International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:3, No:7, 2009

Distributed e-Learning System with Client-Server and P2P Hybrid Architecture

Kazunari Meguro, Shinichi Motomura, Takao Kawamura, and Kazunori Sugahara

Abstract—We have developed a distributed asynchronous Web based training system. In order to improve the scalability and robustness of this system, all contents and a function are realized on mobile agents. These agents are distributed to computers, and they can use a Peer to Peer network that modified Content-Addressable Network. In this system, all computers offer the function and exercise by themselves. However, the system that all computers do the same behavior is not realistic. In this paper, as a solution of this issue, we present an e-Learning system that is composed of computers of different participation types. Enabling the computer of different participation types will improve the convenience of the system.

Keywords—Distributed Multimedia Systems, e-Learning, P2P, Mobile Agent

I. Introduction

Asynchronous Web-Based Training systems (hereafter we abbreviate as WBT) are very popular in e-Learning systems. A WBT allows a learner to complete the WBT on his own time and schedule, without live interaction with an instructor. A large number of studies have been made on asynchronous WBT [1] [2] [3], and all of them are based on the client/server model. The features of the client/server model are that all are to execute management and to offer the exercise by the server machine. Although the client/server model has an advantage of easy construction and maintenance, the client/server systems generally lack scalability and robustness. There is Peer to Peer (hereafter we abbreviate as P2P) model to supplement the disadvantage of client/server model. The feature in the system is based on P2P model that each computer works as a client or a server. The feature can distribute the load to a node. The function of the entire system doesn't stop even if some nodes break down.

We have proposed and implemented a distributed e-Learning system based on P2P architecture [4] [5]. The proposed e-Learning system has two distinguishing features. Firstly, it is based on P2P architecture to improve the scalability and robustness of the system. In the proposed e-Learning system, every user's computer plays the role of a client and a server. While a user uses the system, his/her computer (hereafter we refer to such a computer as a node) is a part of the system. The node receives some number of contents from another node when it joins the system and has responsibility to send appropriate contents to the requesting nodes. In addition to the above advantages of using P2P architecture, the proposed e-Learning system can be constructed at low cost because the system need no server computers. Secondly, each exercise in the system is not only data but also an agent so that it has functions, such as scoring user's answers, telling the correct answers, and showing some related information without human instruction. The proposed system equally treats all nodes, and distributes the functions.

However, when the system is composed of nodes which have a low machine specs or use a narrow bandwidth of network, the performance of the entire system is decreased. A node which executes only exercises cannot participate in the system.

In this paper, as a solution for this issue, an e-Learning system that is composed of computers with different participation type is proposed. That is to say, one system is composed with nodes which have same functions like a pure P2P model, another system is composed with nodes which have function or not like a client/server model. A node only offers data and function like a server, and another node only learns. By constructing the system which uses the nodes with only server function, even if the trouble occurs in one of the nodes that compose a server group, the system will continue service. By constructing the system which uses the nodes to perform only learning, a performance of the whole system will be stable.

This paper is organized in 5 sections. The proposed e-Learning system is described in Section 2. The design overviews and implementation of the proposed system are described in Section 3 and the experimental result in Section 4. Finally, some concluding remarks are drawn in Section 5.

II. Proposed e-Learning System

A. Overview

All exercises in the proposed system are classified into categories, such as "Math/Statistic", "English/Grammar", etc. A user can obtain exercises one after another by specifying categories of the required exercises. While a user uses the proposed e-Learning system, his/her node is to be a part of the system. The node receives some number of categories and exercises from another node when it joins the system and has responsibility to send appropriate exercises to requesting nodes.

The important point to note is that the categories a node has are independent of the categories in which the node's user is interested, as shown in Figure 1. Figure 1 illustrates that user A's request is forwarded first to the neighbor node, and the request is forwarded to the node which has the requested category.

K. Meguro is with Ph.D. student in Tottori University.
S. Motomura is Information Media Center Tottori University, T Kawamura and K Sugahara are with the Department of Information and Knowledge Engineering Tottori University.

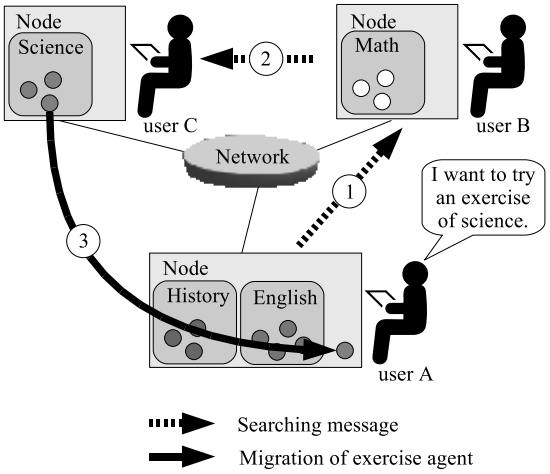

Fig. 1. Migration of exercise agent.

