Mobile Agent Architecture for Networking in Hospital Organizations and Healthcare Enterprises

Meiappane.A, Maheswaran.S, Prabhakaran.M and LakshmiNarayanan.A

***Abstract*—Large-scale networked environments, such as the Internet, possess the characteristics of distributed data, distributed access and distributed control; this gives the user a powerful mechanism for building and integrating large repositories of distributed information from diverse resources set. However, a distributed network system with mobile agent deployment development for a hospital information systems or a health care information portal is still in its infancy. The currently available tools are software-based or Web-based. Software-based tools must be installed on each computer prior to using the tool. For the use of Web-based tools, access to the Internet is required. The shortcomings of the currently available tools have made the use of mobile devices more appealing. In mobile computing, the issues such as low bandwidth, high latency wireless networks, loss or degradation of wireless connections, and network errors or failures need to be dealt with. Other issues to be addressed include system adaptability, reliability, robustness, extensibility, flexibility, and maintainability. Mobile agent approach has emerged as the most viable approach for development of intelligent software applications for wireless mobile devices in a distributed environment. We have developed a demonstration prototype using Java-based agent technologies and Java-based mobile technologies. In this paper, we present the architecture and the demonstration prototype, and discuss the approach, and future plans.**

Index Terms—**Mobile Agent, Distributed Computing, Hospital System, Healthcare Information System.**

I. INTRODUCTION

The mobile agent approach has received much attention in the research community since mobile agents are mobile, flexible, autonomous, dynamic and efficient. When encapsulated with a task, a mobile agent can be dispatched to a remote host by the original host. After executing and accomplishing its tasks locally, it can send the results back by returning to the original host or sending through a message.

If the tasks to be accomplished by a mobile agent involve multiple hosts in a specific order, the mobile agent can decide and visit the next host automatically until its tasks are all accomplished. The mobile agent approach is suitable for deploying parallel processes over distributed environment like hospital networks.

The tasks can be decomposed and encapsulated to multiple mobile agents. Every mobile agent can run alone to accomplish its task. And all the mobile agents can run in parallel on distributed hosts so that the whole tasks can be completed in a short time.

These features show that the mobile agent approach is well suitable for applications where parallel updating in a distributed network which involves wireless devices. A typical example is the network in today's hospital organization and health care enterprises where there is a increasing use of mobile devices. In this case, the mobile agent approach can not only utilize the consumer-agent mode of real life, but also deploy parallel search activities for retrieving information about patients and other related information.

In this paper, we first proposed a framework for building architecture for hospitals based on mobile agents for parallel processing.

II. RELATED WORK

There has been an increasing amount of research activities to exploit mobile agents to support enterprises. These works benefit much from the deployment of mobile agents, such as good mobility, high autonomy as well as the role simulation and role specification that present the realistic simulation to the real commercial activities. But they simply put mobile agents in a serial working pattern and their global control mechanisms are not clear. In addition, a few works apply the mobile agent approach to distributed application.

A. D. Stefano et al. proposed a new model of a distributed DBMS based on the mobile agent programming paradigm instead of the client/server. His work investigated the suitability of the mobile agent approach to the problem of integrating a collection of local DBMS into a single heterogeneous large-scale distributed DBMS and it presented a model for distributing distributed transactions to a set of mobile agents.

The authors proposed that parallel processing was one of the application areas of mobile agent approach. Their paper

A.Meiappane, is with the Department of Information Technology, SMVEC, Puducherry 605 107, India
Maheswaran.S, Final Year (UG), Department of Information Technology, SMVEC, Puducherry 605 107, India
Parbhakaran.M, Final Year (UG), Department of Information Technology, SMVEC, Puducherry 605 107, India
LakshmiNarayanan.A, Final Year (UG), Department of Information Technology, SMVEC, Puducherry 605 107, India

described JAMES, a Java-based platform that provides support for parallel computing.

The group of G. Samaras pointed out that the current commercial applet-based methodologies for accessing database systems offered limited flexibility, scalability and robustness in comparison with the Java based mobile agent systems. Their work showed the performance of the mobile agent system was comparable to, and in some case outperformed, the current approach. They also proposed a mobile agent based parallel processing framework that used multiple mobile agents. The performance issue is another important consideration for adopting the mobile agent approach when building a distributed framework for hospitals. Particularly, doctors need to know the current status of patients and a large number of mobile agents should be employed.

