

An Object, Session, and Room Persistence in VCR for Enhancing Online Learning Effectiveness

Runhe Huang and Jianhua Ma

Faculty of Computer and Information Sciences, Hosei University,
3-7-2 Kajino Cho, Koganei Shi, Tokyo 184-8584, Japan

rhuang@k.hosei.ac.jp, jianhua@k.hosei.ac.jp

Abstract

This paper explains what is an object, a session, and a room persistence in a virtual collaboration room, describes functions of object recording/replaying, session recording/replaying, and object and room persistence, shows how to implement the functions in VCR system, and finally demonstrates by examples how the functions are used to solve schedule conflicting problem in a synchronous group activity and to support asynchronous group activities as well. Online learning is improved with the functions in terms of effectiveness and efficiency.

1. Introduction

With rapid developments of the Internet and multimedia technologies, online learning in distance education plays an important role. A good online learning system not only allows students to read teaching materials placed in a specified web site, but also should support a variety of group activities associated with learning, such as question time (students can ask their teacher or tutor interactively), discussion time (students with or without their teacher are holding on an interactive discussion on a specified topic related learning contents), team work (a group of students are working together on a term project), and so on. VCR (virtual collaboration room) [1] in University21 [2] is aimed at supporting such group activities. Group activities can be either interactive synchronous or asynchronous style. Most groupware systems [3-6] can support one of the styles with a limited number of objects. However, with adding functions related to object, session, and room persistence in VCR, it becomes possible to support both working styles. Especially for those who are late to join to or absent from a group activity do not have to miss the activity, since they can replay the whole process recorded during the group activity. With the function of room persistence, the group activity can be continued from the previous state kept to the next group activity time. Learning in multiple ways, of course, would be more efficient and effective.

VCR is a desktop groupware system that is designed under the framework of using “object” metaphor and “room” metaphor. By emulating real world’s objects and rooms, VCR makes users feel natural and immersed as they are collaborating in the real rooms and using virtual objects like real objects. Although there are some other groupware systems are also designed based on the terms, “object” or “room”, but most of them [7] are simply using the concepts rather than implementing their physical and social characteristics. However, our VCR is designed under emphasis on clarifying physical and social characteristics of virtual objects and virtual rooms and putting the characteristics into implementations. The object recording and session recording in VCR become very much similar to a video camera recording of the process of an object being used and the process of a number of objects being used, respectively. The room persistence is to keep the virtual room’s current state. The object and session replaying is just like a video camera play back to show the whole recorded process of an object or objects being used.

Object, session, room persistence are concepts in VCR and VCR is one of the important components of a cyber-campus in University21. Therefore, this paper will first introduce University21 including its design ideas and system architecture and then describe VCR in section 2 and 3, respectively. Section 4 will be focused on explaining what are object, session, and object and room persistence in a VCR room and showing how the functions of recording/replaying or persisting of an object, a session, and room work. Some examples are given in section 5 to demonstrate how the functions are used to improve learning effectiveness and efficiency. Finally conclusions will be drawn and future work will be addressed in the last section.

2. University21

University21 was designed under the underlying objective: supporting all teaching, learning and administrating activities for the Internet and Web based virtual universities of various scales. It, therefore, is of the characteristics of accessibility (from any platforms), scalability (from a

smaller to a larger), flexibility (for different scale, type, style of activities), extendibility (of updating/adding new functions), and personality (for personalized environment and contents).

As a physical university, a virtual university created under University21 framework consists of four basic kinds of elements: people, cyber-campus, facilities, and agents [8]. The cyber-campus is a virtual site where people are conducting their teaching, learning and administrating work. It consists of five types of rooms: *virtual private office*, *virtual administration office*, *virtual courseroom*, *virtual collaboration room*, and *virtual laboratory*.

A private office is an individual work surface that is owned by an individual for managing personal activities and private data in a virtual university. A virtual administration office is designed and incorporated with online administrative tools for managing users, curriculum, research and systems. A virtual laboratory is designed for students to conduct course related experiments and simulations via networks. The virtual courseroom [9, 10], a core unit in University21, is an Internet and WWW based environment incorporating with many software tools for systematically supporting teachers to author online materials, organize teaching activities and instruct students, and

students to access the online courses, do an exercise/test/exam and efficiently interact with a teacher and other students. Once a new online course is to be created, a corresponding virtual courseroom will be automatically assigned to the course. A virtual collaboration room is closely connected with each virtual courseroom for supporting a course oriented collaborative activities [11] such as class meeting, tele-lecturing, or group team project, etc. It supports non-course oriented group activities as well such as graduation research, open research seminar, or team research project. In a word, the objective of the virtual collaboration room is to support teaching, learning, and research related all possible group activities. How to design such a system that can support various kinds of group activities over the Internet, it will be explained in the next section.