B. P2P network

When the proposed system bootstraps, one initial node has all categories in the system. When another node joins the system, it receives certain number of categories from the initial node. The categories are distributed among all nodes in the system according as nodes join the system or leave the system.

In existing P2P-based file sharing systems, such as Napster [6], Gnutella [7], and Freenet [8], each shared file is owned by a particular node. In the systems, files are originally distributed among all nodes. On the other hand, the categories in the proposed system are originally concentrated. When a new node joins the system, not only location information of a category but the category itself must be handed to the new node. Considering that, the P2P network of the proposed system can be constructed as a CAN [9].

Our P2P network is constructed with 2-dimensional coordinate space [0,1] $\times$ [0,1] to store exercise categories, as shown in Figure 2. The figure shows the situation that node C has just joined the system as the third node. Before node C joins, node A and node B shared the whole coordinate space half and half. At that moment, node A managed “Math/Geometry”, “Math/Statistics”, and “History/Rome” categories and node B managed “English/Grammar”, “English/Reader” and “History/Japan” categories, respectively. When node C joins the system, we assume node C already knows IP addresses of some nodes in the system and node C sends the join request to some node in the list. Then node C is mapped on a certain coordinate space according to a random number and takes on corresponding categories from another node. For example, in the case of Figure 2, node C takes on the “History/Japan” category from node B, then exercises of the category move to node C. After joining, node C gets a list of IP addresses of neighbor nodes in the coordinate space, such as node A and node B. Therefore, neighbor nodes can communicate with each other.

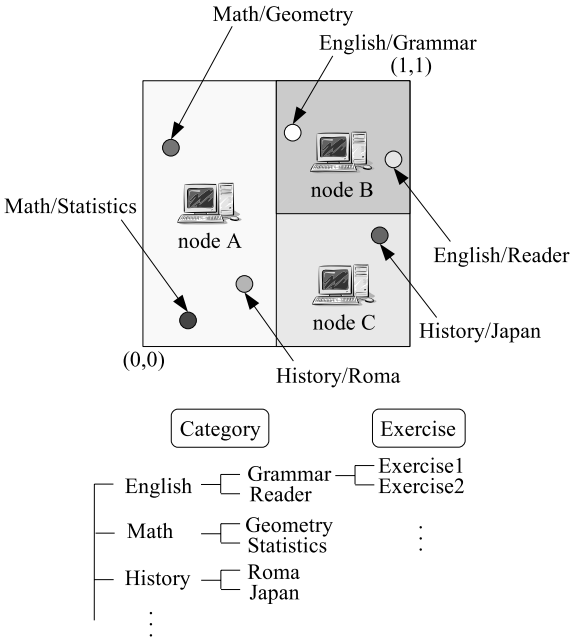

Fig. 2. P2P network of the proposed e-Learning system.

C. Components of system

In order for the proposed system to be considered as a distributed WBT system, it is not enough that only exercises are distributed among all nodes. Functions to provide the above services must be also distributed among all nodes. Mobile agent technology is adopted to achieve this goal.

There are following agents and user interface programs on each node. These agents have implemented in the mobile agent framework that we have developed [10].

- **Node Agent:** Each node has one node agent. It manages the zone information of a CAN and forwards messages to the Category Agents in the node.
- **Exercise Agent:** Each Exercise Agent has questions and functions to score user’s answers, to tell the correct answers, and to show some related information about the exercise.
- **Category Agent:** Each Category Agent stands for a unit of a particular subject. It manages Exercise Agents in itself and sends them to the requesting node.
- **Interface Agent:**There is one interface agent for each user interface, such as a student interface and an exercise manager interface on each node. It plays a role of interfaces between the interface program and other agents, and between agents and applications.
- **User Agent:**Each user has its own User Agent. A User Agent manages its user’s information that includes login name, password, IP address of the user’s computer, online/offline status, and log of studying or a list of created exercises.
- **Student Interface:** One student interface is on each node of which a user logs in as a student. It is a user interface

program for studying.

III. The type of participation node

The following nodes is used in the proposed system.

