International Journal of
Scientific Engineering and Technology Research

ISSN 2319-8885
Vol.03,Issue.13
June-2014,
Pages:2800-2806

www.semargroup.org,
www.ijsetr.com

Mobile Agents for Retrieving Exam Marks in Distributed Database System

LAI YEE MYINT[1], MOE MOE AYE[2]

[1]Dept of IT, Mandalay Technological University, Mandalay, Myanmar, E-mail: leiyee2013@gmail.com.
[2]Dept of IT, Mandalay Technological University, Mandalay, Myanmar, E-mail: moeaye255@gmail.com.

Abstract: Mobile Agents are particularly attractive in distributed information retrieval applications for several reasons, including reduction network load and server access time delay and achievement higher error tolerance and security. Moreover, Mobile Agents can offer a flexible, versatile, and powerful framework for efficiently developing distributed applications on the Internet. In the paper, Mobile Agents for Retrieving Exam Marks in Distributed Database System are described. To produce the result for Centre Of Excellent (COE) students at Mandalay Technological University (MTU), Mobile Agents and distributed databases are used. In the system, two sites are used. They are Student Affairs Site and Department Site. To get the exam marks, the administrator at Student Affairs Site sends the message to the Department Site with necessary information such as subject, major and academic year. It replies the required data to the Student Affairs Site when the message is received by negotiating with Database Agent. In this paper, the IBM's Aglet Software Development Kit (ASDK) is used for creation and deployment of mobile agents to retrieve exam marks in distributed database system.

Keywords: Mobile Agents, Distributed Databases, Exam Marks, Distributed System.

I. INTRODUCTION

The growth of Internet has led to new avenues for retrieving distributed information. In the world of today, the Internet is the most prevalent communication system and plays an essential role in education, research, business and entertainment. The Internet uses the client-server system. Most of the present day systems, client-server architectures techniques have several limitations such as susceptibility to network delays, partitions, etc. The client-server model though powerful, has scalability limitations for distributed database systems. The inadequacy of the traditional client-server approach to support wireless and mobile applications has resulted in the development of new computational paradigms. Mobile Agent is emerged to overcome the weakness of client-server system. Of all, mobile-agent paradigm becomes prevalent in distributed systems. A mobile agent is an executing program that can migrate during execution from machine to machine in a heterogeneous network. In other words, the agent can suspend its execution, migrate to another machine, and resume execution on the new machine from the point at which it left off. On each machine, the agent interacts with stationary agents and other resources to accomplish its tasks.

Mobile agents have several advantages in distributed information retrieval applications. By migrating to an information resource, an agent can invoke resource operations locally, eliminating the network transfer of intermediate data. By migrating to the other side of an unreliable network link, an agent can continue executing even if the network link goes down, making mobile agents particularly attractive in mobile computing environments. Most importantly, an agent can choose different migration strategies depending on its task and the current network conditions, and then change its strategies as network condition change. Moreover, Mobile Agent can reduce network traffic, overcome network latency, and provide to construct highly robust and fault-tolerant systems. Therefore, by using the advantages of Mobile Agent in the system, student exam marks in distributed database system can be retrieved easily, quickly and accurately to produce the result for COE students. The remainder of this paper is organized as follows. Section II describes about related work. Section III provides a brief description of Mobile Agent technology. Section IV explains a brief description of distributed database. Section V discusses the design of the proposed system. Session VI shows the implementation results. Section VII concludes the proposed system.

II. RELATED WORKS

Mobile agents have the potential to be a single, general framework in which a wide range of distributed, information retrieval applications can be implemented efficiently, easily and robustly. There are few works which deal with the use of mobile agents in a distributed database system. Some related work concentrated on the use of mobile agents in distributed Web databases [1, 2]. Simon Pears, Jie Xu and Cornelia Boldyreff [3] have described "Mobile Agent Fault Tolerance for Information Retrieval Applications: An Exception

Handling Approach". They presented an exception model whereby application exceptions are signaled between agent server and mobile agent. And, they used exception handling to maintain mobile agent availability in the presence of agent server crashes and examined the performance of two exception handler designs using an experimental case study application. Although the mobile shadow scheme offered the slower round trip times it introduced the smallest round trip time increase in the presence of agent server crashes. The timeout scheme had the disadvantage that the crash round trip time depended upon the total agent servers visited.

Brian Brewington et al. [4] have presented "Mobile agents in distributed information retrieval". In the system, they discussed the strengths of mobile agents, and argued that although none of these strengths were unique to mobile agents, no competing technique shared all of them. After surveying several representative mobile agent systems, they examined one specific information retrieval application, searching distributed collections of technical reports, and considered how mobile agents can be used to implement the application efficiently and easily. They spent two planning services that allow mobile agents to deal with dynamic network environments and information resources. Mobile agents have the potential to be a single, general framework in which a wide range of distributed, information retrieval applications can be implemented efficiently, easily and robustly.