3. VCR

The VCR is designed under a framework of using both a room metaphor and an object metaphor, i.e., emulating fundamental characteristics of real rooms and real objects, respectively. The layout of a VCR is given in Figure 1.

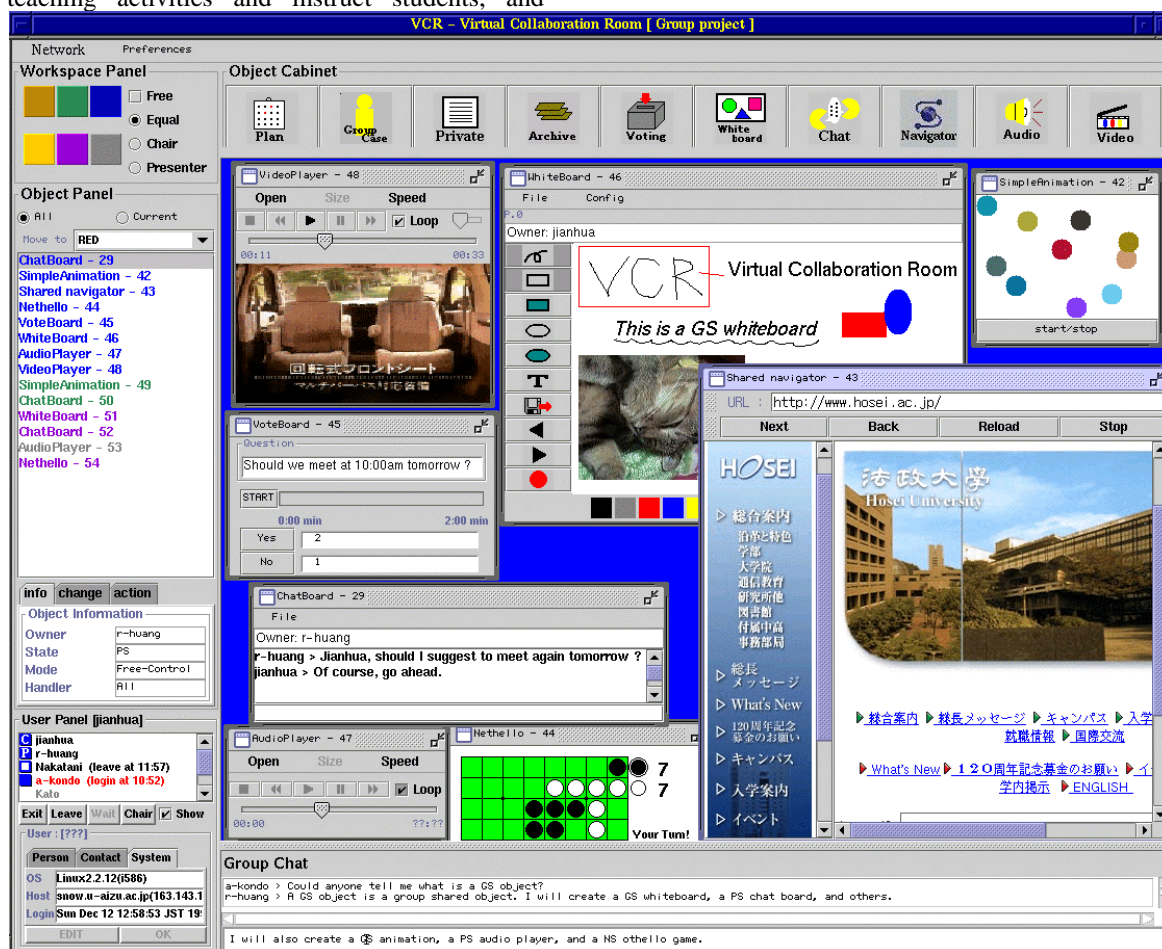


Figure 1. A VCR for group collaborative activities

By emulating physical objects, virtual objects are of the object physical characteristics that refer to rules of object presence and behavior in the associative workspace and the object social characteristics that refer to relationships of objects with users and rules of operating the objects among multi-users. The object physical characteristics include object life cycle, spatiality, temporality, persistence, and record-ability. The object social characteristics include object sharing state, owner, handler, manipulation mode, and their changeability during a collaboration period. Like a real room, VCR can incorporate many virtual objects, i.e. shared applications in this context like whiteboard, chatboard, audio player, video player, Web browser [12], voting board, animation, simulation, game, and aggregate objects in a group case, private drawer and archive in the Object Cabinet as shown in Figure 1. Users can get as many objects even same type as necessary and place them in workspace for smoothing their group activities. Once created (got from the Object Cabinet), an object is assigned a unique ID as listed in the Object Panel. Whether it is shared by all members or partial of members, or even not shared at all, can be initiated when it is created. Such sharing state, however, can be changed in the late during collaborations when it is necessary. In fact, apart from the sharing state, the ownership, manipulation mode, and the handler of an object can also be changed during collaborations based on some defined regulations. It is done by clicking “change” in the Object Panel and selecting an item in the newly popped up facet. Any created object can be recorded or destroyed by clicking “action” in the Object Panel and selecting a corresponding item from the newly popped up facet. The detailed about object recording/replaying, session recording/replaying, and room persistence will be described in the next section.

4. Object, Session, Room and Persistence

4.1. Object and Its Recording and Replaying

