1998 International Computer Symposium
Workshop on Computer Graphics and Virtual Reality
December 17-19, 1998, N.C.K.U., Tainan, Taiwan, R.O.C.

COLLABORATIVE ACTIVE TEXTBOOKS: ALGORITHMS ANIMATION ON THE WEB

Yi-Hsing Chang, and Rong Chen
Department of Information Management,
Nantai Institute of Technology, Tainan County, Taiwan, R.O.C
Email:yhchang@nantai.ntc.edu.tw.

Abstract

The paper describes the design and implementation of a java-based system, JCATA of building Web-based collaborative active textbooks on algorithms. JCATA augments the power of Web pages for publishing passive multimedia information with an interactive algorithm animation system. In JCATA: Teachers can join the system and teach different topics. Students can select one topic or one teacher and form a teaching group. In a teaching group, the teacher and students can teach and discuss synchronously. A demonstration panel is provided to demonstrate the algorithms animatedly. In addition, A test panel is also provided to clients. Then they can exercise the examples of algorithms by themselves. Our experimental results show that a java-based system of building Web-based collaborative active textbooks on algorithms can achieve good learning effect for distance education.

Keywords: Java, distance education, animation, algorithm, World Wide Web

1. Introduction

During the recent years, Web technologies-in particular linked Java server and clients- has been growing at an amazing pace. The million use of the WWW for information exchange has led to the development of a cottage industry for producing Web-based applications. Especially, the applications on education are growing rapidly. The Web provides rich information content (text, image and video etc.), attractive and intuitive interface, and platform independence (universal solution to GUI). However, the traditional WWW is actually a passive method of accessing and displaying information. It is still hard to design a collaborative environment on this technology, since such environment requires high interaction. Coupled with the advent of Java introduced by Sun Microsystems in 1995, WWW becomes an active document capable of securely executing code distributed to the clients from a WWW server [1]. The Java byte-code can be executed on different platforms without modification.

2. Previous Work

Many systems applied in different area are developed recently based upon WWW for education. In [12], a world wide web course center(W3CC) is developed. Many functions are provided in W3CC such as chat, class discussion, etc.. But, the course contexts are edited with HTML. Thus, it can not achieve the synchronous teaching and animated effect. The same situation occurred in these systems[2,3,4,6,7, 9,10,11]. Systems developed in [5,8] are java-based collaborative active textbooks on algorithms. But these systems do not support the ability of synchronous message communication, demonstration and test.

In this paper, JCATA, a java-based system of building Web-based collaborative active textbooks on algorithms are designed and implemented. It owns the functions of synchronous and asynchronous teaching. In addition, test and synchronous communication are also designed in the system. In the following sections, the design and implementation of the system are given. Our experimental results show that a java-based system of building Web-based collaborative active textbooks on algorithms can achieve good learning effect for distance education.

3. The design of the JCATA

The system is designed in a client-server architecture as shown in Figure 1.

In the client site: Clients can connect the JCATA when they key in the I.P. address of the server in their browsers. When clients call the remote objects, they in fact call the stub program resided in clients. The purpose of the stub program is to find the address of remote object, pass and convert the parameters between clients and the server.

In the server site: Skeleton program receives the parameters passed from clients and converts them into correct format of the machine. It then calls the remote object to execute the request and get the results. The results are then converted into original format and send back to the clients. Finally, the stub program in clients receives the results and passes them to the clients after the results are converted into the correct format of browsers.

Under the architecture, the interface and functions of JCATA are designed as shown in Figure 2 and Figure 3. In Figure 2, there is a menu listing the kinds of algorithms. After a kind of algorithm is selected, the following window in Figure 3 is shown.

