems is quality of services (QoS). Although some of the functions, such as recording, communication, and retrieval of patient medical records can be efficiently realized on the

Web, functions with real-time requirements can hardly be implemented in this environment using conventional client server or n-tier paradigm.

3. THE ARCHITECTURE

Telemedical systems work in heterogeneous processing environment and require a flexible system reconfiguration. Physician may switch from a personal digital assistant to a high performance – high resolution workstation very often, expecting the telemedical system to reconfigure and provide the support within seconds. In order to actually realize complex telemedical services an adequate technological approach is required.

Typical scenario that illustrates some of the main points of the problem of building telemedical applications includes acquisition, processing, distribution and presentation of data. Data acquisition can be performed on patient side. Captured data are transferred and archived in patient medical record. Physicians examine data either on-line, in real-time during recording, or off-line from archive. In addition, physicians can add their findings to the patient medical record.

Agent-oriented approaches are well suited for developing complex, distributed systems Analyzing, designing, and implementing complex software systems as a multiagent system affords software engineers a number of considerable advantages over existing methods [2,5,13]. Agents can react dynamically to unfortunate situations, increasing robustness and fault-tolerance, aspects that are very important in telemedicine. Possibility of agent's execution in asynchronous and autonomous way is of great importance in places with fragile network connections, and for mobile devices.

In this section, we will present architecture of telemedical applications realized as a multiagent system. We have built an agent-based framework for development of rather complex telemedical applications. We use 4+1 view model of architecture to outline architecture description of our solution. However, in this paper to describe novel architecture a telemedical application using Virtual Medical Devices, we will present primarily logical view.

In 4+1 view model of architecture, the logical view primarily supports the functional requirements, and describes the services the system should provide to its end users. In this view designers usually decompose the system into a set of key abstractions, taken mainly from the problem domain. The logical view aids functional analysis and decomposition, since it identifies mechanisms and design elements that are common across the system.

We propose expressing the functional view of telemedical applications using a set of Virtual Medical Devices (VMD). A Virtual Medical Device is a novel conceptual model of medical services. Virtual Medical Devices describe telemedical services using metaphor of conventional health care service. This metaphor includes physical presence of a patient, physicians, medical stuff and patient medical records (Figure 3.1).

VMDs provide specific views on telemedical services. Using VMDs we are able to separate the specification of some medical service from its implementations.
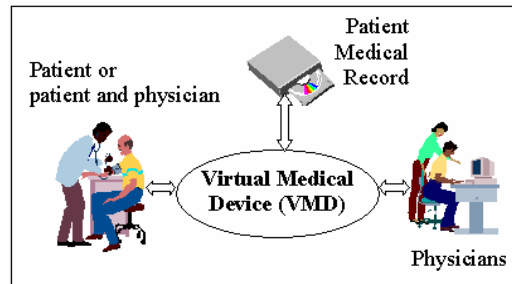
Figure 3.1 Metaphor of conventional health care service represented with VMD

This separation is often absent in conventional medical systems where the specification is frequently hardwired with implementation. Medical service can be defined as a set of functions that some device provides. A device can be software, hardware, a medical unit or some combination of these three elements.

We have decided to realize description of VMDs using UML and XML technologies. According to this, our concept of a VMD can be viewed as a domain model in the Model Driven Architecture (MDA) that is recently proposed by Object Management Group (OMG) [9]. MDA uses UML and XML Metadata Interchange (XMI) to describe platform independent elements of application.

Having service description in XML allows sharing of the logical view among various platforms. It is possible to create libraries of VMDs, improving interoperability among developers.

4. AN AGENT BASED FRAMEWORK FOR VMDs

In this section, we will present details of the implementation of VMDs based on the Java mobile agent technology. Java has become de facto standard for Java mobile agents, and there exist a great number of Java based agent platforms that could be used as a basis for our solution [12]. We have decided to use Aglet Framework, since it is highly accepted and lots of interesting projects have used this framework. Choosing Aglets we were able to reuse some of experiences from these projects.

Our agent based framework offers mechanisms for building telemedical applications by networking local system resources such as various medical devices, medical sensors or medical information systems. This framework provides ad-hoc agent interaction, mobile agents, and different user interface components to telemedical system.