Objects in VCR are virtual objects that are emulating real objects (shared facilities for collaborations) in a room of the real world. During collaborations, a group of people may use only one object and sometimes they may need to use two or even more objects. VCR is designed to enable this feature and flexibility. Various kinds of virtual objects are stored in the Object Cabinet and ready to be born and placed in workspace for use. For example, if a user wants to have a chat with his friends, what he does is to click “Chat” icon in the Object Cabinet, a unique object ID is assigned and at the same time a dialogue box is popped up for specifying the sharing state of the object. If partially shared state is specified, then a list of members is popped up for selecting partial members from the list. The object is thus born or created and placed on the left top corner of the current workspace. The user who creates the object is owner and only the owner can destroyed the object. The process is equivalent to get a chatboard from a facility storehouse to

have a chat with his friends in the real world. Figure 2 shows popped up windows for specifying object’s sharing state and selecting users.

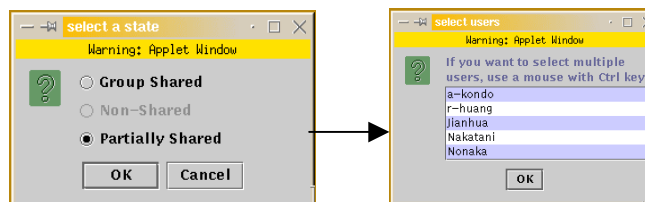


Figure 2. Specifying object sharing state

Once an object is created, a new file, “ID.log”, under the directory “Log” is generated, where ID is the object ID. After being created, the object can be viewed, accessed, and manipulated by a group of users who are sharing it at different sites over the Internet. From the time it is born until it is destroyed, the whole process or a part of the process of being manipulated by the group of users can be recorded upon the request. This is so called object recording. If only the current last state of the object is recorded, this is so called object persistence. For a memory object, it is necessary to record all history events and for a memory-less object, recording of the last event is enough. Requests for object recording or object persistence can be made as the following steps: clicking an object to specify the object from the object list in the Object Panel, clicking “action” button, a facet with a list of items is opened, and clicking a selected item. If “O-Record” is clicked, the object will be recorded. In a same way, only current state of the specified object is recorded if “O-Persist” is clicked and the specified object will be destroyed if “Destroy” is clicked as in Figure 3.

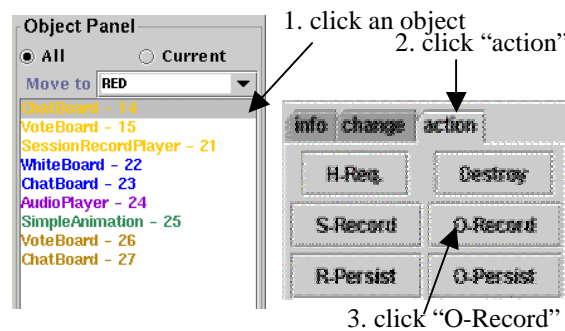


Figure 3. The process of recording an object

When the room server receives the request of object recording, it adds a record event with time stamp to the file, “ObjectRecord.log”, in the directory “Log”. An example file is shown in Figure 4.