In such an environment, obviously serial migration will not give satisfied performance and novel dispatch models for large-scale mobile agents are desirable.

III. Enterprise Digital Hospital System

Managing clinical information is a challenge with unique requirements, and so far, no system has been able to address the complexity of the entire hospital environment. Some medical information systems such as Hospital Information System (HIS), Picture Archiving and Communication System (PACS), Radiology Information System (RIS) and Laboratory Information System (LIS) are used in hospitals now, but they are usually heterogeneous and isolated. Data is incomplete, workflow is discontinuous, and management is not uniform.

Therefore, it has been medical staffs' dream to establish the Enterprise Hospital Information System that could integrate all heterogeneous systems and make all clinical data including clinical report, lab results and medical images available whenever and wherever they are needed. Many efforts have been made on integrating the heterogeneous systems in hospital. Some of them define standardized interfaces to many healthcare "Object Oriented Services" such as CORBAmed (Common Object Request Broker Architecture in Medicine), which realizes the share of common functionalities like access control among different systems. Others, like DICOM (Digital Image Communication in Medicine), HL7 (Health Level Seven) and the initiative of IHE (Integrating the Healthcare Enterprise), specify the guidelines or standards for exchanging messages among different systems, which make the different systems working in harmony and implement the workflow integration. Due to the existence of a large variety of heterogeneous systems, three aspects of integration are required to implement an enterprise system, including data integration, workflow integration and function integration. However, the above efforts have only dealt with one or two of them, which makes the integration insufficient, e.g., it's difficult to establish one data center containing all clinical data only through message exchanging or common functionalities.

To solve the problems, this article proposes an architecture design that deploys mobile agent to accomplish the specified task. This design achieves the goal of complete integration among heterogeneous systems within the hospital.

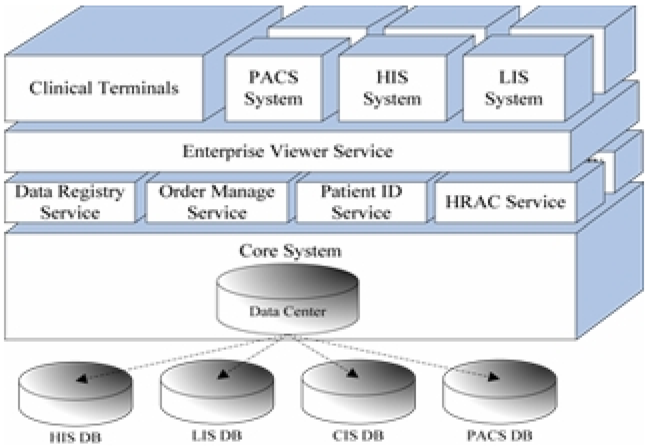

Fig. 1: The Enterprise Digital Hospital System.

IV. The Design of Architecture

In order to establish an enterprise hospital information system, the integration among these heterogeneous systems must be considered. Complete integration should cover three aspects: data integration, function integration and workflow integration. However most of the previous design of architecture did not accomplish such a complete integration.

This architecture design of the enterprise hospital information system based on the concept of mobile agent deployment in the network. It covers all three aspects of integration, and eventually achieves the target.

A. . Three stages for a mobile agent

When a mobile agent is dispatched to a remote host to accomplish a specified task, the whole process can typically be decomposed as follows.

(1) Dispatching

In this stage, the dispatched mobile agent should first be created by the master agent, which is mobile or stationary. When it is created, some arguments are encapsulated into the mobile agent, including the task, the address information of the master agent for sending back results. The code for accomplishing the task should also be included in the dispatched mobile agent. After the mobile agent has been created, the master agent will dispatch it to the remote host.

Generally, the time for this stage depends on the bandwidth, traffic state of the network and the size of the mobile agents and the dispatch algorithm. The dispatch process is mainly a network-intensive job.

(2) Accomplishing Tasks

If the dispatched mobile agent has been successfully dispatched to the remote host, it should begin to execute and access local data so as to accomplish its task. Due to the characteristics of the task, the mobile agent can communicate with local stationary agent or access the local data, such as XML documents or database, directly. Since the mobile agent approach is well suitable for deploying parallel processing over distributed data resources, a mobile agent can be assigned a simple task so that it can visit only one remote host to accomplish its task. A mobile agent can also be assigned a set of tasks that should be accomplished by visiting a set of remote hosts. If these tasks are semantically dependent and should only be finished in a specific order, dispatching one mobile agent is essential and good enough that it can migrate in an itinerary pattern. Otherwise, if these tasks are semantically independent and the number of remote hosts that should be visited is large, these tasks should be distributed to multiple mobile agents so that each mobile agent has only one relatively simple task that it will not take a long time to accomplish it. Thus, the master agent can get all the final results in a short time since these dispatched agents can execute in parallel over different processors.