Our framework is well suited as a platform for development of telemedical applications specified with VMDs. A VMD can be viewed as an interface to multiagent organizations that implements declared telemedical services. The framework is comprised of a set of distributed agents. In addition to agents, the framework contains a VMD applet as an interface to World Wide Web. In remaining sections, we will describe agents and the applet.
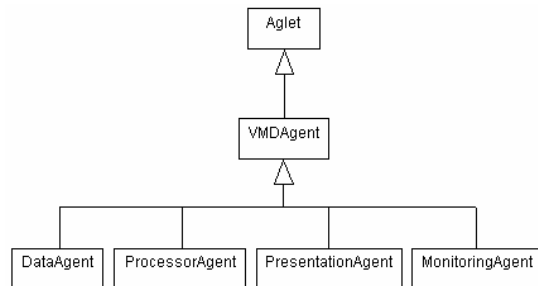
Figure 4.1   Taxonomy of agents in proposed framework

4.1 AGENTS

Our framework is comprised of a set of Java agents that cooperate to efficiently implement some telemedical service. Having in mind typical scenario of telemedical services, we have developed a VMD agent framework with four types of agents: data agents, processing agents, presentation agents and monitoring agents. All agents in the framework are derived from VMDAgent class (Figure 4.1). This class is the extension of Aglet class, and defines some common structures and mechanisms for all agents in our framework. For instance, these mechanisms include support for XML based definition of VMD services and a XML based agent communication. Aglet class is a part of the Aglet Framework. Aglet framework requires that every agent be a subclass of Aglet class. Each new telemedical application consists of agents and applets that are derived from our framework.

Every aglet can be dispatched to any remote host that supports the Java Virtual Machine. Accordingly, this requires from the remote host to preinstall a tiny aglet server program, called Tahiti. Tahiti server captures aglets and provides them with an aglet context. Within its context, an aglet can be cloned or disposed. Also, an aglet can communicate with other aglets, collect local data, halt its execution and be dispatched to another host.

4.2 DATA AGENTS

Data agents abstract data source, creating uniform view on different types of data, independent of data acquisition device. Data can be fetched online from a data acquisition board or offline from a file (Figure 4.2).
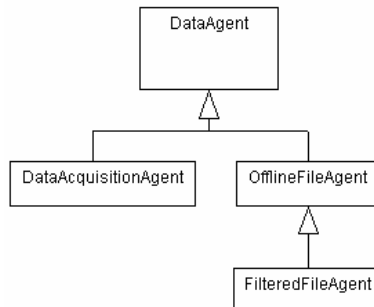

Figure 4.2 Data agents

These agents are primarily stationary, since they are connected with some hardware device or with a file. In order to access some hardware device, agents have to use some device driver. This usage is usually implemented via Java Native Interface (JNI) mechanism. Offline file agents simply read data from the file that is created by some other data acquisition process. They also can access database using Java Database Connectivity (JDBC) interface. To reduce network load, data can be first filtered, and then transmitted to some other agent. Data agents provide other agents with acquisitioned data in XML format.
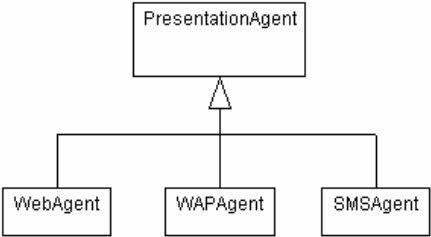
4.3 PROCESSING AGENTS

Processing agents produce derived data from data fetched by data agents. For example, these agents can produce power spectrum calculating Fast Fourier Transformations (FFT), or calculate values doing Independent Component Analysis (ICA) (Figure 4.3).

Usually, these agents collaborate with data agents from which they receive low-level data in XML format. Then they calculate some derived data, and transmit it to other agents in XML format, too. These derived data can be further processed or presented via a user interface. Also, processing agents can collaborate with each other to produce some more complex data. In this case one of the agents usually plays the role of a coordinator, receiving and correlating data from different agents.
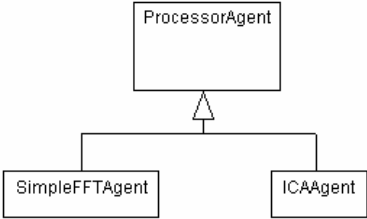
Figure 4.3 Processing agents

Figure 4.4 Presentation agents

Processing agents can be distributed on different nodes in a network to balance network load or to optimize processors workload. For instance, an agent can change a host if the host becomes overloaded choosing less loaded one. On the other hand, processing agents can travel to data agent host and process data locally, reducing network load.

4.4 PRESENTATION AGENTS

Presentation agents supply user interface components with data in the format that interfaces can interpret. These agents collaborate with processing and data agents from which they receive data in XML format. The data in XML format cannot directly be presented on various user interfaces, but need to be transformed into a form suitable for presentation.