- **Server Node:**Only data and function are offered.
- **Common Node:**Data and function are offered, it has agent's system requirements, and it is operated directly as a learner and a teacher.
- **Client Node:**Neither data nor function is offered, it has the environment that agent operates, and it is operated directly as a learner and a teacher.
- **User Interface Node:**Neither data nor function is offered, it doesn't have the environment that agent operates, and it is operated directly as a learner and a teacher.

A. Design and Implementation of the Client Node

In the proposed system,the Distributed Hash Table(hereafter we abbreviate as DHT) that does the mapping on the coordinate space two dimensions is used. The area of DHT is given to the node which participates in the system. As a result, the node composes a P2P network, and the contents are managed.

However, the Client Node doesn't manage the contents, and the function of P2P network is used. To realize the participation type which use only the function of P2P network, when the Client Node participates in the P2P network, it should not receive the area of DHT. Instead of not receiving the area of DHT, the Client Node registers the address of the node that participates in the P2P network as a connection destination. The registered node is named the Proxy Node. The Proxy Node is chosen by the following procedures.

- When the Client Node participates in a system for the first time, the Client Node connects with the node that it has already known. The node is named the Known Node.
- The Known Node generates a coordinate.
- A node including the coordinate is chosen as Proxy in the domain to manage of its own.

When the Proxy Node leaves, the inquiry of the Known Node is executed again, and the node that includes coordinate in the area is retrieved, then it registers as a new Proxy Node. When the area of the Proxy Node changes and the area doesn't include the coordinate which is generated by the Known Node, it retrieves by the inquiry of the Known Node, and it registers as a new Proxy Node. The Client Node can use all functions except the content delivery. There is an advantage that there is no influence on the P2P network even if the trouble occurs in the Proxy Node, for the Client Node one-sidedly executes the registration of the Proxy Node.

B. Design and Implementation of the User Interface Node

The Interface Agent intermediates between an agent and a user interface which a learner uses. If the procedure can be remotely called, which is one of the functions of the Interface Agent, users can exercise by using the Student Interface. In other words, to make the User Interface Node participate as a learner, it only has to prepare the Interface Agent in the connected node. Therefore, the Interface Agent for the User Interface Node to exercise is prepared.

In the proposed system, the agent ID is essential to communicate with the agent. The agent ID is necessary to avoid the confusion of the message, it is the same with communication between the user interface and the Interface Agent. In communicating, since the Node Agent is only one in each server, it doesn't need the agent ID. The User Interface Node requests the Node Agent in the Known Node to generate the Interface Agent. Then, the Node Agent generates the Interface Agent, and returns the ID to the User Interface Node. The User Interface Node can exercise by this mechanism.

However, there is one issue that should be thought when User Interface Node is designed. It is the case that the node which generates the Interface Agent leaves. The method of the migration to the node that integrates the DHT areas of the leaving node cannot be realized for two reasons. One of the reason is that the Interface Agent exists only by one or less in one node. The other is that the user interface cannot find the integrated node even if two or more the Interface Agent can exist in one node. In the proposed system, communication to the agent can be established from the user interface, the opposite communication cannot be established. Being not able to establish the communication of the user interface from the agent, it means that the Interface Agent cannot tell the user interface migrating to the integrated node, as a result, the user interface cannot find the Interface Agent.

As a solution of the above issue, the generated Interface Agent is mapped in DHT. When the Interface Agent is mapped in DHT, the user-name is treated as a key. Then the Interface Agent automatically migrates the node according to the change in the area. As the user interface can use the function of the Node Agent as long as the Known Node is identified, it becomes possible to find the Interface Agent that makes the user-name a key and uses the search function of the Node Agent. When the User Interface Node is exercising, the User Agent of a user and the Exercise Agent under exercise are sent to the node where the Interface Agent exists. If the Interface Agent manages the User Agent and the Exercise Agent, these agents can be migrated together. This method makes it possible to continue exercising, since the Interface Agent can be find even if the user interface loses sight of the Interface Agent.

IV. Experiment

This section presents the experimental results for the e-Learning system where the node with different participation type exists together. The following experiments are examined. Table I shows the machine specification.

- Confirming the operation of the system
- Comparison of response speeds by number of participation nodes

A. Confirming the operation of system

The system is constructed "N1:Server Node", "N2:Common Node" and "N3:User Interface Node". TableII shows the experimental conditions. In the experiment, it aims to confirm the following.

TABLE I
THE MACHINE SPECIFICATION.