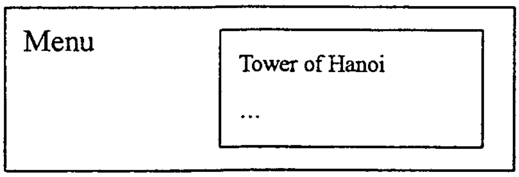


Figure 2

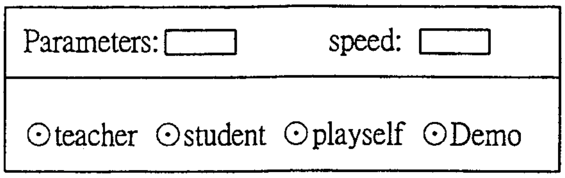


Figure 3

The window in Figure 3 is divided into two windows:

(1) The top window: Two kinds of data are listed in this window. One kind of data is the parameters needed for the algorithm and the other controls the speed of the algorithm.

(2) The bottom window: there are four items listed in the window for clients to choose.

(A) Teacher: when it is chosen, it means that the client wants to play the roll of a teacher. The following functions are implemented in it.

(a) Name list: the students joining the teacher are listed. These students and the teacher form a teaching group.

(b) Synchronous discussion: he can synchronously send message to all or specific students in the teaching group.

(c) Synchronous teaching: when a teacher is teaching, the animation on his screen sends synchronously to the screens of all students in the teaching group.

(B) Student: when it is chosen, it means that the client wants to play the roll of a student. The following functions are implemented in it.

(a) Name list: the teachers joining the system are listed. He then can choose a teacher to join and the student waits for teaching.

(b) Synchronous discussion: he can synchronously send message to the teacher.

(C) Demo: when it is chosen, the algorithm animation is shown in the screen of students. From this, he can asynchronously learn the algorithms.

(D) Playself: when it is chosen, a test of the algorithm is presented. The client can actually play the example and test whether he understands the algorithm. Note that, during the process of playing, the system automatically backs to the previous state when a mistake is taken.

4. The implementation of JCATA

The algorithm, tower of Hanoi is preliminarily implemented in JCATA. The programs needed to implement such a system are shown in the following. Note that, Each program is implemented as an applet. The communication between applets is implemented based upon java's remote method invocation(RMI) since it allows applets on different machines to communicate with each other.

(1) message.java: is a remote interface.

(2) server.java: is a main program in a server that passes the parameters between clients and the server.

(3) HanoiTeaching.java: the main applet window that supplies four items in Figure 3 for clients to choose.

(4) photo.java: is a graphic window embedded in the web page of clients.

(5) ti.java: is a remote interface.

(6) t.java: the main program to generate the screen of a teacher. It provides the functions that the teacher can synchronously send message to students and teach.

(7) si.java: is a remote interface

(8) s.java: the main progam to generate the screen of a student. It provides the functions that students can synchronously send message to a teacher and accept teaching.

(9) teachhanoi.java: is a teaching program of tower of Hanoi in the teacher site.

(10)studhanoi.java: is a teaching program of tower of Hanoi in the student site.

(11)hanoi.java: is an animatedly demonstrative program of tower of Hanoi.

(12)handhanoi.java: is an animatedly teaching program for clients to play by themself.

The structure of classes is listed in Figure 4.

5. Executing Results

In the following, we demonstrate the executing steps and some results of the system. The system is executed in the window95's environment and JDK1.1.5 is used to compile and execute programs.

Step 1: Execute the register program in the DOS mode: start rmiregistry. The system is then into a waiting state as shown in Figure 5.

Figure 5

Step 2: Execute the following server program on the same platform in the DOS mode: start java server. It then appears the message that the server is starting as shown in Figure 6. It means that the server is starting.