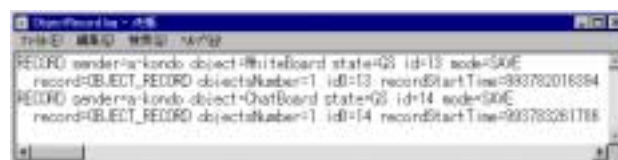


Figure 4. An object recording log file

A recorded object as a new created object is placed in the archive (an aggregate object case). When one would like to replay the recorded object, he clicks “Archive” icon and select a facet “ObjectRecord”. A list of recorded objects is listed in the display area and he selects the one and specifies its sharing state for replaying. This request of replaying an object is sent to the server, the server places an “ObjectPlayer” window in workspace. When a user clicks “START”, the server first gets the object in the last state before recording and displays it in the “ObjectPlayer” window, and then replays each event related to the object in time order from the record starting time to the current. For a recorded object, its current sharing users set (denoted as CUList) is defined as the logical AND set of the original sharing user set (denoted as OUList) and the replaying sharing user set (denoted as RUList), having,

$$CUList = OUList \cap RUList$$

Since each created object has a log file that keeps all events with time stamps, implementations of replaying a recorded object become relatively easy. The server reads the object log file, for example, “14.log” as shown in Figure 5, and gets the starting time of the object recording from the object recording log file, “ObjectRecord.log”, so as to set initial object state for replaying (the initial state for replaying is the last state before recording). After that, the server just gets each event from “14.log” for replaying in time order. If we take a look at “14.log” closely, the 8th event is starting point of the replay since the record starting time is 993783261786, which is in between the 7th event (993783246920) and the 8th event (993783288285). Generally, an object is replayed in real time, which means the replaying process sleeps between two events. One, of course, can replay the object faster than real time if he clicks “F_FORWARD” button.



Figure 5. An Object log file

4.2. Session and Its Recording and Replaying

A session is a sequence of events happened in the current workspace of a virtual collaboration room during a period of time. A session recording is to record all events happened in the current workspace within the specified period like a video recording of activities conducted in a physical room. It is strictly defined from a starting time to an ending time. In an object recording, events are a user or users’ manipulations on an object while in a session recording, events are a user or users’ manipulations on an object or a number of objects. Thus, it can be noticed that an object recording is a special

case of a session recording when there is only an object being used during the specified period of time. Since there are six workspaces and only the current workspace is visible, a log file that records the changes of current workspace is necessary. If there is no change of the current workspace, the log file is null, otherwise, each line in the log file is a change event as shown in Figure 6.



Figure 6. The current workspace change log file

To request a session recording, one clicks “action” facet in Figure 3 and then clicks “S-Record” button, and then the request is sent to the server, the server appends a session record event to the session record log file. After being clicked, “S-Record” side is turned down and “End” side is turned up. This is a doubly sided button. When one would like to terminate the session recording, he clicks “End”, the session ending time is added to the end of the session record event as shown in Figure 7.



Figure 7. The session record log file

Since each object has own log file stored in the server, the session recording becomes rather simply. Only the current workspace ID number and session starting time and ending time need to be recorded.

A recorded session is also placed in the archive as a new aggregate object. When one requests to replay a recorded session, he has to visit archive and get the recorded session he wants from the list of recorded sessions (by clicking “SessionRecord” facet, highlighting an item, and then clicking “OK”). Do not forget to specify the recorded session sharing state. Since each event in an object log file has the object ID and its sharing state attached (referring to Figure 5), each replayed event is only sent to those users who are specified in both the sharing state of the recorded session and the object. If the object related to an event in the recorded session is non-shared, the event is only sent to the object’s owner no matter how many users are selected to share the replayed session. SessionRecordPlayer is loaded to replay the specified recorded session. The SessionRecordPlayer first is to work out the initial state of the recorded session, which is the last state of the current workspace before the session recording starts, and then to display the initial state in the display area in the SessionRecordPlayer window. When a user clicks “START”, the session replaying starts on the top of its initial state. If there is a workspace change, the background color of the

window is changed to the corresponding current workspace color. Session state is also changed to the initial state of the corresponding current workspace. Since each object log file contains space number that is from 1 to 6, the implementation of finding initial state of a workspace is not difficult. It first finds objects that are in the specified workspace number and gets each object's final state before the session recording starts. The session recording time can be found from the session record log file. After the initials state is displayed and one clicks "START", the server sorts all events in the found objects in time order, and replays each of the sorted events by sending each event to its associated clients. Figure 8 shows the SessionRecordPlayer that are replaying a recorded session that includes two objects, an animation object and a whiteboard object.