CPU	Intel Pentium4 3.0GHz
Memory	1GB
Network	1000Base-T
OS	TurboLinux 10 Desktop

- When the node with a different function in itself exists, the category and the exercise can be made and can be acquired.
- When the User Interface Node participates in the system, the Interface Agent corresponding to it can be made.
- Even if the proxy node leaves, the User Interface Node can be connected with a new proxy node.

TABLE II
THE EXPERIMENTAL CONDITIONS.

Number of node	N1:49, N2:1, N3:5
Number of group	50
Number of user account	student:50, teacher:3
Number of category	20
Number of exercise	60

Figure3 shows the appearance of area of DHT divided with the node that participates, where the Category Agent and the Interface Agent are made corresponding to a learner. The P2P network is constructed according to the following procedures.

1) First of all, nines node of the N1 and one node of the N2 participate in the system, then forty node of the N1 participate. Each node is named node 1, node 2, ..., and node 50 in the order of participation respectively.
2) On node 10 that is the node of N2, the system is logged in as a teacher by using the user interface for the teacher.
3) The teacher makes four subjects of English, Math, Biology and Physics, and makes five categories for each subject, and makes three exercises for each category.
4) The teacher logs out once by the system after executing these operations, and logs in the system again as a teacher.

As a result, the categories and the exercises are added to this system normally. When the teacher logs in the system again, the categories and the exercises are found in the system.

Then, the P2P network is constructed according to the following procedures.

1) Five node of the N3 participate in the system, then the learners log in each node through the Student Interface. Each node is named node 51, node 52, ..., and node55 in the order of participation respectively.
2) After logging in the system, each learner requests the teacher who has already logged it in to send the exercise.
3) Node 7 in which the Interface Agent corresponding to node 51 and node 53 exists, after it leaves from the system, these learners request the exercises.

As shown in Figure 3, the Interface Agent corresponding to each learner is made on node 5, node 7, node 9, and node 12 respectively. The category name of the registered exercises are obtained respectively, and the category is displayed on the interface of the learner. When the corresponding Interface Agent left, the learners can continue studying.

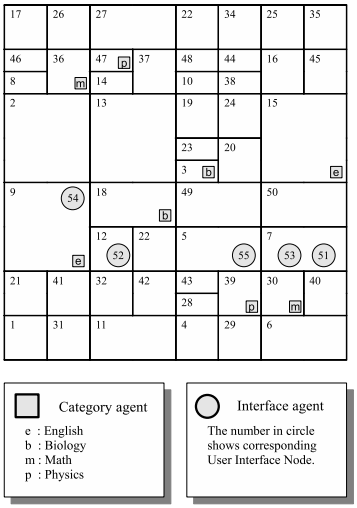

Fig. 3. Division of management area and the distributed agents.

B. Comparison of the response time corresponding to number of the Server Node

The system is constructed “N1:Server Node” and “N3:User Interface Node”. It aims the examination of the number of Server Node of the relation to response time. The response time in this experiment means the time which takes to display after the learner request the exercise on Student Interface

The number of nodes of the N1 is 1, 2, 3, 6, 12, and 20. The number of nodes of the N2 is 5. The average of response time is measured repeating five times to acquire the exercises from all nodes at the same time. The average times are summarized in Figure 4.

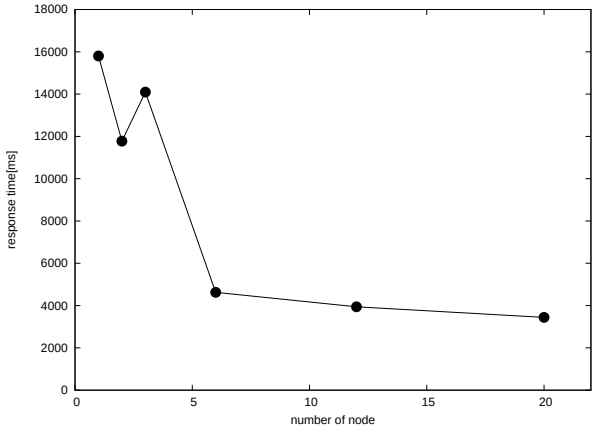

Fig. 4. Mean time of response.

The case where the node of the N1 is three is excluded, as the number of nodes increases, the response time shortens. When the number of nodes of the N1 becomes 6 or more, the width of a decrease at response time becomes small.

V. Conclusion

A distributed e-Learning system in which different use types can participate has proposed. The kind of the nodes which are implemented the proposed system have been “only participate in the P2P network as a server”, “participate in the P2P network and participate as a learner”, “only participate as a learner with agent’s system requirements” and “only use as a learner”.