(3) Sending Results Back

When a dispatched mobile agent has accomplished its task, it should send back the results. It can either dispatch itself to the origin host carrying its results or send the results back through a message. Generally speaking, the latter way can be faster since the former way should send back both the results and the code of the mobile agent. This way is necessary when the network is partially connected or the connection is dynamically changed, where the autonomous migration of a mobile agent can help to choose different route for returning. As introduced above, the execution time for a mobile agent at the remote server side depends on many factors. They include the nature of the task, such as how many data the mobile agent should access, the complexity of the task, such as whether it is a simple query or a conjunctive query, and the current state of the remote server. For the period of sending back results, when a large number of mobile agents are dispatched, it is difficult to restrict a model for data collection since most mobile agents have different tasks and the individual execution time of each mobile agent is dependent on many factors as discussed above. In the hospital applications, the size of result data sets is generally large in most cases.

Therefore, when a large number of data sets of small size are sent back in different time periods, it is unlikely to result in network congestion. The whole execution time is hereby dependent on the mobile agent who is the last one to complete its task.

However, when a large number of mobile agents are dispatched for the same kind of task, such as searching the patient details and reports, the dispatch process can cause bottleneck easily at the dispatching side. In this case, the serial dispatch process is obviously not a good choice. If the dispatching process can be divided into several segments so that some of them can be moved to different hosts and they can be processed in parallel, the total dispatch time can be reduced. Based on this idea, hierarchical dispatch models can greatly improve the performance.

B. Parallel dispatch models

In a hierarchical model, the Master Agent is only responsible for dispatching Primary Worker Agent (PWA) and the dispatch work is partially moved to PWAs. Each PWA is responsible for dispatching a cluster of PWAs or Worker Agents (WA). A WA performs the assigned task at the remote server. In comparison to the model in which the Master Agent should dispatch all the mobile agents, the Master Agent in the hierarchical model has greatly reduced its load.

(1) Master Agent (MA): A MA is a mobile agent who is responsible for offering the interface to end users for inputting query tasks, decomposing these tasks, dispatching mobile agents for accomplishing the tasks, and collecting results.

(2) Primary Worker Agent (PWA): A PWA is created and dispatched by a MA or a PWA. When it has been dispatched to a remote server, its main task is to dispatch Worker Agents to remote hosts, and distribute the tasks from the MA to these Worker Agents. In an optimized model, a PWA can also carry its own data access task besides the dispatching tasks and can begin to accomplish the task after it has finished the dispatch tasks.

(3) Worker Agent (WA): A WA is a mobile agent who is created by a MA or a PWA and dispatched to a remote server for accomplishing the task specified by its master agent. After having accomplished its task, the WA should report to its master agent, a PWA or a MA, for sending the results back. To illustrate each hierarchical model clearly, we introduce the notion of Dispatch Tree (D-Tree). A D-Tree is a tree in which the root vertex is the Master Agent, each leaf is a WA and each non-leaf middle vertex is a PWA if it exists. Every edge is a directed edge denoting the dispatching process that the parent vertex dispatches the son vertex. To be consistent to the hierarchical models discussed, we restrict the height of a D-Tree to be no less than 1 and it should have at least 2 leaves.

To simplify, the analysis and discussion are taken in this paper with the assumption that the time for dispatching a PWA or a WA in each model is the same.

Before discussing any hierarchical dispatch models, we first introduce the simplest model to dispatch multiple mobile agents, which is termed model Serial dispatch model.

Serial Model: In this model, the Master Agent dispatches a cluster of mobile agents one by one. It is in fact a Serial dispatching model. In the dispatch tree corresponding to this model, the height of the tree is 1. This model is the simplest one and obviously the slowest. Its time complexity is O(n) where n is the number of dispatched WA's(Fig:2).

