

Software Agent Based Approach Towards Tele-Electrocardiography

*Pronab Ganguly , +Pradeep Ray & #Graham Low
School of Information Systems, Technology and Management,
The University of New South Wales, Kensington,
NSW 2052, Australia,

*Tel: 61 2 9385 5803, Fax: 61 2 9662 4061; Email: p.ganguly@unsw.edu.au
+Tel: 61 2 9385 5890, Fax: 61 2 9662 4061; Email: p.ray@unsw.edu.au
#Tel: 61 2 9385 4413, Fax: 61 2 9662 4061; Email: g.low@unsw.edu.au

Abstract

Telemedicine involves the integration of information, telecommunications, human-machine and healthcare technologies. One of the aspects of telecardiology known as tele-electrocardiography deploys Electrocardiography machines to transmit Electrocardiograms (ECG) over networks. Tele-electrocardiography diagnosis and ECG interpretation is simple, reliable and cost-effective. As different parts of this tele-electrocardiography system work on different computing environments, software interoperability is a major issue. Software agent technology supports heterogeneous computing environment. As this technology reacts dynamically to adverse conditions, it is suitable for development of fault tolerant distributed systems. Additionally, agent technology represents the intentions, desires and resources of the participants - so it can be effectively used in on-line healthcare application. The features of software agent technology are potentially suitable for the development of interoperable telemedicine systems. This paper presents a software agent based interoperable scheme for tele-electrocardiography application.

1. Introduction

Telemedicine involves the integration of information, telecommunications, human-machine and healthcare technologies. As in other distributed systems, interoperability is a major problem in telemedicine. The interoperability problem in telemedicine is manifested in patient monitoring, diagnostic, decision support and communication systems required at the point of care. Software interoperability provides necessary mechanisms to integrate these disparate but connected resources into a single computational environment to achieve effective utilization. Software agents have potential to offer semantic interoperability. A scheme based on software agents is presented here for tele-electrocardiography.

The next section presents a brief description of the distributed tele-electrocardiography application. The subsequent section describes software agent-based scheme for tele-electrocardiography application. The concluding section mentions evaluation strategy and further research needed in this area.

2. Distributed Tele-electrocardiography Application

Early detection of coronary heart diseases is very challenging task. The ElectroCardioGram (ECG) is signature of the heart and records voltage changes transmitted to the body surface from electrical events in the heart muscle. One of the aspects of telecardiology known as tele-electrocardiography deploys ECG machines to transmit ECGs over network. Various research studies show that tele-electrocardiography diagnosis and ECG interpretation is simple, reliable and substantially cheaper in cost in comparison to conventional referral systems [14]. The major benefits of tele-electrocardiography are [14]: faster access to diagnosis, improved quality of care, reduction in hospitalization cost and better patient management.

