SICE Annual Conference 2010
August 18-21, 2010, The Grand Hotel, Taipei, Taiwan

# Development of User Interface Supporting Multi Web Browsers for Distributed e-Learning System

Takashi Hirata, Takao Kawamura,
Toshihiko Sasama and Kazunori Sugahara
Graduate School of Engineering Tottori University
4-101,Koyama-Minami Tottori,JAPAN
Email: {s052052,kawamura,sasama,sugahara}@ike.tottori-u.ac.jp

Kazunari Meguro
Information Media Center Tottori University
4-101,Koyama-Minami Tottori,JAPAN
Email: meguro@tottori-u.ac.jp

***Abstract*—We have proposed and implemented a distributed e-Learning system. The user interface to communicate between agent and user's browsers is necessary in this system. And our old user interface analyses data and generate HTML on the browser-side. Therefore, there is a problem of depending on a specific browser. In this study, as a solution for this problem, we propose and implement the user interface that analyses data and generates HTML on the agent-side. As a result, the user can use this system in a favorite environment.**

***Keywords*—P2P, Mobile Agent, e-Learning.**

## I. Introduction

We have proposed and implemented a distributed e-Learning system [1] using Maglog that is a Prolog-based framework for building mobile multi-agent systems we have also developed[2]. In order to improve the scalability and robustness of this system, all exercises and functions, such as scores user's answers are realized on mobile agents. These agents are distributed to computers, and they constructed with a P2P network that modified Content-Addressable Network (CAN)[3]. In this paper, we present a user interface that support multi web browsers for the proposed system. The proposed system is a distributed system that used mobile agents. Therefore, a user interface for communication is necessary between the browser of user and the mobile agent. Maglog has the function of the XML-RPC server, and can communicate with mobile agents by XML-RPC. Therefore, we implemnted this user interface as a Mozilla Firefox extension because we needed a browser to be able to interpret XML-RPC. This user interface analyses data and generate HTML on the browser-side. Therefore, there is a problem of depending on a specific browser. We intend to achieve a user interface that solves this problem. Then, we propose and implement the user interface that analyses data and generate HTML on the agent-side. Concretely, we implement the user interface on the agent server that is agent's runtime environment. As a result, the user can learn by access to URL from web browser like concentrated type system. Therefore, the proposed system becomes a more ideal distributed system.

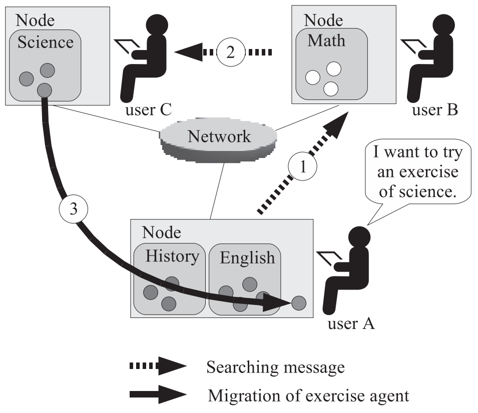


Fig. 1. Proposed e-Learning system.

## II. Proposed e-Learning System

While a user uses the proposed e-Learning system, his/her computer is a part of the system. Namely, it 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 is that nodes that joins in the system share categories as shown in Fig. 1. Fig. 1 illustrates that user A's request is forwarded first to the neighbor node, next forwarded to the node which has the requested category. As a result, exercise agent moves to the node of A, and offers A the exercise through the user interface.

## III. Architecture of the User Interface

The proposed system implements by mobile agent framework Maglog. Maglog prepares the data exchange area that is called a field for communication among agents. And, the agent server has the function of the XML-RPC server. Our old user interface is implemented as a Mozilla Firefox extension. This user interface analyses data given by the mobile agent and generate document written in the HTML format. And we develop this user interface by using XUL. Each function of the old user interface is described with JavaScript. Because JavaScript is the only language that can be called from

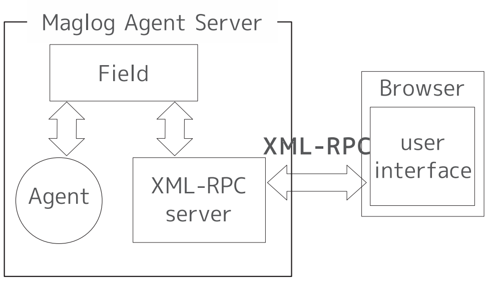


Fig. 2. Concepts for communication in the previous system.