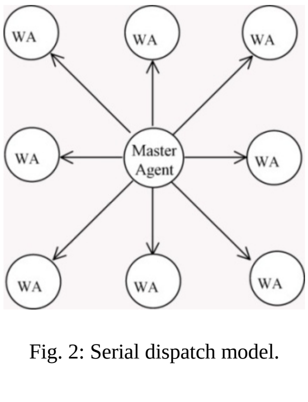

Fig. 2: Serial dispatch model.

The next simple hierarchical model H1, the height of the D-Tree is fixed to 2.That means, as shown in Figure 3, that the Master Agent dispatches a group of PWAs and each PWA dispatches a cluster of WAs which try to accomplish their tasks.

H1 is easy to operate when programming. The Master Agent can divide all the WAs into several groups and distribute them, together with their corresponding tasks and the addresses of the destination hosts, to PWAs.

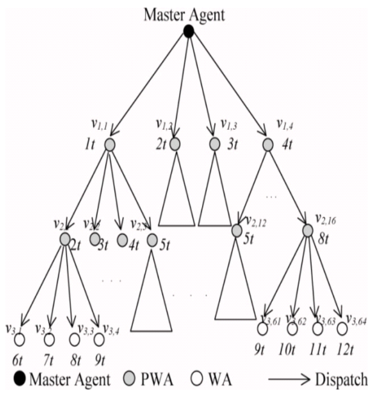

Fig. 4 Hierarchical dispatch model: H2

Since there are 64 WAs to be dispatched finally, they are first distributed to the groups hold by 4 PWAs. Each PWA is responsible for dispatching 16 WA's altogether. Taking v1, 1 as an example, it should dispatch four PWAs (i.e., vertex v2, 1, v2, 1, v2, 3, v2, 4) first and each PWA dispatches 4 WAs so that 16 WAs are dispatched finally in this group. Suppose the time for dispatching a PWA or a WA is t, the total dispatch time by model H2 is 12t, which is greatly shorter than 64t of Serial model.

C. The structure of the mobile agent platform

Aiming at the shortcoming of the general mobile agent platform, a new type of mobile agent platform is putted forward whose structure is showed as figure 4. Combining the characteristic of distributed web database, the platform also adopts distributed structure. According to the structure, its frame can be divided into five parts.

(1)The basic construction: The construction of the server of distributed database application.

The mobile agent offers the interface accessing the distributed database; Agent security server offers services of monitoring agent state, fault tolerance and mobile agent platform registering; Center data server offers the services of interpreting task and generating the task mobile agent; Local data server offers the data needed and the interface of access the data.

(2)Mobile agent sub-platform: These platforms rest on different servers respectively and offer their respective services. Agent server for entrance resting on the server offers sending and call backing function of the mobile agent of task guide and the transmitting function of the external mobile agent; Agent server for security resting on agent security server takes charge of registering the agent platform, monitoring the mobile agent and the mobile agent's fault tolerance, and have the function of checking and transmitting the external mobile agent, but it doesn't offer the function of executing the mobile agent; Agent server for task resting on center data server takes charge of generating and managing the task mobile agent. In the new mobile agent platform, it is the sole mobile agent sub-platform that possess the function of generating mobile agent, and it is the only interface with which the external mobile agent and the application exchange; Agent server for data-access resting on local data server offers the interface with which the mobile agent and local data exchange. In this way it's beneficial to the data's security and the application's expandability.

(3)Mechanism of accessing data: It belongs to the application management layer. Firstly, with the mobile agent of task guide, it generates user's request of accessing data, which improves the application's usability and strengthens the support for the complex inquiry; after that, the generating and sending of the mobile logic of the task mobile agent are completed by the center data server, as a result, the difficulty of developing and using this application are decreased; at last, the mobile agent complete the data access through the data interface on the local data server.

(4)Mechanism of system security: As the nucleus of the application management layer, mechanism of system security composed of modules with different functions and distributed on each server (or mobile agent platform) is responsible for the security and monitor of the mobile agent.

(5)Mechanism of system communicates: It's also the nucleus of the application management layer. Mechanism of system communicates, which is called intelligent express system, takes charge of the communication between the mobile agents. It also makes the communication process of the agent's lucid us to the developer and the user, lower the difficulty of developing and using, and improve the communication efficiency. The application management layer is composed of three parts: Mechanism of accessing data, intelligent express system and Mechanism of agent security, which distribute on each server respectively. Mechanism of accessing data is the main line and the destination of the application; express and security mechanism is the means of strengthening the usability of the application.