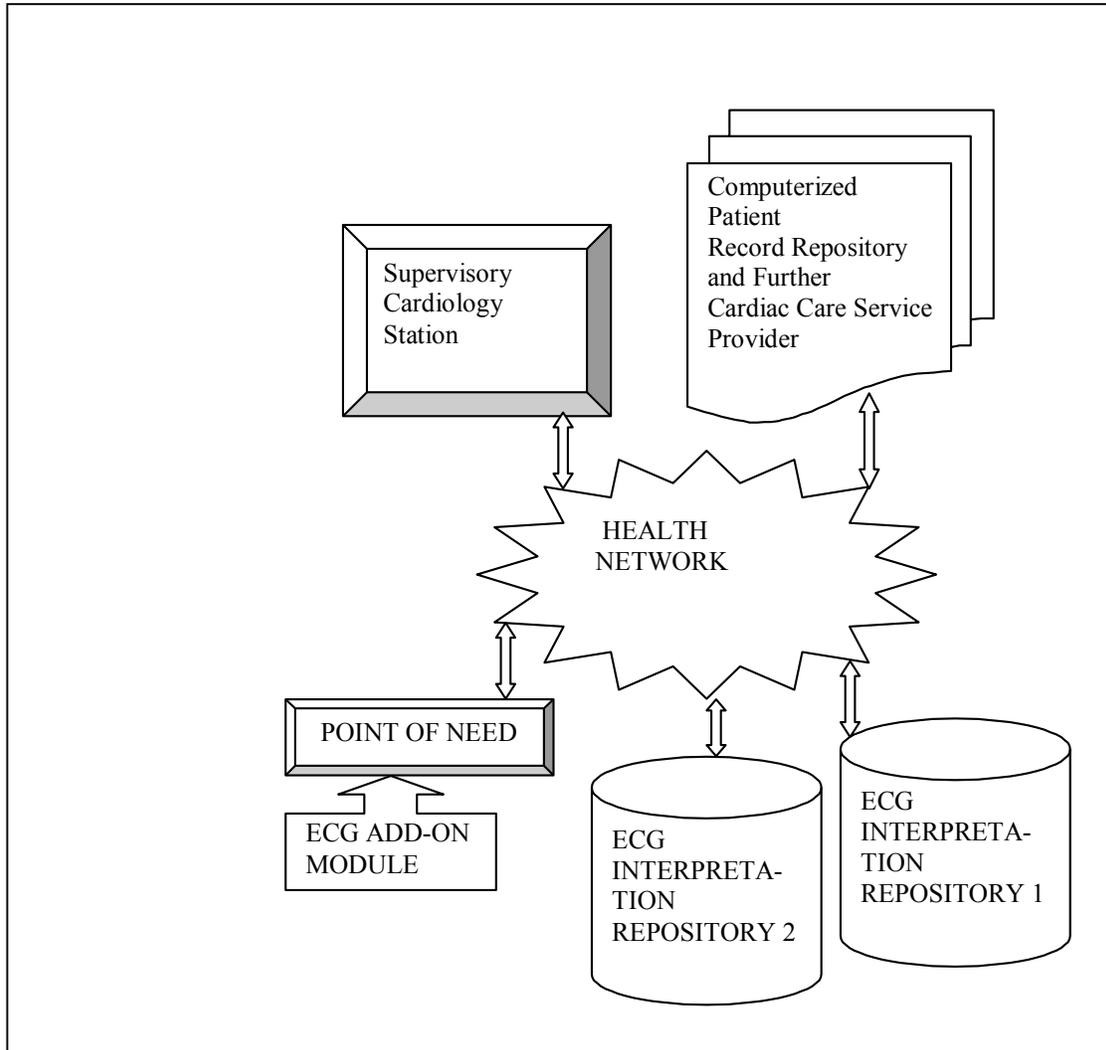


Figure 1 Distributed Framework for ECG diagnostic system

In a distributed framework for a tele-electrocardiography system [5] shown in Figure 1, the point-of-need user with the help of front-end ECG data-acquisition machine accesses and invokes the distributed knowledge repository over the network (private or public with necessary security). The interpretation repositories diagnose the ECGs on-line, and forward the ECG data along with multiple interpretations to the on-line supervisory cardiologist. For normal cases, results are distributed over the network by the Supervisory Cardiology Station to the point-of-need and Computerized Patient Record Repository. For abnormal cases, Further Cardiac Care Service Provider is additionally alerted to initiate action. In this

framework [5], different users might use heterogeneous computing platforms such as the point-of-need user might use a Windows NT platform with a compatible set of application software whereas the Supervisory Cardiology Station might use Unix computing environment with a different set of application software. The other building blocks in the framework might use various other computing platforms as well. These different computing platforms need to interconnect, share data and act cooperatively with each other. In this application, the key issues related to interoperability are [1], [13]: interoperability between various platforms, standard communication protocol between ECG data and knowledge repositories, ECG data storage and presentation of ECG. SCP-ECG Standard Communication Protocol (from Comite European de Normalisation (CEN) / TC251) specifies the format of ECG data to be exchanged with interpretation repositories.