In[5], Harald Kosch, Mario Doller and Laszlo Boszormenyi have implemented "Content-based Indexing and Retrieval supported by Mobile Agent Technology". They presented the Multi Media Database Mobile agent technology (M^3) which supports personalized content-retrieval and indexing in a distributed Oracle 8i DB. They described an agency on top of the Oracle 8i JServer and realized mobility with the embedded Visbroker Corba ORB. A performance comparison of mobile agent technology with a client-server solution for a nearest-neighbor search in an image database shows the efficiency of the proposed solution. In the context of distributed multimedia systems, mobile agents are successfully utilized for Quality of Service (QoS) negotiations [6,7]. This paper shows that it is possible and profitable to establish an agency system inside a distributed Oracle 8i multimedia database.

III. MOBILE AGENT TECHNOLOGY

Mobile agents are the basis of an emerging technology that promises to make it very much easier to design, implement, and maintain distributed systems [8]. Mobile agents reduce network traffic, provide an effective means of overcoming network latency, and perhaps most importantly, through their ability to operate asynchronously and autonomously of the process that created them, help us to construct more robust and fault-tolerant systems. They have the ability to interact with their execution environment, and to act asynchronously and autonomously upon it. No one is required either to deliver information to the agent or to consume any of its output. The agent simply acts continuously in pursuit of its own goals.

An agent is software object that:

- is situated within an execution environment;
- possesses the following mandatory properties:

Reactive—senses changes in the environment and acts in accordance with those changes;
Autonomous—has control over its own actions;
Goal-driven—is pro-active;
Temporally continuous—executes continuously;

- and may possess one or more of the following orthogonal properties:

Communicative —can communicate with other agents;
Mobile—can travel from one host to another;
Learning — adapts in accordance with previous experience;
Believable—appears believable to the end-user.

Mobility is an orthogonal property of agents. That is, all agents do not necessarily have to be mobile. An agent can just sit there and communicate with its surroundings by conventional means. These means include various forms of remote procedure calling and messaging. Agents are called that do not or cannot move stationary agents. In contrast, a mobile agent is not bound to the system where it begins execution. The mobile agent is free to travel among the hosts in the network. Created in one execution environment, it can transport its state and code with it to another execution environment in the network, where it resumes execution. By the term "state", the agent attribute values that help it determine what to do when it resumes execution at its destination. By the term "code" in an object-oriented context, the class code necessary for the agent to execute. Agents are defined in the following way: A mobile agent is not bound to the system where it begins execution. It has the unique ability to transport itself from one system in a network to another. The ability to travel allows a mobile agent to move to a system that contains an object with which the agent wants to interact, and then to take advantage of being in the same host or network as that object. [9]

Mobile agents provide the benefits for the creation of distributed systems. There are seven good reasons for using mobile agent technology.

- They reduce the network load
- They overcoming network latency
- They encapsulate protocols
- They execute asynchronously and autonomously
- They adapt dynamically
- They are naturally heterogeneous
- They are robust and fault-tolerant [10]

IV. DISTRIBUTED DATABASE

In today's world of universal dependence on information systems, all sorts of people need access to companies' databases. In addition to a company's own employees, these include the company's customers, potential customers, suppliers, and vendors of all types. It is possible for a company to have all of its databases concentrated at one mainframe computer site with worldwide access to this site provided by telecommunications networks, including the Internet. Although the management of such a centralized system and its databases can be controlled in a well-contained manner and this can be advantageous, it poses some problems as well. For example, if the single site goes down, then everyone is blocked from accessing the databases until the site comes back up again. Also the communications costs from the many far PCs and terminals to the central site can be expensive. One solution to such problems, and an alternative design to the centralized database concept, is known as distributed database.

The idea is that instead of having one, centralized database, the data are being gone to spread out among the cities on the distributed network, each of which has its own computer and data storage facilities. All of this distributed data is still considered to be a single logical database. When a person or process anywhere on the distributed network queries the database, it is not necessary to know where on the network the data being sought is located. The user just issues the query, and the result is returned. This feature is known as location transparency. This can become rather complex very quickly, and it must be managed by sophisticated software known as a distributed database management system or distributed DBMS. [11] A distributed database (DDB) is a collection of multiple, logically interrelated databases distributed over a computer network. A distributed database management system (DDBMS) is the software that manages the DDB, and provides an access mechanism that makes this distribution transparent to the user. Distributed database system (DDBS) is the integration of DDB and DDBMS. This integration is achieved through the merging the database and networking technologies together. Or it can be described as, a system that runs on a collection of machines that do not have shared memory, yet looks to the user like a single machine. [12] A distributed database system consists of loosely coupled sites that share no physical component. Database systems that run on each site are independent of each other. [13]