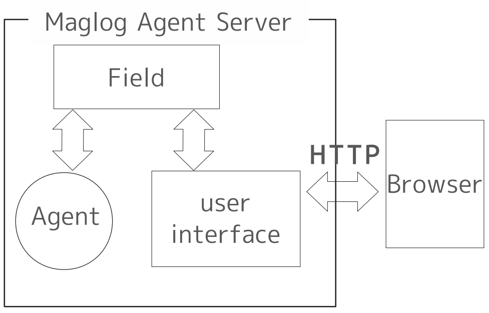
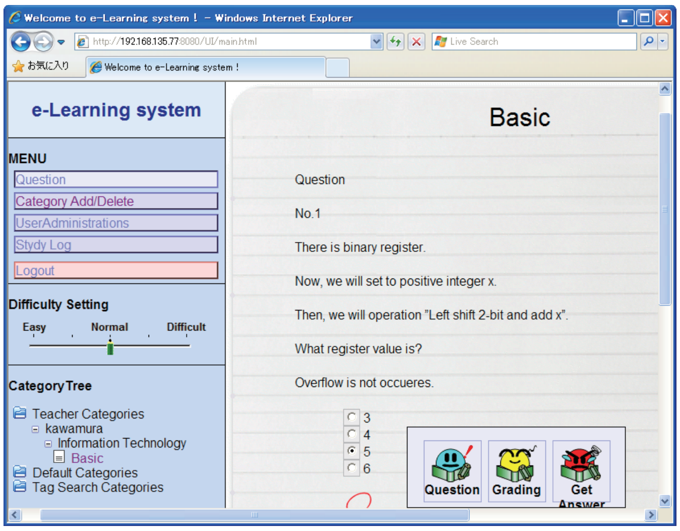


Fig. 3. Concepts for communication in the proposed system.

XUL. However, JavaScript cannot directly access the field of Maglog. Therefore, the access to the field used XML-RPC as shown in Fig. 2. However, XML-RPC is not suitable for the sending and receiving of mass binary data because of the XML encoding. Moreover, there is a problem of depending on a specific web browser for the communication method shown in Fig. 2. The reason is that web browser that can interpret XML-RPC is needed. Then, we propose the user interface that analyses data and generate HTML on the agent-side. And we change the communication method from XML-RPC to HTTP. Then, we implement the user interface with the function of the web server in the agent server by using Java Servlet as shown in Fig. 3. The servlet runtime environment of the agent server can use the Java class that can access the field of Maglog. The user interface is directly accessed to the field of Maglog without the XML encoding by using it. Therefore, the response speed in the user interface quickens. We improve the environment that the user can learn by the HTTP access. In other words,the user can learn with a favorite web browser.

## IV. View Part of the User Interface

The user interface treats contents that change dynamically including the function to tell the user grading results etc. Thus, the user interface needs the mechanism to generate document written in the HTML format dynamically. Above the reason and because it can directly access field of Maglog, the user interface is developed by Java Servlet. We intend to enable learn by a mobile device with a web browser that doesn't support JavaScript. Therefore, we implement two kinds of view parts for a browser which support JavaScript and do not support it. Fig. 4 shows a screenshot of the user interface for a web browser which support JavaScript. The reason to support JavaScript is to achieve the user interface that uses



Basic

Question

No.1

There is binary register.

Now, we will set to positive integer x.

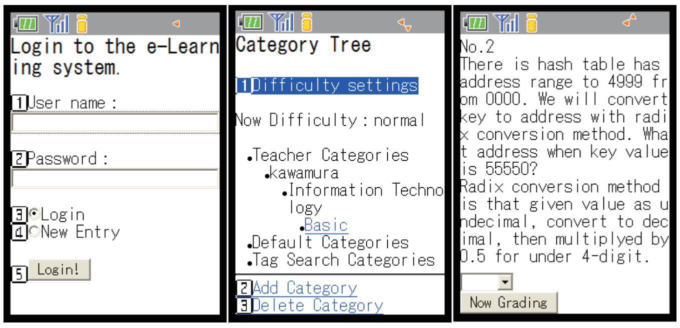
Then, we will operation "Left shift 2-bit and add x".

What register value is?

Overflow is not occueres.

3
4
[✓ 5]
6

Fig. 4. The screenshot of the user interface.



Login to the e-Learning system.

1 User name :

2 Password :

3 [✓ Login]
4 New Entry

5 Login!

Category Tree

1 Difficulty settings

Now Difficulty : normal

- Teacher Categories
  - kawamura
    - Information Technology
      - Basic
- Default Categories
- Tag Search Categories

2 Add Category
3 Delete Category

No.2
There is hash table has address range to 4999 from 0000. We will convert key to address with radix conversion method. What address when key value is 55550?
Radix conversion method is that given value as undecimal, convert to decimal, then multiplyed by 0.5 for under 4-digit.

Now Grading

Fig. 5. Screenshots of the user interface for mobile device.

Ajax. It is necessary to change the entire page if Ajax is not used. It increases the network load. Therefore, we suppress the network load by rewriting only a part of a page by using Ajax. Fig. 5 shows screenshots of the user interface for a web browser which do not support JavaScript. It suppressed the load of packet transmission by reduction of display menu item. In addition, it is easy to use the user interface because the shortening access is possible.

## V. Conclusion

In this study, we have developed the user interface that support multi web browsers. As a result, the user can use the proposed system in a favorite environment. We expect the proposed system to become more popular by the user interface.

## References

[1] T. Kawamura and K. Sugahara, "A mobile agent-based p2p e-learning system," *IPSJ Journal*, vol. 46, no. 1, pp. 222–225, 1 2005.

[2] S. Motomura, T. Kawamura, and K. Sugahara, "Logic-based mobile agent framework with a concept of "field"," *IPSJ Journal*, vol. 47, no. 4, pp. 1230–1238, 4 2006.

[3] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Schenker, "A scalable content-addressable network," in *Proceedings of the 2001 conference on applications, technologies, architectures, and protocols for computer communications*. ACM Press, 2001, pp. 161–172. [Online]. Available: citeseer.ist.psu.edu/article/ratnasamy01scalable.html