In order to support implementation of telemedical applications in the Internet and cellular networks environments, we have developed three appropriated presentation agents: WebAgent, WAPAgent and SMSAgent (Figure 4.4).

A Web agent supplies Java applets or some other Web presentation components with the data in the proper format. For example, this agent can transform XML data into a standard HTML format. A WAP agent transforms data into a Wireless Markup Language (WML) format suitable for presentation on Wireless Application Protocol (WAP) enabled cellular phones. A SMS agent transforms data into a sequence of SMS messages, which most of cellular phones can send and receive. WAP and SMS agents usually implement their functionality via WAP and SMS gateways. Communication between agents and gateways goes over HTTP protocol.

4.5 MONITORING AGENTS

Monitoring agents collaborate with data and processing agents providing support for data mining operations, and search for relevant patterns. For example, data agents can automatically collects patient vital signs such as EEG signals, blood pressure, pulse, respiratory rate, oxygen saturation. A monitoring agent continually processes these signs, and upon finding some pattern, the agent can transmit emergency signal augmented with vital signs to some emergency center. Typical example is monitoring for possible epileptic attacks.

A monitoring agent can send emergency signal to SMS presentation agent. SMS presentation agent sends a SMS message to a physician through the cellular network. After that, the monitoring agent can continuously send vital signs to Web presentation agent, that physician can use to remotely examine a patient using Web browser.

4.6 VMD APPLET

We have developed an abstract class, called VMD applet, to control agent based telemedical applications through a Web interface. This applet provides basic functions for hosting and work with VMD agents. To allow aglets to be maintained within applets, an abstract applet class, called "FijiApplet", is provided as part of the Aglet Framework. This class is placed in a Java package, called "Fiji Kit". The FijiApplet maintains some kind of an aglet context. From within the context, aglets can be created, dispatched, and retracted back to the FijiApplet. The VMD applet extends the FijiApplet.

The Aglet Framework provides two supplementary components for a Java-enabled Web browser to host and fire aglets, which are needed for execution of our VMD applet. These are an aglet plug-in and an aglet router. The aglet plug-in allows the browser to host aglets. Due to Java applet security restrictions, the aglet must first go through the Web

server from where the HTML page was downloaded. At the Web server machine, the aglet router captures incoming aglets and immediately forwards it to its destination.

5. CASE STUDY: MONITORING AND ANALYSIS OF EEG SIGNAL

We have used our VMD agent framework to create telemedical application for monitoring and analysis of EEG signal. We implemented prototype of this agent-based application in Institute for Mental Health in Belgrade.

Data acquisition agent fetches signals from MEDELEC 1A97 EEG machine (MEDILOG BV, Nieuwkoop, The Netherlands) that was connected with PC-based data acquisition workstation via Data Translation DT2801 A/D converter board. We have developed data acquisition driver in C, and connected it with data agents via JNI mechanism. The converter board and the drive convert 16 channels of EEG in real-time with the sampling frequency of 256 Hz and 12-bit precision. Raw EEG data are then distributed to the processing agents. Data acquisition agents can optionally archive raw data in a file in XML format. In order to achieve compatibility with existing applications we implemented filter for EEG data format generated by *RHYTHM 8.0* software (Stellate Systems).

We have implemented several processing agents, including power spectrum computation agents, which use the fastest software FFT processing in PC environment.

We developed several presentation agents as a general purpose EEG analysis user interfaces [6]. In a Web page interface with a VMD applet, brain electrical activity is represented using animated topographic maps projected on 3-D head model. Visualization could be synchronized with data sonification of EEG data [7]. Sonification is implemented by modulation of natural sound patterns to reflect certain features of processed data, and create pleasant acoustic environment. This feature is particularly important for prolonged system use. We have used Virtual Reality Modeling Language (VRML) in our Web page interfaces (Figure 5.1).

The VRML is simple language for describing 3-D shapes and interactive environment. VRML also supports Java language as scripting language, which provides lots of opportunities for integration with Java based agents. In our system, VRML world is controlled by Java applet that communicates with Web VMD presentation agents.
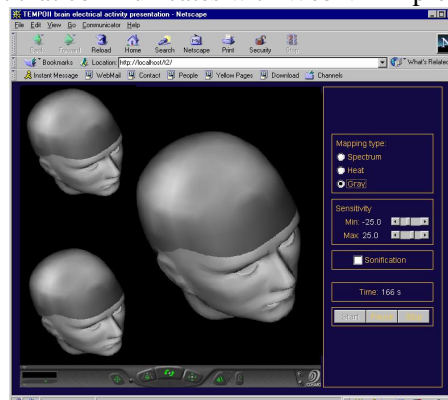

Figure 5.1 A multimodal Web interface for animated topographic map presentation

We have researched VMD Web interface with speech control based on Microsoft Speech API (SAPI) integrated in control applet. Applet has used services of SAPI via Java Native Interface (JNI) mechanism. As a result, it is possible to start and stop presentation of data, and to change mapping type by speech commands.