Method of Information Exchange	Interoperability Paradigm
ECG data & Interpretation by Fax [11]	Physical / Data-type
Store and Forward of ECG File	Data-type
PC based networked system [1]	Data-type
Client-Server architecture [15]	Data-type
Laptop & wireless Transmission [7]	Datatype
Open European Electrocardiological Data Interchange Project [13]	Data-Type
Corbamed / ActiveX based Project [5]	Specification-level

Table 1: System solutions for existing tele-electrocardiography programs

3. Software Agent-based Scheme for Tele-electrocardiography

Software interoperability may be divided into four paradigms [8]: Physical, Data-type, Specification-level and Semantic. A brief summary of existing system solutions for tele-electrocardiography is given in Table 1. Semantic interoperability assumes that the components of the distributed application will have different expectations of the data. In this approach, a system is designed to use different abstract views of shared entities. This model inherently represents design intent, behavior and structured description of the entities. Usually human intelligence is required to assess semantic information. Trends are emerging to automate this process. Software agent-based paradigms offer a potential solution. A software agent is a proactive object. A software entity is an agent if it has [4]:

- ❑ the data and code encapsulation of a software object;
- ❑ its own thread of control (active); and
- ❑ the ability to execute autonomously without being invoked externally.

The design of the agent based system involves [9], [12], [16]:

- Conceptual Analysis
- Role Playing
- Implementation Design

3.1 Conceptual Analysis

In conceptual analysis, system level behavior is specified and kinds of agents are defined [16]. For our system, the following agents and their behaviors are identified. Figure 2 provides a higher level scheme for agent-based tele-electrocardiography system.

- User Interface Agent for ECG Acquisition (UIAEA): This agent acquires the ECG data at the point-of-need.
- Point-of-need Data Management Agent (PDMA): This agent manages the local database related activities at the point-of-need.
- User Interface Agent for On-line Supervisory Station (UIAOSS): This agent provides the necessary diagnosis interface for online supervisory station.
- Supervisory Station Data Management Agent(SSDMA): This agent manages the local database related activities in online supervisory station.