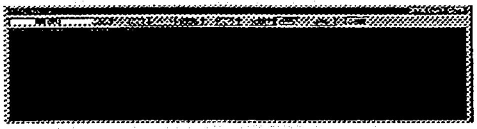
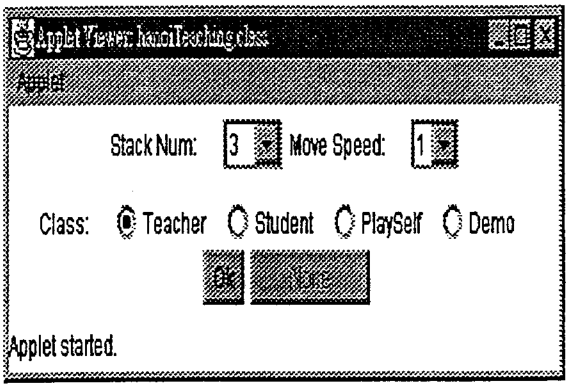


Figure 6

Step 3: Use any browser that supports the Sun RMI to connect the homepage of the system. For example, the instruction, appletviewer http://192.192.214.194/title.htm, is executed and the result is shown in Figure 7, where 192.192.214.194 is the I.P. address of the server. These items in Figure 7 are explained in the following.



Stack Num: 3 Move Speed: 1

Class: ● Teacher ○ Student ○ PlaySelf ○ Demo

Applet started.

Figure 7

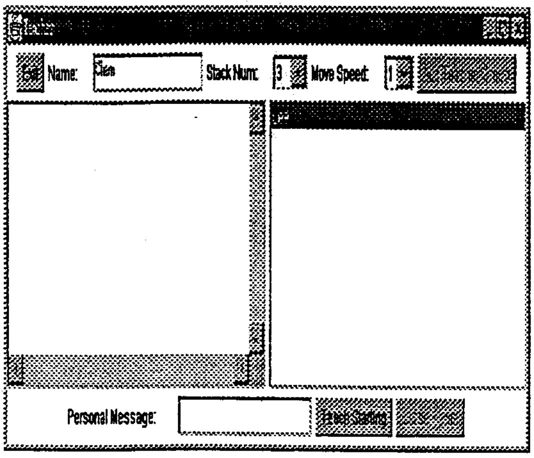
(A) Stack Num: is a popup menu. The number of disks is chosen from here.

(B) Move Speed: is a popup menu. The moving speed of disks is controlled from here.

(C) Class: there is four items to choose.

(a) Teacher: when the button is chosen, the window shown in Figure 8 is appeared. The items are explained in the following.

(1) Exit: exit the teaching.



Name: Chen

Stack Num: 3

Move Speed: 1

Figure 8

(2) Name: input the teacher's name. The teacher, Chen joins the system in the example.

(3) Add Teacher Group: join teaching.

(4) Left window: display the message.

(5) Right window: the students that join the teacher are listed here. The student, Lee joins the teacher in this example.

(6) Broadcast Message: send message to students from here. Note that: the specific students can be selected from right window and just send message to these students. The prompt is then changed to "Personal Message". The prompt is "Personal Message" in this example because specific student Lee is selected.

(7) Teach starting: open the teaching window of the teacher and students in the teaching group. Then, the synchronously animated teaching of tower of Hanoi can be started. The results are shown in Figure 9 and Figure 10 and both of the windows are the same.

(8) Close Hanoi: close the teaching windows of the teacher and students.