Also, we have developed simple WAP based user interface, for textual presentation, and static 2D map presentation with Wireless Bitmap (WBMP) format. We have tested our WAP interfaces on Nokia 7110 cellular phone, and on number of WAP simulators. WAP devices receive data from our WAP VMD presentation agents. The SMS interface allows distribution of simple textual messages, and simple queries for medical data. SMS devices receive data from our SMS VMD presentation agents. For SMS and WAP communication, we have used Kannel open source SMS gateway [8]. Presentation agents for WAP and SMS interface communicate with Kannel over HTTP protocol.

We have developed our system for EEG diagnosis and analysis in Laboratory for Multimedia Communications at the Faculty of Organizational Sciences, and we tested it in Institute for Mental Health in Belgrade. We received very good response from physicians in experimental and clinical settings.

We were able to experience that proposed agent based framework can significantly reduce implementation efforts. We have found that the flexibility of distributed agent architecture is well suited for the telemedical application domain. This flexibility is particularly important in the case of an emergency, enabling swift system reconfiguration on the fly.

When designing user interfaces, we have found that selection and presentation of physiological signals is a delicate issue relying on human perception. We believe that multi sensory perception could improve insight into complex biomedical phenomena.

CONCLUSIONS

Although standardization is key issue for wider acceptance of telemedical information systems, experience have shown that implemented systems often remain proprietary. Developed systems are often expensive, and does not have enough input to allow seamless integration with existing system. In this paper, we presented the telemedical environment based on Virtual Medical Devices implemented with Java mobile agent technology. Proposed agent based framework requires only Java environment, which is widely accepted as a standard environment in the area of information and communications technologies.

Our agent based VMD implementation provides ad-hoc agent interaction, with support for mobile agents and different user interface components in the telemedical system. We have found that the flexibility of distributed agent architecture is well suited for telemedical application domain. This flexibility is particularly important in the case of an emergency, enabling users to quickly build new applications and reconfigure the system on the fly. A proposed agent based solution allows novel approach to design of telemedical services.

BIBLIOGRAPHY

[1] Filman R.E., (2001), "Medicine on the Net", IEEE Internet Computing, May/June, pp. 40-41;

[2] Fricke S., et al. (2001), "Agent-based Telematic Services and Telecom Applications", Comm. of the ACM, Vol. 44, No. 6, April, pp. 43-48;

[3]Holzman T.G., (1999), "Computer-Human Interface Solutions for Emergency Medical Care", Interactions, May/June, pp. 13-24;

[4] Huston L.T., Huston L. J., (2000), "Is Telemedicine A Practical Reality", Comm. of the ACM, Vol. 43, No. 6, June, pp. 91-95;

[5] Jennings N.R., (2001), "An Agent-based Approach for Building Complex Software Systems", Comm. of the ACM, Vol. 44, No. 6, April, pp. 35-41;

[6] Jovanov E., et al., (1998), "Multi Modal Viewer for Telemedical Applications", 20th Annual Int'l Conf. IEEE Engineering in Medicine and Biology, Hong Kong, 1998.

[7] Jovanov E. et al., (1999), "Perceptualization of biomedical data", IEEE Engineering in Medicine and Biology Magazine, Vol 18, No. 1, pp. 50-55, 1999;

[8] Kannel – Open Source SMS and WAP Gateway Web site, http://www.kannel.org/;

[9] Object Management Group (OMG) Web site, (2003), http://www.omg.org/;

[10] TeleEEG, A Telemedical Software Package for EEG. University of Calabria, http://www.parcolab.unical.it/ Euromed/TeleEEG/;

[11] Starcevic, D. et al., (2000), "Virtual Medical Devices for Telemedical Applications", chapter in Spasic, P., Milosavljevic, I., Jancic-Zguricas, M., Ed's, Telemedicine, Academy of Medical Sciences of Serbian Medical Association, 2000.

[12] Wong D., Paciorek N., Moore D., (1999), " Java-based Mobile Agents", Comm. of the ACM, Vol. 42, No. 3, March, pp. 92-102;

[13] Wooldrige M.J., Jennings N.R., (1999), "Software Engineering with Agents: Pitfalls and Pratfalls", IEEE Internet Computing, May/June, pp. 20-27.