Distributed database system (DDBS) technology is the union of what appear to be two diametrically opposed approaches to data processing: database system and computer network technologies. Database systems have been taken from a paradigm of data processing in which each application defined and maintained its own data to one in which the data are defined and administered centrally. This new orientation results in data in dependence, where by the application programs are immune to changes in the logical or physical organization of the data, and vice versa. One of the major motivations behind the use of database systems is the desire to integrate the operational data of an enterprise and to provide centralized, thus controlled access to that data. The technology of computer networks, on the other hand, promotes a mode of work that goes against all centralization efforts. At first glance it might be difficult to understand how these two contrasting approaches can possibly be synthesized to produce a technology that is more powerful and more promising than either one alone. The key to this understanding is the realization that the most important objective of the database technology is integration, not centralization. It is important to realize that either one of these terms does not necessarily imply the other. It is possible to achieve integration without centralization, and that is exactly what the distributed database technology attempts to achieve. [14]

V. ARCHITECTURE OF THE PROPOSED SYSTEM

The architecture of the proposed system is shown in figure 1. In this system, two kinds of agents are used. They are stationary agent and mobile agent. Data Request Agent (DRA) and Database Agent (DA) are stationary agents located at the corresponding host. Mark Request Mobile Agent (MRMA) is a mobile agent that can migrate from one host to another host over the network. The associated tasks of each agent are expressed as the following sub-divisions:

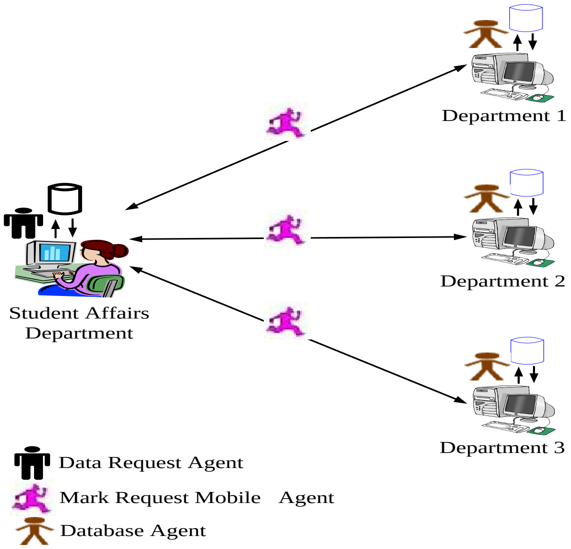


Figure1. System Overview Design.

A. Data Request Agent (DRA)

DRA is responsible to create MRMA and dispatch it to each department. It supplies a graphical user interface (GUI)

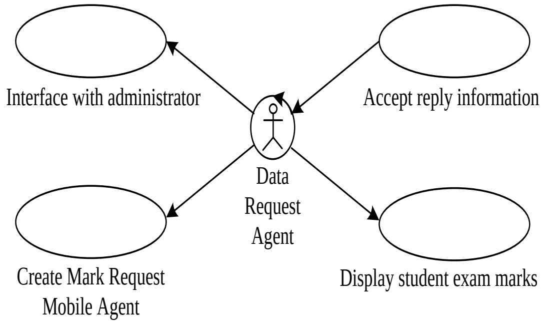

Interface with administrator
Accept reply information
Data Request Agent
Create Mark Request Mobile Agent
Display student exam marks

Figure2. Data Request Agent and its Associated Tasks.

for an administrator at Student Affairs Department to receive the reply information carrying back by MRMA, show the student exam marks as output and produce the result. The associated tasks of Data Request Agent are shown in figure 2.

B. Mark Request Mobile Agent (MRMA)

MRMA is established at the Student Affairs Department. It moves to each department to get the necessary marks information. And then, it negotiates with the DA at each department to achieve the marks information for the students. After that, it carries back the necessary student exam marks to the Student Affairs Department. The associated tasks of Mark Request Mobile Agent are shown in figure 3.

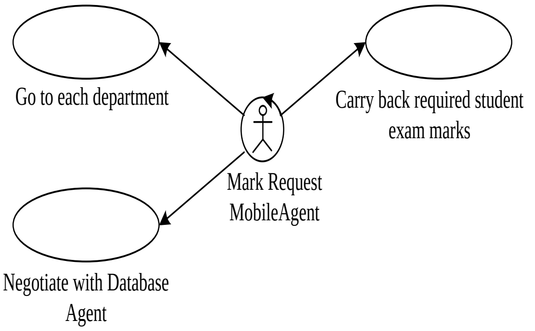

Go to each department
Carry back required student exam marks
Mark Request MobileAgent
Negotiate with Database Agent

Figure 3. Mark Request Mobile Agent and its Associated Tasks.