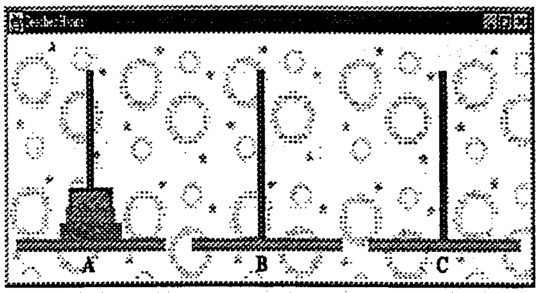


Figure 9 the teaching window of the teacher

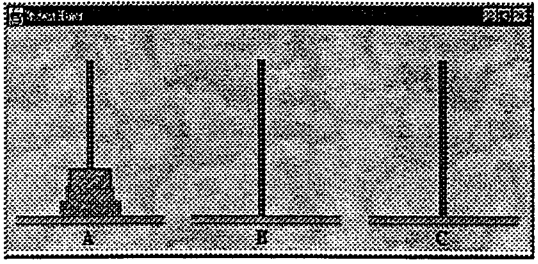
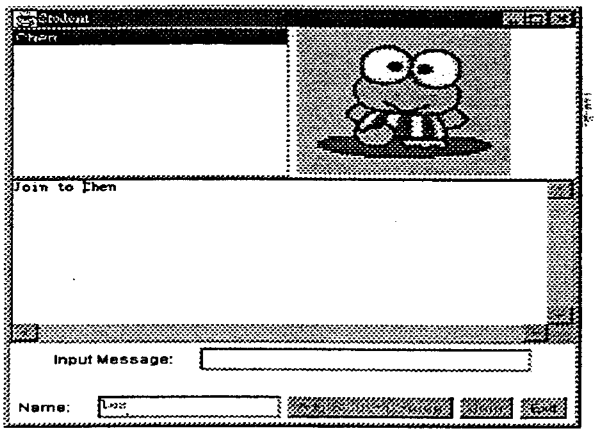


Figure 10 the teaching window of a student in the same teaching group

(b) Student: when the button is chosen, the window shown in Figure 11 is appeared. The items in it are explained in the following.



Join to Chen

Input Message:

Name: Lee

Figure 11

(1) the top window: lists the teachers which join teaching.

(2) the bottom window: displays the message. Chen joins teaching in this example

(3) Input Message: send message to the teacher.

(4) Name: input the student's name. Student Lee joins the teaching.

(5) Add Student Group: join the group of students.

(6) Join: join the teacher selected from the

name listed in the top window. Note that, a student can not join a teacher when the teacher is teaching. The result is shown in Figure 12.

(7) Exit: exit the teaching group.

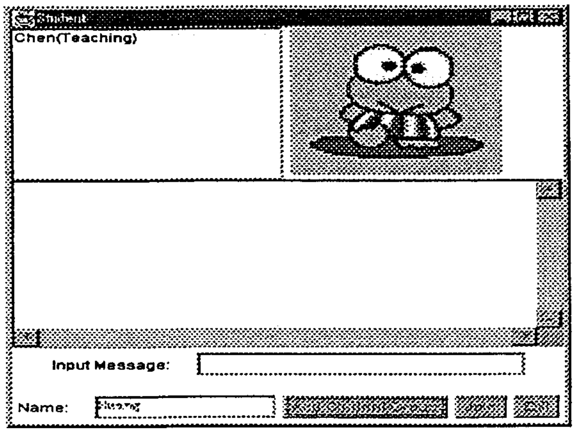


Figure 12

(c) PlaySelf: when the button is chosen, the animated teaching applet as shown in Figure 13 is starting. Each client can move the disks by himself. A disk will return to the original place if the movement of it is wrong.

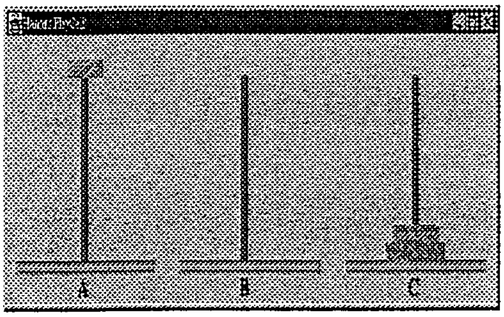


Figure 13

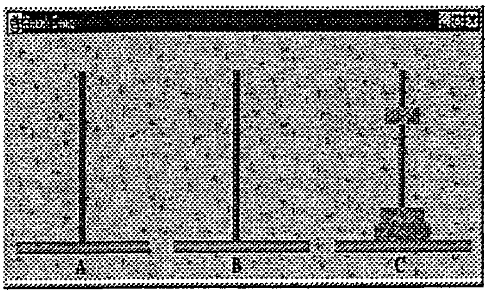


Figure 14

(d) Demo: when the button is chosen, the animatedly demonstrative applet as shown in Figure 14 is starting. It will show all the steps of the algorithm in a correct manner.