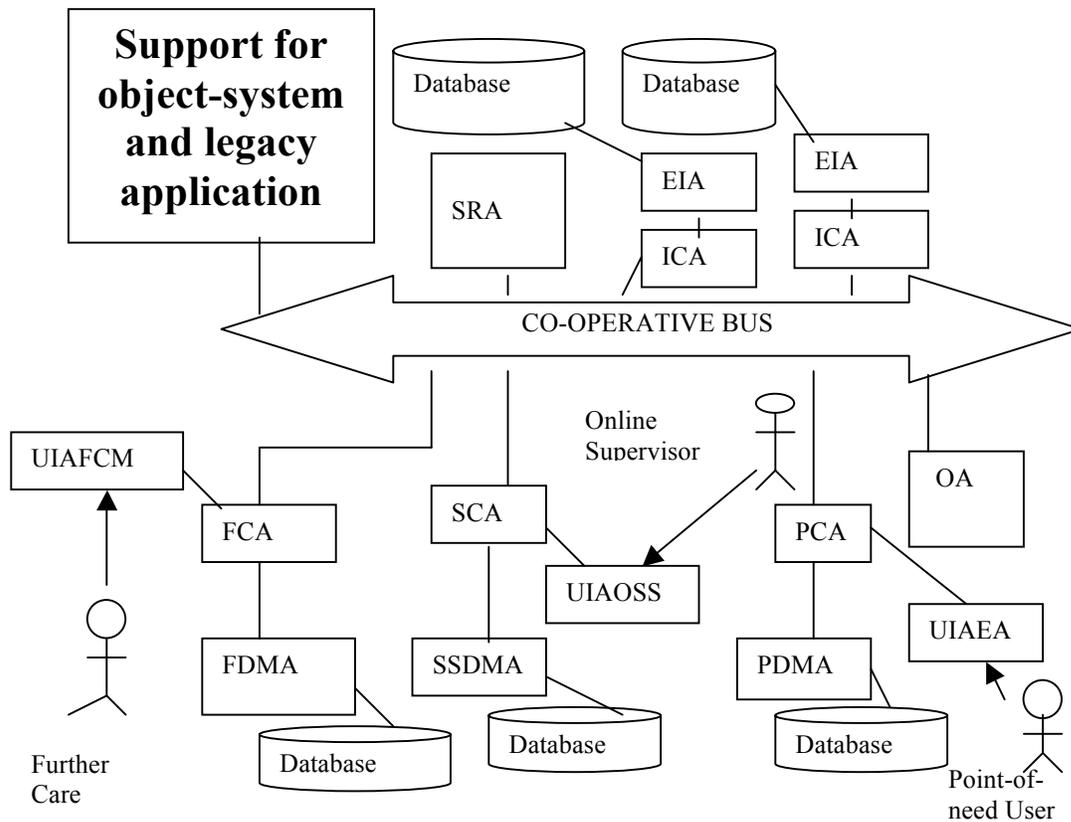


Figure 2 Scheme for Agent-based Tele-electrocardiography System

- User Interface Agent for Further Care Mechanism (UIAFCM): This agent provides the necessary interface for the further care mechanism.
- Further-care Data Management Agent (FDMA): This agent manages the local database related activities in further care mechanism centre.
- ECG Interpretation Agent (EIA): This agent presents the ECG data to online knowledge repository and obtains the automated ECG interpretation.

3.2 Role Playing

In this phase, the interactions between the proposed agents are defined [16]. We have adopted a federated approach for interaction between agents in our application. In this approach, the local agents interact with facilitator - we specify them as Cooperation Agents (CA). These cooperation agents interact through cooperative bus over the network. In our application, PDMA and UAIEA interact with a Point-of-need Cooperation Agent (PCA). UIAOSS and

SSDMA interact with Supervisory Cooperation Agent (SCA). UAIFCM and FDMA interact with Further-care Cooperation Agent (FCA). EIA interacts with Interpretation Cooperation Agent (ICA). PCA, SCA, FCA and ICA interact through a cooperative bus. The Ontology Agent (OA) provides a common ontology related to ECG interpretation. This ontology such as details of ECG format (sample rate, duration etc) provides mutual understanding between cooperation agents. Semantic Router Agent (SRA) stores ontological agreements.

3.3 Implementation Design

For the implementation of a prototype, there is a need for a platform which provides the basic infrastructure and a communication protocol for implementation of inter-operable software agents. We have chosen ObjectSpace's Voyager for prototype implementation over other agent platforms for the following reasons:

- Seamless integration of agent framework with distributed computing;
- Supporting communication architecture for CORBA, COM, RMI (Remote Method Invocation) and EJB (Enterprise Java Beans); and
- Better support for security with secure channels [10].

There are various academic and commercial tools available for design of agent based systems. However for implementation, detailed documentation and support are very important. Practical criteria include [16]:

- implementation language: the programming language in which the agents will be implemented;
- hardware: the hardware/OS on which these agents can exist;
- availability: the ease of availability of the platform; and
- domains: the domains this platform has been used in;

A brief comparison list [10], [16] for well-supported agent platforms is given below. In our case the ECG knowledge repository is available in a DCOM environment. So direct compatibility for DCOM is a prime requirement.

Agent Platforms	Language	Hardware/OS	Direct Compatibility with CORBA & RMI	Direct Compatibility with DCOM
IBM's Aglet	Java	Java Virtual Machine	Limited	NO
JATLite	Java	Java Virtual Machine	Limited	NO
Voyager	Java	Java Virtual Machine	YES	YES

Table 2 Brief comparison of Agent Platforms

In Voyager, secondary objects, known as facets, can be attached to primary objects in run-time. The primary object with its facets is known as aggregate. An agent is made up of one or more objects. In general, the agents and repositories are being developed with Voyager's special classes and interfaces. They include Agent, Messenger and Database [3]. We are currently working on detail system design and criteria of evaluation for proposed prototype system.