C. Database Agent (DA)

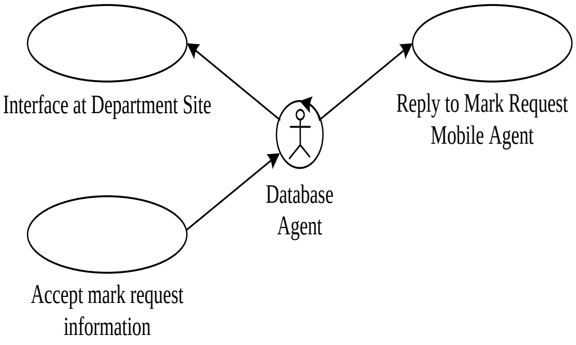
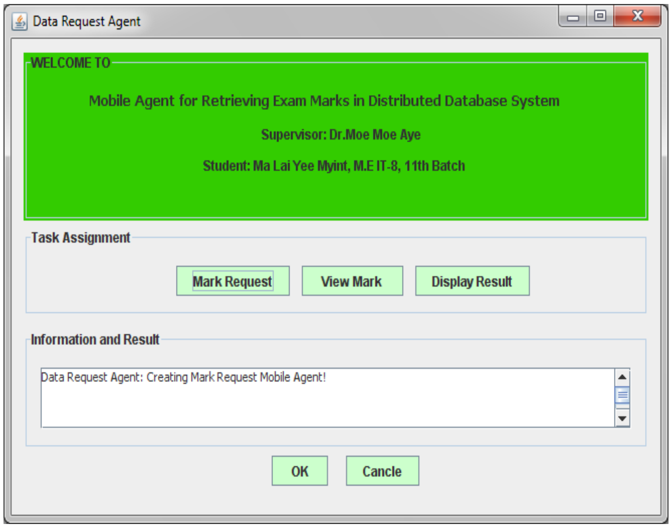

Interface at Department Site
Reply to Mark Request Mobile Agent
Database Agent
Accept mark request information

Figure 4. Database Agent and its Associated Tasks.

DA provides a user interface to an administrator of each department to receive the mark request information from MRMA for the students. After negotiating with MRMA, it replies MRMA with the required information. Figure 4 describes the functions of database agent. To achieve many advantages, the system to retrieve exam marks in distributed database is implemented by coordinating the agents mentioned above.

VI. IMPLEMENTATION RESULTS

The figures in this section show the user interface windows of the proposed system. The administrator at the Student Affairs Site can provide command in the Data Request Agent window to retrieve the student exam marks.



Mobile Agent for Retrieving Exam Marks in Distributed Database System

Supervisor: Dr.Moe Moe Aye

Student: Ma Lai Yee Myint, M.E IT-8, 11th Batch

Data Request Agent: Creating Mark Request Mobile Agent!

Figure 5. Main Window of Data Request Agent.

Figure 5 displays all processing of information flows between the Data Request Agent and Mark Request Mobile Agent. This frame consists of a series of buttons which link to the Retrieve Process. Thus, if the administrator at the Student Affairs Site wants to retrieve the student exam marks, he simply clicks on the "Mark Request" button and Mark Request Mobile Agent frame appears.

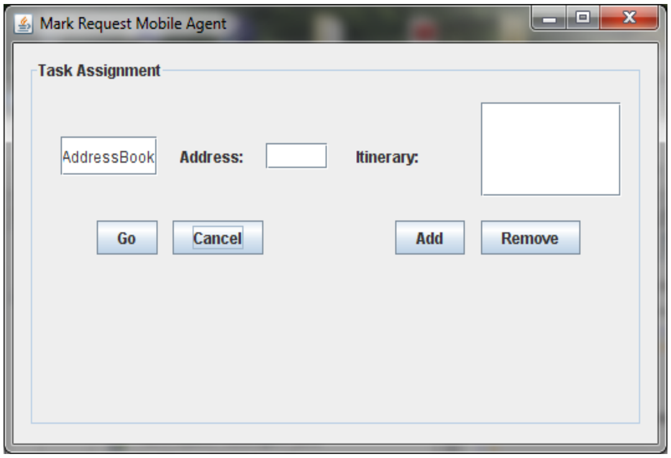
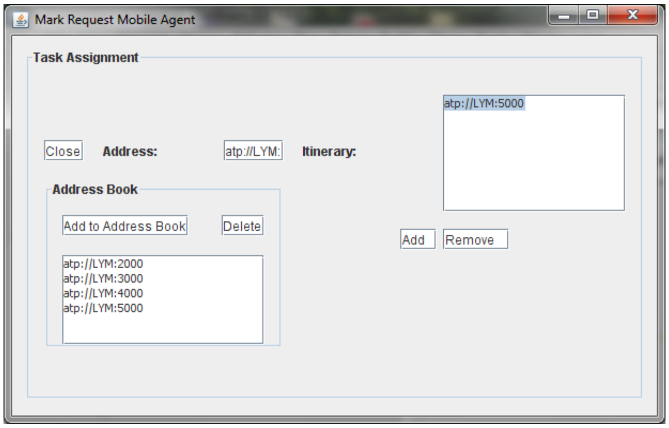


Figure 6. Mark Request Mobile Agent.