Increasing the number of nodes that participate as a server, it not only has become easy to maintain the node that composes the server group but also has distributed the load to each node that composes the P2P network, and the response of the entire P2P network has improved.

However, in the kind of a node “only use as a learner”, even if the node where the Interface Agent was mapped doesn’t leave, the Interface Agent migrate with the change of the area of DHT, then the Interface Agent is retrieved again. Therefore, in the e-Learning system which the area of DHT changes frequently, the node cannot exercise efficiently . In future work, we should verify the usability with a larger-scale system.

References

[1] Nishita, T. and et el,: Development of a Web Based Training system and Courseware for Advanced Computer Graphics Courses Enhanced by Interactive Java Applets, *Proceedings of International Conference on Geometry and Graphics*, Vol. 2, pp. 123-128 (2002).
[2] Homma, H. and Aoki, Y.: Creation of WBT Server on Digital Signal Processing, *Proceedings of 4th International Conference on Information Technology Based Higher Education and Training*, (2003). Marrakech, Morocco.
[3] Helic, D., Krottmaier, H., Maurer, H. and Scerbakov, N.: Enabling Project-Based Learning in WBT Systems, *International Journal on E-Learning*, Vol. 4, No. 4, pp. 445-461 (2005). Norfolk, VA.
[4] Kawamura, T. and Sugahara, K.: A Mobile Agent-Based P2P e-Learning System, *IPSJ Journal*, Vol. 46, No. 1, pp. 222-225 (2005).
[5] Motomura, S., Nakatani, R., Kawamura, T. and Sugahara, K.: Distributed e-Learning System Using P2P Technology, *Proceedings of the 2nd International Conference on Web Information Systems and Technologies*, pp.250-255 (2006). Setubal, Portugal.
[6] Napster, http://www.napster.com (1999).
[7] Gnutella, http://welcome.to/gnutella/ (2000).
[8] Clarke, I., Sandberg, O,. Wiley, B. and Hong, T. W.: Freenet: A Distributed Anonymous Information Storage and Retrieval System, http://freenetproject.org/papers/freenet.pdf (2000).
[9] Ratnasamy, S., Francis, P., Handley, M., Karp, R. and Schenker, S.: A scalable content-addressable network, *Proceedings of the 2001 conference on applications, technologies, architectures, and protocols for computer communications*, ACM Press, pp. 161-172 (2001).
[10] Motomura, S., Kawamura, T. and Sugahara, K.: Logic-Based Mobile Agent Framework with a Concept of Field , *IPSJ Journal*, Vol. 47, No. 4, pp. 1230-1238 (2006).

Kazunari Meguro was born in 1977. He received his B.Eng. and M.Eng. degrees from Hiroshima city University, Japan, in 2002 and 2004, respectively. He is currently a Ph.D. student in Tottori University. His research interests multi-agent systems and distributed systems.

Shinichi Motomura was born in 1973. He received his B.Eng. and M.Eng. degrees in Computer Engineering from Toyohashi University of Technology, Japan, in 1995, 1997, respectively. From 2004 to 2008, he joined Tottori University as a research associate. In 2008, he received the D.Eng. degree from Tottori University, Japan. Since 2008 he had been in Tottori University as an associate professor in the Information Media Center. His research interests include multi-agent systems and distributed systems.

Takao Kawamura was born in 1965. He obtained his B.Eng., M.Eng. and Ph.D. degrees in Computer Engineering from Kobe University, Japan, in 1988, 1990 and 2002, respectively. Since 1994 he had been in Tottori University as a research associate and has been in the same University as an associate professor in the Faculty of Engineering since 2003. His current research interests include mobile-agent systems and distributed systems. Dr. Kawamura is a member of IPSJ and IEICE.

Kazunori Sugahara received the B.Eng. degree from Yamanashi University, Japan, in 1979 and M.Eng. degree from Tokyo Institute of Technology, Japan, in 1981. In 1989, he received the D.Eng. degree from Kobe University, Japan. From 1981 to 1994, he was on the staff of the Department of Electronic Engineering, Kobe City College of Technology. In 1994, he joined Tottori University as an associate professor of the Department of Electrical and Electronic Engineering and he is a professor of the department of Information and Knowledge Engineering. His current interest lies in the fields of computer architectures and hardware realizations of image processing algorithms. Prof. Sugahara is a member of IEEE, IPSJ and IEICE.