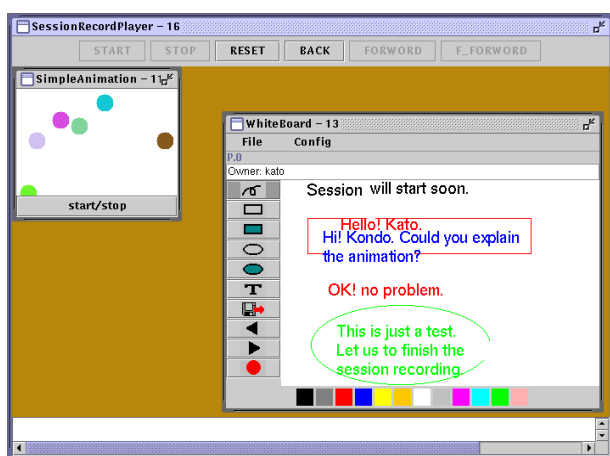


Figure 8. A recorded session being replayed

4.3. Room Persistence

Room persistence means to keep a virtual collaboration room that has been used in its state even when all users have exited from the room. When entering the room again next time, they can see everything is exactly same as before and can continue their activities from the persistent state rather than from completely scratch. It is very convenient for asynchronous collaborative work and learning. Our most work is conducted in asynchronous style although synchronous work sometimes is very necessary. Especially in network and information era, people tends to be freedom and like to have more flexible schedule, this function meets their tastes and make them happy.

The function of room persistence is enabled when one clicks "action" facet and then presses "R-Persist" button. The server generated a log file with a name "user_nameRoom.log". For example, "a-kondoRoom.log" means the user, *a-kondo*, requests to make room persistent at the current state as shown in Figure 9. In the file, it records the total number of objects in the room and each object's type and ID, workspace ID, and the record starting time. One can update the room persistence by clicking "R-Persist" again. It is recommended to update the room persistence just

before all members exit the collaboration room so that there is no problem to continue previous group work or activity. When a user enters the room again, the server reads each object log files, and places the persistent state of each object into the corresponding workspaces. So people see the room as the same as before they exited.



Figure 9. A room persistence log file

5. How Online Learning Is Improved

Distance education has been brought people a new style of teaching and learning. It takes advantages of network technologies such that education organizations can reduce campus and building cost, teachers and students can save time and cost without traveling, people can have more flexible schedule of conducting work related to teaching and learning, and students can have more alternative choices of being involved in learning without the constraint of physical location. It, however, is challenge for both teachers and students. On the one hand, students have to adapt themselves to the new learning environment and teachers have to find better methodologies with considerations of students to improve their teaching effectiveness in the new teaching environment. On the other hand, as designers of a distance education system, they have to take many factors into considerations in their designs. Different systems are designed under different emphases [2, 3-6, 7]. However, the objective is the same, that is, to make teachers teaching and students learning more efficient and effective in their system to be designed. Many researches have been devoted to study of how to make distance teaching/learning more efficient and effective.

Since educations in a physical university are of a long history and have many mature methodologies, University21 is designed under the emphasis of trying to emulate a physical education system, such as a virtual private office emulates a personal office, a virtual courseroom emulates a classroom, and a virtual collaboration room emulates a group activity room in a physical university. Most time students are conducting learning in a courseroom, such as reading teaching materials, doing self-tests, working on assignments, and etc. in their flexible schedules. Sometimes, it is necessary for them to take apart in some group activities like tele-lecturing, question and answer time, course seminar or forum, term project, group research, research meeting, and etc. If a group activity is course oriented, students enter a virtual collaboration room from a courseroom. Otherwise they enter a virtual collaboration room from their individual virtual private offices. Group activities are classified into synchronous and asynchronous two work styles. In a

synchronous activity, a group of users have to be present at the same time, such as tele-lecturing. While in an asynchronous activity, a group of users can join the activity in different time zone and the activity is always in a continual state such that later comers can continue the work done by earlier comers.