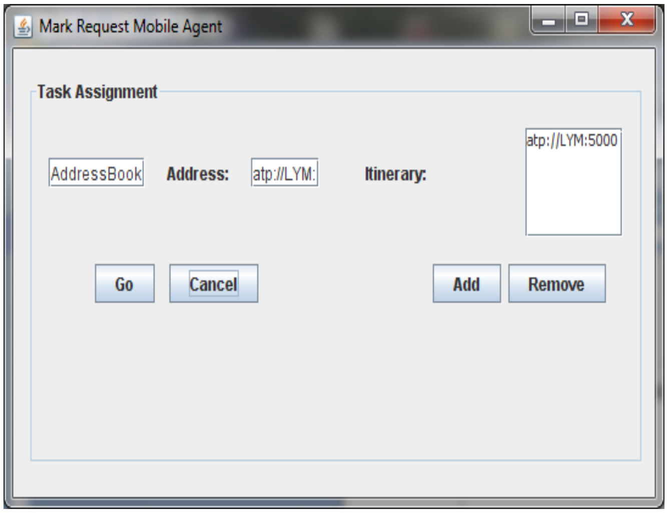


Address: atp://LYM:

Itinerary: atp://LYM:5000

atp://LYM:2000
atp://LYM:3000
atp://LYM:4000
atp://LYM:5000

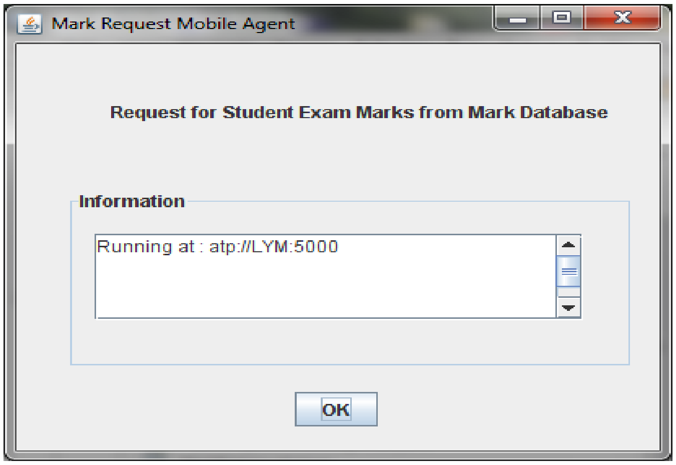
Figure7. Selecting and Adding the Address of the Remote Database.



Address: atp://LYM:

Itinerary: atp://LYM:5000

Figure 8. Dispatching the Mark Request Mobile Agent.

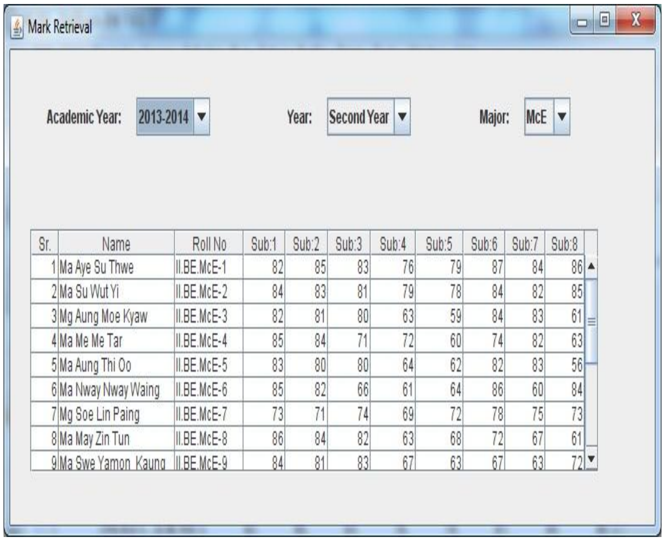


Request for Student Exam Marks from Mark Database

Information

Running at : atp://LYM:5000

Figure 9. Mark Request Mobile Agent Frame at Remote Site.