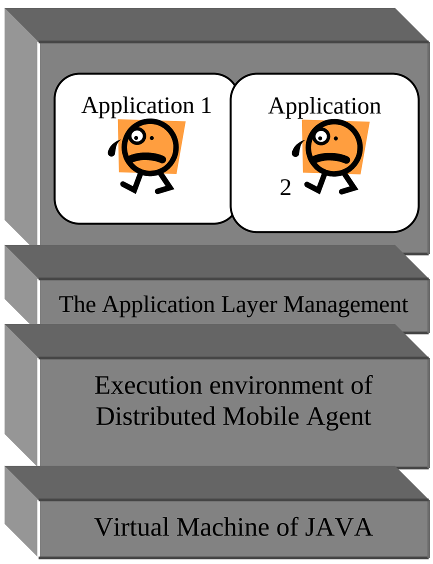

Fig. 5: The Structure of Mobile Agent Platform.

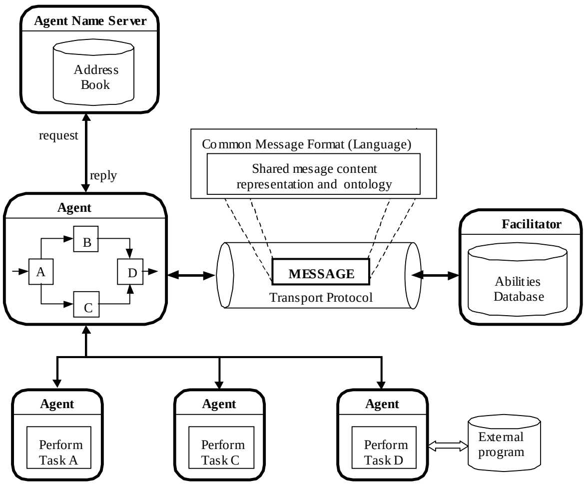

Fig. 6: Application Management.

D. Data Integration

With more and more systems being introduced into hospital, a key problem is how to keep an integrated data set. To keep all clinical data in one storage seems to be a good solution, but due to the large quantities of clinical data and the complexity to unify all data from heterogeneous systems, it proves to be effective only in small hospitals or newly-built hospitals.

In the architecture mentioned above, the concept of mobile agent for the data centers has been introduced. The clinical data are distributed and archived in corresponding departmental systems or enterprise systems. The data center stores the linkage information of each data record in different heterogeneous systems, thus functioning as a "virtual" data center for all clinical data.

The data records stored in the actual data center are 6W1H (Who, When, Where, to Whom, What, Why, How) information for the clinical events. To be ideal, any event, as long as it happens in the clinical environment, it should be recorded, including making clinical report, injecting medicine to a patient and measuring body temperature for a patient, to name a few. For an example, when a radiologist makes a report, the information of the radiologist who makes the report (Who), the time and the department of the report (When & Where), the patient whom the report belongs to (to Whom), the event type of the report (What), the event of the radiology order of the report (Why), the report status of creating, editing or confirming (How), and the data link of the report are recorded. Compared to the data structure based on patient or visit, 6W1H data structure is more flexible and could be used for different requirements of data view such as the patient record view (the events happened on the patient), the doctor task view (the events initiated by the doctor), and even the process management view (the events linked each other by the "Why" element). According to the practical situation of each existing system, the event recorded could be either cursory if the system is difficult to be expanded or in detail if the system could be tailored.

E. Function Integration

Hospitals have a strong need for affordable, interoperable information systems. These systems must operate seamlessly across a wide variety of institutions – pharmacies, laboratories, physician practices of all sizes, outpatient clinics, community hospitals, and tertiary/quaternary care regional medical centers. They have many operations (methods) in common across the clinical departments. In the architecture, these common operations are realized as a set of web services on an enterprise basis to implement the function integration.

1) Data Registry Service: In order to implement the "virtual" data centre, it's necessary for all medical information systems to register the data through the Data Registry Service to the actual data server wherever and whenever the clinical event happens through the Data Registry Service. The systems must at least register the data linkage information to the data center while the data record is created, deleted and updated. The schema of storage has the good expansibility since new systems are easier to be merged into the architecture by complying with the interface of Data Registry Service.