As a virtual university, there is much more beyond just providing learning materials in a web site and students are not simply reading learning materials through the Internet. Group activities in fact play an important role in online learning and functions for enabling asynchronous activities are indispensable for students to improve their learning effectiveness. For example, a student can't attend a tele-lecturing due to his conflicting schedule but he does not want to miss the lecture, so he asks his classmate to record the session and recorded session is placed in archive of the virtual collaboration room. The student thus can enter the room later when his schedule is available and replay the recorded session. Take another example, a teacher uses a group shared chatboard object in a VCR room to answer student question in a scheduled question/answer time. However, a student will be late for attending but he does not want to miss the part when he is absent, so he asks his teacher to record the chatboard object. The recorded object is also placed in the archive of the room. When he enters the room, replays the recorded object as a non-shared object, and attends the rest part of the question/answer at the same time. Both above examples solve schedule conflicting problems in synchronous activities. Now, let us take one more example to see how an asynchronous activity is conducted. Assume that several students are involved in a term project, and each one is assigned a subtask. Some subtasks have to be implemented in ordered time and some subtasks can be implemented concurrently. So it does not require all students present simultaneously but someone may start to implement a function until other finishes some parts of his subtask. It is in fact an asynchronous collaborative work. Since the task may not be finished in a short period of time and is of continuity, it is necessary to make room persistent each time before one exits the room using the function of object persistence or room persistence such that the previous work can be kept and the next incoming students can continue the work.

6. Conclusions and Future work

VCR, as an important component in University21, is aimed at enabling a variety of different types of group work and collaborative activities. People can conduct not only synchronous activities (like in most current groupware systems) but also asynchronous activities in VCR. This is achieved by adding functions of object recording/replaying, session recording/replaying, and object and room persistence to the VCR system. With the functions, VCR can support asynchronous group activities as well as solve users' schedule conflicting problem occurred in synchronous group

activities. When people can't attend or are late for a synchronous group activity, they can ask one of the attendees to record an object or a session. They can make up for what they missed by replaying the recorded object or session. The recorded objects and sessions as history data are also important for group members to review and recall back. Reviewing what you have learnt and what you have done is one of the important means in learning. In an asynchronous group work, people do not have to work simultaneously and the group work is of continuity. Object persistence and room persistence can enable such continuity of different members' work in different time zone at different geographical locations.

It is worthwhile to point out that distance education overcome geographical constraint but bring a new problem, that is, awareness of objects, people, and environments. One of our future work is to support awareness of objects, people, and environment in VCR so as to further make group collaboration more effective and efficient.

References

- [1] J. Ma and R. Huang, "Towards a Natural Internet-Based Collaborative Environment with support of Object Physical and Social Characteristics", *International Journal of Software Engineering and Knowledge Engineering*, World Scientific, No. 1, 2001.
- [2] J. Ma, R. Huang, and T. L. Kunii, "University21: An Integrated Educational System", pp109-139, Chapter 7 in the Book: *International Perspective on Tele-education and Tele-learning*, edited by G. Orange and D. Hobbs, Ashgate publishing Limited, ISBN 0-7546-1202-3, 2000.
- [3] Virtual-U, <http://virtual-u.cs.sfu.ca>
- [4] Web-CT, <http://homebrew1.cs.ubc.ca/webct/>
- [5] WCB, <http://www.madduck.com/wcbinfo/wcb.html>
- [6] LearningSpace, <http://198.114.68.60/>
- [7] S. Greenberg and M. Roseman, "Using a Room Metaphor to Ease Transitions in Groupware", Research report 98/611/02, Department of Computer Science, University of Calgary, Calgary, Alberta, Canada, January, 1998.
- [8] R. Huang, J. Ma, S. Wada and M. Osano, "Designs of an Open Agent Architecture for a Virtual Learning Community System", *Journal of Three Dimensional Images*, Vol. 14, No. 3, December 2000.
- [9] J. Ma, R. Huang and E. Tsuboi, "Cheer: a Computer-based Hyper-Environment for Educational Reformation", *Proc. Intl. Conf. on Computational Intelligence and Multimedia Application*, Australia, 1998.
- [10] R. Huang, J. Ma and T.K. Shih, "Authoring, Teaching and Learning Online Course in Virtual Courseroom over the Internet", In the Edited Book: *Advanced Research in Computers and Communications in Educations*, Vol. 2, IOS Press, 1999.
- [11] J. Ma and R. Huang, "Collaborative Teaching and Learning in Virtual Collaboration Rooms over the Internet", In the Proceedings of the International Conference on Computer Assisted Instruction and Internet Computing, Taipei, October 2000.
- [12] J. Ma, R. Huang and A. Kondo, "A Shared Browser for Synchronous Web Navigations by Multi-Users", in Proceedings of the International Conference on Information Society in the 21st Century, Aizu-Wakamats, November 2000.