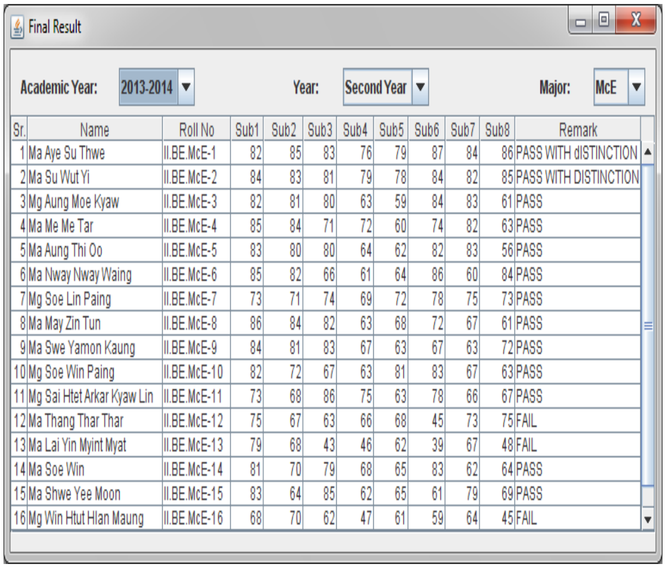
Academic Year: 2013-2014   Year: Second Year   Major: McE

Sr.	Name	Roll No	Sub:1	Sub:2	Sub:3	Sub:4	Sub:5	Sub:6	Sub:7	Sub:8
1	Ma Aye Su Thwe	II.BE.McE-1	82	85	83	76	79	87	84	86
2	Ma Su Wut Yi	II.BE.McE-2	84	83	81	79	78	84	82	85
3	Mg Aung Moe Kyaw	II.BE.McE-3	82	81	80	63	59	84	83	61
4	Ma Me Me Tar	II.BE.McE-4	85	84	71	72	60	74	82	63
5	Ma Aung Thi Oo	II.BE.McE-5	83	80	80	64	62	82	83	56
6	Ma Nway Nway Waing	II.BE.McE-6	85	82	66	61	64	86	60	84
7	Mg Soe Lin Paing	II.BE.McE-7	73	71	74	69	72	78	75	73
8	Ma May Zin Tun	II.BE.McE-8	86	84	82	63	68	72	67	61
9	Ma Swe Yamon Kaung	II.BE.McE-9	84	81	83	67	63	67	63	72

Figure 10. Mark Retrieval Window.

The Mark Request Mobile Agent shown in Figure 6 is used to retrieve the student exam marks. The administrator must give the remote database address through this interface.

The administrator then can interact with the agent by adding destination addresses as shown in Figure 7 and then dispatching the Mark Request Mobile Agent by clicking the "Go" button in Figure 8. When the Mark Request Mobile Agent arrives at the destination, the Mark Request Mobile Agent frame opens up its window at the remote site as shown in Figure 9 and negotiates with the Database Agent.