2) Patient Identification (ID) Service: Throughout an individual's lifetime they may have episodes of care provided by healthcare organizations. When a patient comes into a healthcare organization for care there is a need to find the records for any previous care. Each system may have used a different scheme (e.g. numbering system) to identify the patient. It is desirable to combine the medical records from multiple institutions in order to show a complete picture of a person's health record. One of the major impediments to this sharing of patient records between organizations is a lack in the ability to identify a patient in a consistent manner. Due to this inability Patient ID Service specifies the common interfaces that allow multiple systems to interoperate.

3) Healthcare Resource Access Control (HRAC) Service: The complexity of the healthcare security requires exercising more sophisticated access control policies rather than the normal ones. At the same time, requirements of commonality among systems across healthcare computing environments promote and require exercising fine-grained access control policies in a uniform and standard way. It is expected that HRAC service will fully address the security concerns and requirements of healthcare industry, including ones related to access control, auditing, non-repudiation, and notification of security breaches, and other related themes.

4) Order Management Service: Orders such as prescriptions, radiology inspection requests, ultrasound inspection requests and laboratory test requests are usually involved in the interoperation among several systems in a hospital, e.g. the HIS system requests a radiology inspection (entry an order), the RIS system receives the order and fills it. Order Management Services specifies the interfaces of order entry, order filling and order management, which facilitate the order related functions of systems involved.

IACSIT
International Association of Computer Science and Information Technology
WWW.IACSIT.ORG

F. Workflow Integration

Data integration is oriented to create the data center to manage the entire distributed clinical data, while function integration is oriented to provide common functionalities for systems in healthcare institutions. None of them focus on the workflow, which is a technology that manages and monitors processes and allows the flow of work between individuals and/or departments to be defined and tracked. It is usually implemented by the data relationship of the database in the client-server model. But in a distributed environment such as a healthcare institution, the implementation will be more complicated. Middleware technologies such as Web Service could be used to implement the digital workflow by tackling with distributed database environment just like one database in client-server model. But it has no expandability in terms of adding new system into the architecture especially when the data structures are unknown. Standard messages such as

DICOM and HL7 have already been applied in many systems to implement the workflow between different systems, but a complete integration is still difficult if there is no framework or model to guide the system to exchange the correct message in correct time.

IHE is an initiative designed to stimulate the integration of information systems that support modern healthcare institutions. Using a common framework, IHE employs existing protocols like DICOM and HL7 to connect devices, terminals and information systems in a hospital so as to fulfill the digital workflow. Up to now, it contains 12 integration profiles for radiology technical framework each of which specifies the model of one clinical application environment. Fig.7 depicts the Scheduled workflow integration profile. The Scheduled Workflow Integration Profile establishes the continuity and integrity of basic departmental imaging data acquired in an environment where examinations are being ordered. It specifies a number of transactions that maintain the consistency of patient and ordering information as well as defining the scheduling and imaging acquisition procedure steps.

There are actually four different systems involved, HIS corresponds to ADT and Order Placer actors, RIS corresponds to DSS/Order Filler and Performed Procedure Step Management actors, PACS corresponds to Image Manager, Image Archive, Image Display and Evidence Creator actors, Modality corresponds to Acquisition Modality actor. By implementing each system with the standard transactions, we could easily achieve the workflow integration.

V. DISCUSSIONS AND CONCLUSIONS

The concept of “Digital Hospital” has been put forward for years, but what kind of hospital is “Digital Hospital” is still unclear for many hospital administrators and researchers.

From our point of view, three words well summarize the characteristics of the digital hospital: “Digitization”, “Specialization” and “Visualization” well summarizes the characteristics of the digital hospital.

“Digitization” means that all clinical data, as long as it is created in the hospital, should be digitalized and managed in the data center.

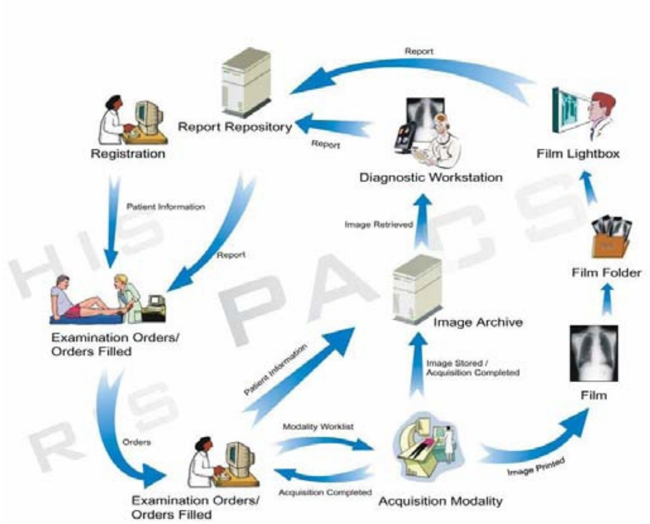

Fig. 7: IHE based workflow integration.