6. Conclusion and Future works

This paper presents a system, JCATA, a java-based system of building Web-based collaborative active textbooks on algorithms. Our experimental results show that a java-based system of building Web-based collaborative active textbooks on algorithms can achieve good learning effect for distance education.

In future, the functions of JCATA will be extended to form a complete teaching environment on algorithms such as identification, homework judgement, history management, etc.. Load balancing will also be considered when the load of the server is too heavy.

References:

[1] Arthur, Van Hoff, Sami Shaio and Orca Starbuck, Hooked on Java, Sun Microsystems Press, 1995.

[2] Chen, F.C. and Chiu, C.H., The Coinstructivistic Hypermedia-Aided-Instruction Courseware Design Based on the WWW, the 7th International Conference on Computer-Assisted Instruction(ICCAI), Vol.2, pp.182-185, 1998.

[3] Di, M., En., Z.H., Chen. G. and Chien, B.C., Design and Implementation of an Integrated Virtual Classroom, the 7th ICCAI, Vol.1, pp. 287-294, 1998.

[4] Hong, J.S., The Development of An English Sentence Database and WWW Query Interface for Scientific Paper Writing, the 7th ICCAI, Vol.1, pp.322-327, 1998.

[5] Ibrahim, B., World-wide Algorithm Animation, Proceedings of the First International on the World Wide Web, Paper NO:32, CERN, Geneva, Switzerland, 1994.

[6] Liaw, S.Y., The Development of Multimedia CAI System Applied in Environmental

Protection, the 7th ICCAI, Vol.2, pp.149-153, 1998.

[7] Ma, H., A Java-Based Learning on Demand System, Proceedings of 1998 Workshop on Distributed System Technologies and Application, pp.53-59, 1998.

[8] Marc, H.B. and Roope, R., JCAT:Collaborative active textbooks using Java, Computer Networks and ISDN Systems, vol. 29, pp.1577-1586, 1997.

[9] Shyu, S.J., J., S.S., S., C.H., J., S.S., L, J.J., Chang, H.C. and Wang, L.T., A Virtual classroom for the fire art education in the Internet, the 7th ICCAI, Vol.2, pp.44-51, 1998.

[10] Wang, F.H., An Intelligent learning-support environment for geographic climates exploration on WWW, the 7th ICCAI, Vol.1, pp.468-475, 1998.

[11] Yang, C.C., Development of the Soil Testing Course for Teaching and Application on World Wide Web, ICCAI, pp.295-302, 1998.

[12] Yuan, S.M., The Discussion Environment of WWW-based distant cooperative learning, 1997.