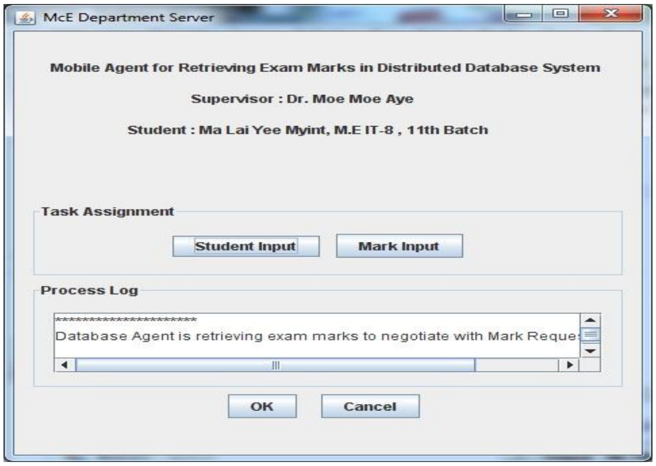


Academic Year: 2013-2014   Year: Second Year   Major: McE

Sr.	Name	Roll No	Sub1	Sub2	Sub3	Sub4	Sub5	Sub6	Sub7	Sub8	Remark
1	Ma Aye Su Thwe	II.BE.McE-1	82	85	83	76	79	87	84	86	PASS WITH dISTINCTION
2	Ma Su Wut Yi	II.BE.McE-2	84	83	81	79	78	84	82	85	PASS WITH DISTINCTION
3	Mg Aung Moe Kyaw	II.BE.McE-3	82	81	80	63	59	84	83	61	PASS
4	Ma Me Me Tar	II.BE.McE-4	85	84	71	72	60	74	82	63	PASS
5	Ma Aung Thi Oo	II.BE.McE-5	83	80	80	64	62	82	83	56	PASS
6	Ma Nway Nway Waing	II.BE.McE-6	85	82	66	61	64	86	60	84	PASS
7	Mg Soe Lin Paing	II.BE.McE-7	73	71	74	69	72	78	75	73	PASS
8	Ma May Zin Tun	II.BE.McE-8	86	84	82	63	68	72	67	61	PASS
9	Ma Swe Yamon Kaung	II.BE.McE-9	84	81	83	67	63	67	63	72	PASS
10	Mg Soe Win Paing	II.BE.McE-10	82	72	67	63	81	83	67	63	PASS
11	Mg Sai Htet Arkar Kyaw Lin	II.BE.McE-11	73	68	86	75	63	78	66	67	PASS
12	Ma Thang Thar Thar	II.BE.McE-12	75	67	63	66	68	45	73	75	FAIL
13	Ma Lai Yin Myint Myat	II.BE.McE-13	79	68	43	46	62	39	67	48	FAIL
14	Ma Soe Win	II.BE.McE-14	81	70	79	68	65	83	62	64	PASS
15	Ma Shwe Yee Moon	II.BE.McE-15	83	64	85	62	65	61	79	69	PASS
16	Mg Win Htut Hlan Maung	II.BE.McE-16	68	70	62	47	61	59	64	45	FAIL

Figure 11. Final Result Table.

When the administrator wants to look the student exam marks, he requires clicking on the "View Mark" button. After clicking this button, he can see "Mark Retrieval Window" in Figure 10. By clicking "Display Result" button, the administrator can see the result as shown in Figure 11. Figure 12 shows the main window of Database Agent at department site. In this window, "Student Input" button is to input the required student information and "Mark Input" button is to input the student exam marks. If the administrator at department site clicks on the "Student Input" and "Mark Input" buttons, "Student Information Input Form" and "Mark Input Window" are appeared as shown in Figure 13 and 14, respectively.



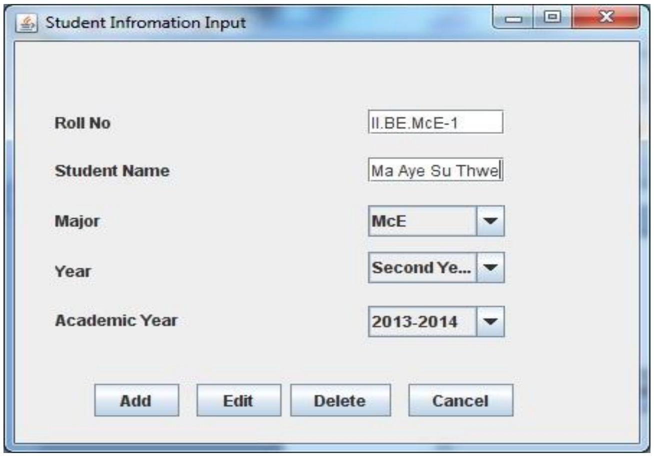
Mobile Agent for Retrieving Exam Marks in Distributed Database System

Supervisor : Dr. Moe Moe Aye

Student : Ma Lai Yee Myint, M.E IT-8 , 11th Batch

```
**********************
Database Agent is retrieving exam marks to negotiate with Mark Reque
```

Figure 12. Main Window of Database Agent.



Student Infromation Input

Roll No: II.BE.McE-1

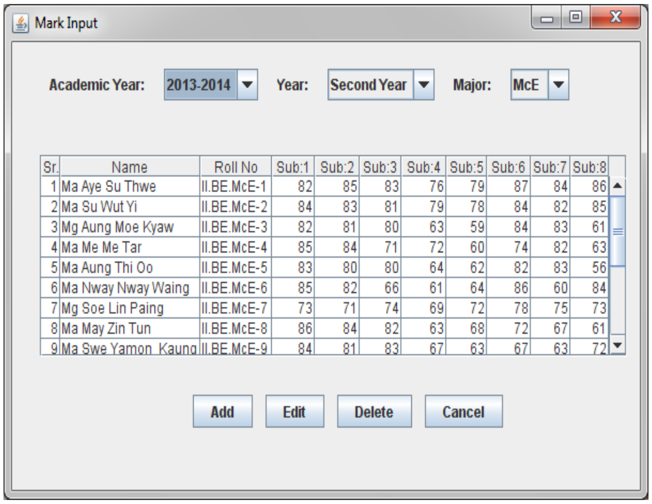
Student Name: Ma Aye Su Thwe

Major: McE

Year: Second Ye...

Academic Year: 2013-2014

Figure13.Student Information Input Form at Department Site.