“Specialization” means that the specialized data view should be provided for users of each role, which in turn is provided using the mobile agent in this paper.

“Visualization” means that the specialized data view should be visualized through technologies such as virtual reality.

A future hospital should be entirely digitized: All doctors, patients, medicines and even materials could be taken as one object and visualized in the system; the static and dynamic properties of each object are presented conveniently in front of the screen; users could communicate with the all the objects through user interface rather than conversations with person or operations on device.

The architecture described in this article represents the first two concepts of the “Digitization” and “Specialization” by the implementation of three aspects of integration among the heterogeneous systems in hospital by the deployment of the mobile agents.

REFERENCES

[1] G. O. Young, “Synthetic structure of industrial plastics (Book style with paper title and editor), ” in *Plastics*, 2nd ed. vol. 3, J. Peters, Ed. New York: McGraw-Hill, 1964, pp. 15–64.
[2] W.-K. Chen, *Linear Networks and Systems* (Book style). Belmont, CA: Wadsworth, 1993, pp. 123–135.
[3] H. Poor, *An Introduction to Signal Detection and Estimation*. New York: Springer-Verlag, 1985, ch. 4.
[4] B. Smith, “An approach to graphs of linear forms (Unpublished work style), ” unpublished.
[5] E. H. Miller, “A note on reflector arrays (Periodical style—Accepted for publication), ” *IEEE Trans. Antennas Propagat.*, to be published.
[6] J. Wang, “Fundamentals of erbium-doped fiber amplifiers arrays (Periodical style—Submitted for publication), ” *IEEE J. Quantum Electron.*, submitted for publication.
[7] C. J. Kaufman, Rocky Mountain Research Lab., Boulder, CO, private communication, May 1995.
[8] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, “Electron spectroscopy studies on magneto-optical media and plastic substrate interfaces(Translation Journals style), ” *IEEE Transl. J. Magn.Jpn.*, vol. 2, Aug. 1987, pp. 740–741 [*Dig. 9th Annu. Conf. Magnetics* Japan, 1982, p. 301].
[9] M. Young, *The Techincal Writers Handbook.* Mill Valley, CA: University Science, 1989.

[10] (Basic Book/Monograph Online Sources) J. K. Author. (year, month, day). *Title* (edition) [Type of medium]. Volume(issue). Available: http://www.(URL)
[11] J. Jones. (1991, May 10). Networks (2nd ed.) [Online]. Available: http://www.atm.com
[12] (Journal Online Sources style) K. Author. (year, month). Title. *Journal* [Type of medium]. Volume(issue), paging if given. Available: http://www.(URL)
[13] R. J. Vidmar. (1992, August). On the use of atmospheric plasmas as electromagnetic reflectors. *IEEE Trans. Plasma Sci.* [Online]. *21(3).* pp.876—880.Available: http://www.halcyon.com/pub/journals/21ps03-vidmar

A. Meiappane has received his M.Tech. in Computer Science & Engg. from Pondicherry University (Central), Puducherry, India. Currently he is working as Assistant Professor in Sri Manakula Vinayagar Engineering College, Puducherry India. His different research areas are, Distributed Systems, Software Engineering & Metrics.
Email: auromei@yahoo.com

Maheswaran.S is pursuing his B.Tech. in Information Technology at Sri Manakula Vinayagar Engg. College, Puducherry, affiliated to Pondicherry University (Central), Puducherry, India. His research areas include Mobile Computing and Artificial Intelligence.
Email: macmax420@gmail.com

Prabhakaran.M is pursuing his B.Tech. in Information Technology at Sri Manakula Vinayagar Engg. College, Puducherry, affiliated to Pondicherry University (Central), Puducherry, India. His research areas include Wireless Networking.
Email: haryharane@gmail.com

LakshmiNarayanan.A is pursuing his B.Tech. in Information Technology at Sri Manakula Vinayagar Engg. College, Puducherry, affiliated to Pondicherry University (Central), Puducherry, India. His research areas include Data Mining and Warehousing.
Email: sabarialaren@gmail.com