4. Conclusion

The system needs to be deployed and evaluated in a clinical environment. An evaluation [2] strategy based on functional, technical, organizational, clinical and economic factors are under consideration. Though semantic interoperability is increasingly being investigated by software agent paradigms, issues such as scalability, crash recovery, inconsistencies and security need to be properly addressed [6].

References

- [1] Berdusis K; The state of the art: Tele-echocardiography and telecardiology; *Telemedicine Today*; 7(3); 25-6, 35; June 1999
- [2] Degoulet P et al; Multimedia workstations: electronic assistants for the healthcare professional; *Yearbook '96 of Medical Informatics*; Jan H van Bommel, Alexa T. McCray (eds); Stuttgart: Schattauer, 1996, page 65-75
- [3] Falchuk B & Karmouch A; Visual Modeling for Agent-based Applications; *Computer*; ISSN 0018-9162; December 1998; page 31 –37
- [4] Franklin Stan & Graesser Art; Is it an Agent, or just a Program?; A Taxonomy for autonomous Agents; *Proceedings of the Third International Workshop on Agent Theories, Architectures, and Languages*, Springer-Verlag; 1996
- [5] Ganguly P & Ray P “Telemedicine over Enterprise-wide Networks : A case study”; *Proceedings of Globecom'98*; Sydney, 8 –12 November, 1998
- [6] Genesreth R. M, An Agent-Based Framework for Interoperability, *Software Agents*; Jeffery M. Bradshaw; AAAI / MIT Press; 1997; page 317- 345;
- [7] Giovas P et al ; Transmission of electrocardiograms from a moving ambulance; *Journal of Telemedicine and Telecare*; 4 (Suppl 1): 5-7; 1998
- [8] Howie T. Cameron et al; Software Interoperability; *Center of Integrated Facility engineering*; Stanford University; 25/12/1996
<http://www.dacs.dtic.mil/techs/interop/title.shtml>
- [9] Leclercq E et al; ISIS: A semantic Mediation Model and an Agent Based Architecture for GIS Interoperability; *Proceedings of 1999 International Database Engineering and Applications Symposium*;
- [10] Mahmoud QH; *Distributed Programming with Java*; Manning Publications Co; 1999; page 238-239; 240-241; 255-289
- [11] Ong K et al; A telemedicine system for high-quality transmission of paper electrocardiographic reports; *Journal of Telemedicine and Telecare* 1 (1) : 27-33; 1995
[http:// church.computer.org/proceedings/ideas/0265/026500087.pdf](http://church.computer.org/proceedings/ideas/0265/026500087.pdf); Accessed on 29/12/1999
- [12] Ramesh v; et al; An intelligent agent-based architecture for interoperability among heterogenous medical databases; <http://hsb.baylor.ramsower/ais.ac.96/papers/ramesh2.htm>;
23/12/1999
- [13] Rubel P. et al; Open European Electrocardiological Data Interchange; *Proceedings of Health Telematics*, 1995, ISBN: 88-7958-003-5; page 381-386
- [14] Shanit D et al; Towards a comprehensive telecardiology monitoring centre for community-based services; *Journal of Telemedicine and Telecare* 3(Suppl): 60-2; 1997
- [15] Tong DA et al; A client/server system for remote diagnosis of cardiac arrhythmias; *Proceedings of the Annual Symposium on Computer Applications in Medical Care*: 601 – 5; 1996
- [16] Van Dyke Parunak, *Practical and Industrial Applications of Agent-Based Systems*, Industrial Technology Institute, 1998; <http://www.iti.org/~van/> ; 03/04/99