Mark Input

Academic Year: 2013-2014 Year: Second Year Major: McE

Sr.	Name	Roll No	Sub:1	Sub:2	Sub:3	Sub:4	Sub:5	Sub:6	Sub:7	Sub:8
1	Ma Aye Su Thwe	II.BE.McE-1	82	85	83	76	79	87	84	86
2	Ma Su Wut Yi	II.BE.McE-2	84	83	81	79	78	84	82	85
3	Mg Aung Moe Kyaw	II.BE.McE-3	82	81	80	63	59	84	83	61
4	Ma Me Me Tar	II.BE.McE-4	85	84	71	72	60	74	82	63
5	Ma Aung Thi Oo	II.BE.McE-5	83	80	80	64	62	82	83	56
6	Ma Nway Nway Waing	II.BE.McE-6	85	82	66	61	64	86	60	84
7	Mg Soe Lin Paing	II.BE.McE-7	73	71	74	69	72	78	75	73
8	Ma May Zin Tun	II.BE.McE-8	86	84	82	63	68	72	67	61
9	Ma Swe Yamon Kaung	II.BE.McE-9	84	81	83	67	63	67	63	72

Figure14. Mark Input Window.

VII. CONCLUSIONS

Fast and efficient information retrieval in distributed environment is indeed a challenging task. In this paper, the architecture of Mobile Agent for Retrieving Exam Marks in Distributed Database System is presented. A mobile agent based framework can be used effectively for such distributed database access. It is proven to be better in performance than the conventional client – server methodology. Java-based mobile agent system, called Aglets workbench, created by IBM are fully deployed in order to develop the proposed system. Mobile agents are gaining importance in variety of applications that include distributed information retrieval, data mining, e-commerce and network management. By using the software agent in the proposed system, it can provide the advantages such as saving time, reducing the paper work and the human's effort for administrators in different departments at MTU.

VIII. ACKNOWLEDGEMENT

First, the author would like to thank her supervisor, Dr. Moe Moe Aye, Associate Professor, Department of Information Technology, Mandalay Technological University. The paper is accomplished under her supporting and supervising. Her study attitude, scientific attainments, responsibility and conscientious working give me a deep impression, and give her power to accomplish the paper. And then thank very much for all her teachers at Mandalay Technological University. They give me many suggestions, helping to finish the paper on time. Thank for her parents. They give her power to finish the paper. Thanks a lot for everyone.

IX. REFERENCES

[1]S. Papastavrou, G. Samaras, and E. Pitoura. Mobile agents for WWW distributed Database access. In Proceedings of the International Conference on Data Engineering (ICDE), pages 228 -237, Sydney, Australia, March 1999.

[2] Dejan Milojicic. Mobile agent applications. IEEE Concurrency, 7(3): 80-90, July/September 1999.

[3] Simon Pears, Jie Xu and Cornelia Boldyreff, "Mobile Agent Fault Tolerance for Information Retrieval Applications: An Exception Handling Approach", Computer Science, University of Durham, Durham, DH1 3LE, UK.

[4] Brian Brewington, Robert Gray, Katsuhiro Moizumi, David Kotz, George Cybenko and Daniela Rus, "Mobile agents in distributed information retrieval", Thayer School of Engineering, Department of Computer Science, Dartmouth College, 1999.

[5] Harald Kosch, Mario Doller and Laszlo Boszormenyi, "Content-based Indexing and Retrieval supported by Mobile Agent Technology", Institute of Information Technology, University Klagenfurt, Austria.

[6] L. A. Guedes, P. G. Oliveres, L. F. Paina, and E. Cordozo. An agent based approach for supporting quality of service. Computer Communications, 21:1269-1278, 1998.

[7] S. Manvi and P. Venkataram. QoS management by mobile agents in multimedia communication. In IEEE CS Press, editor, Proceedings of the International DEXA 2000 Workshops, pages 407-411, Greenwich, London, UK, September 2000.

[8] White, J. (1996), "Telescript Technology: Mobile Agents," In Software Agents, J. Bradshaw Ed., MIT Press.

[9] Danny B. Lange and Mitsuru Oshima, "Mobile Agents with Java: The Aglet API^{1}".

[10] Lange, D., and Oshima, M, Programming and Deploying Java Mobile Agents with Aglets, Addison-Wesley Longman, Reading, Mass., 1998.

[11] Mark L. Gillenson. 2004. Fundamentals of Database Management Systems. Wiley E-Books. www.wiley.com/

[12] Haroun Rababaah, "DISTRIBUTED DATABASES FUNDAMENTALS AND RESEARCH", Department of Computer and Information Sciences Indiana University South Bend, 2005.

[13] Silberschatz, Korth and Sudarshan, "Database System Concepts".

[14] M. Tamer Özsu and Patrick Valduriez, "Principles of Distributed Database Systems", November 2010.