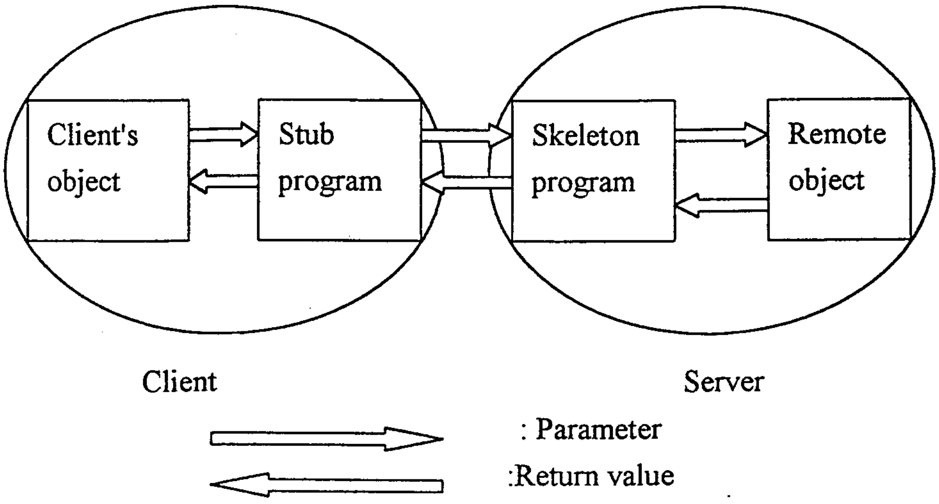


Figure 1

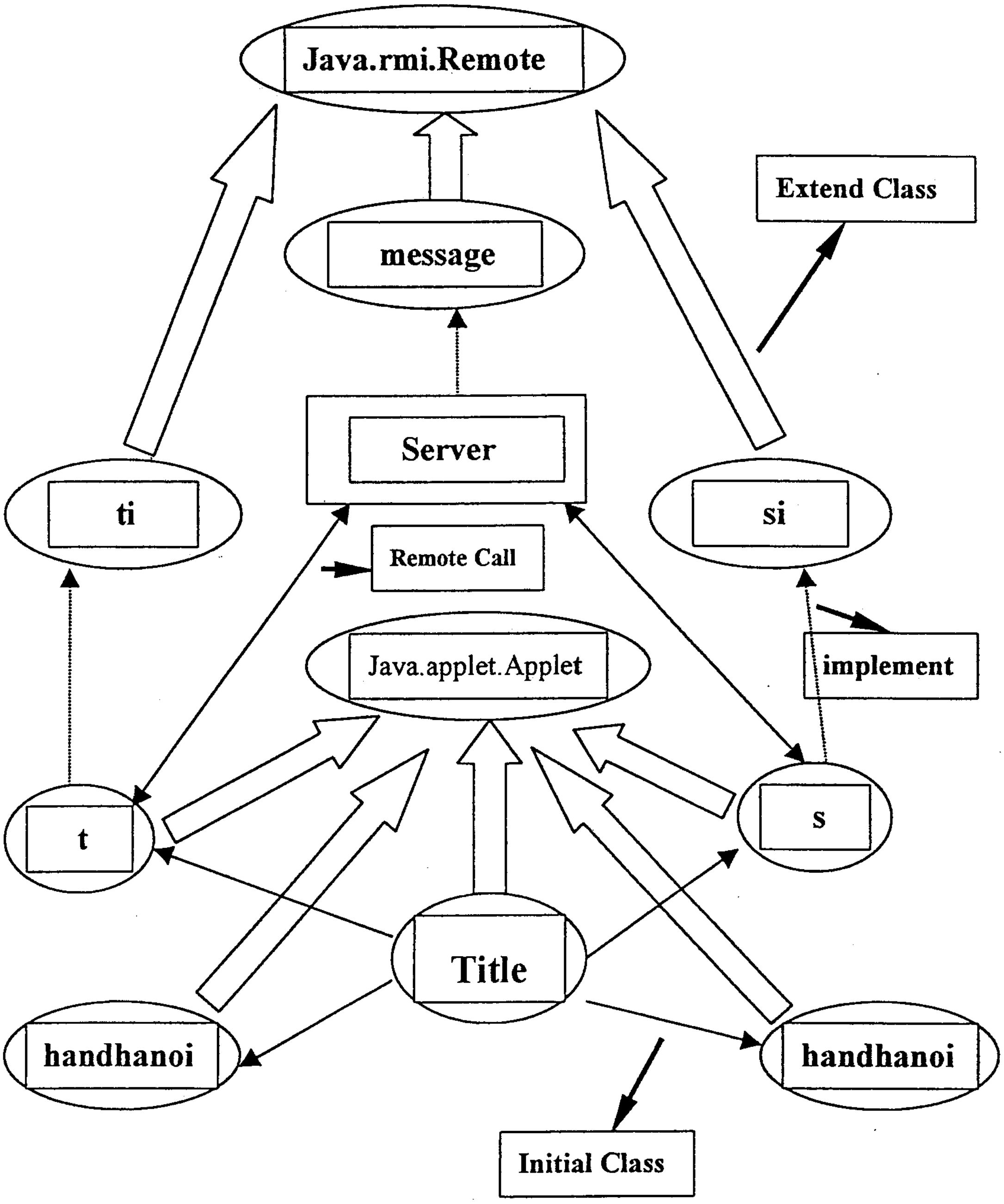


Figure 4 the class structure