The Design of a Synchronous Virtual Writing Clinic

doi:10.6688/JISE.2003.19.2.2

Journal of Information Science and Engineering, 19(2), 2003

作者/Author： 郭經華(Chin-Hwa Kuo);衛友賢(David Wible);陳孟彰(Meng-Chang Chen);宋立群(Li-Chun Sung);周家麟(Chia-Lin Chio)

頁數/Page： 249-265

出版日期/Publication Date：2003/03

JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 19, 249-265 (2003)

The Design of a Synchronous Virtual Writing Clinic*

CHIN-HWA KUO, DAVID WIBLE**, MENG-CHANG CHEN+,
LI-CHUN SUNG AND CHIA-LIN CHIO
Computers and Networking (CAN) Laboratory
Department of Computer Science and Information Engineering
Tamkang University
Tamshui, 251 Taiwan
***Research in English Acquisition and Pedagogy (REAP) Group*
Department of English
Tamkang University
Tamshui, 251 Taiwan
+*Advanced Internet Protocols and Services Group*
Institute of Information Science
Academia Sinica
Nankang, Taipei, 115 Taiwan

The design of an online synchronous writing clinic is described. Two novel mechanisms have been designed to provide the two fundamental capabilities for the clinic: (1) synchronous text co-editing, and (2) Internet voice delivery. Therefore, the participants are able to synchronously edit a common text file on-line with voice communication support by using the synchronous text co-editing tool. Furthermore, the designed policy-based forward error correction voice transmission mechanism delivers good voice conversation quality when handling delay, jitter, and packet loss on the Internet. The developed system not only has great value in supporting applications such as CSCW, Internet Telephony, or Multimedia Instruction on Demand (MID), but also has been applied in the area of Distance Language Learning by exploiting the integration of computer and networking capabilities with linguistic and pedagogical principles.

Keywords: computer supported cooperative work (CSCW), voice transmission, co-editing, distance learning, computer assisted language learning (CALL)

1. INTRODUCTION

With the rise of the World Wide Web, there has been a corresponding resurgence in Computer Assisted Language Learning (CALL). Currently, there are numerous commercially available packages ranging in format from CD-ROMs to interactive systems over the Internet. While CD-ROMs are limited in terms of the resources that can be included and in the level of interactivity and responsiveness to individual users, interactive systems have the advantage of transcending these limitations.

Among the distance learning systems on the Internet, a fundamental distinction can be drawn between two basic modes of interaction that are available: *asynchronous* and

Received April 19, 2000; revised August 29 & December 20, 2000 & January 8, 2001; accepted May 15, 2002.
Communicated by Chuen-Tsai Sun.
* The present work is partly supported by National Science Council Taiwan ROC under NSC 88-2213-E-032-002.

synchronous interaction. In systems using the asynchronous mode, lecture content is provided, typically on web pages. Learners may pose their questions or opinions on discussion boards, and e-mail serves as the main communication channel. These systems lack capabilities for real-time communication between instructors and learners or among peers. In environments that use the synchronous mode, on the other hand, systems are required that support real-time communication modules and interactivity.

In the field of CALL, the use of Computer Mediated Communication (CMC) has been examined to some extent. Of course, the potential for educational applications of CMC has risen commensurately with the advent and rise of the Internet. The design and use of synchronous CMC systems for CAI and more specifically for CALL on the Internet have not been widely reported in the literature [19-22]. The available studies make some instructive points which our design has taken into consideration.

While neither a pipe dream nor cure-all, synchronous online CMC as a pedagogical tool has been reported to increase learners' willingness to engage in discussions with teachers and fellow learners [18, 19]. In fact, Ess reports a carry-over effect where less communicative students not only become more expressive in CMC modes, but also subsequently participate more in classroom interactions as well.

The design of the present system has taken these issues into consideration. The system reported on here is a synchronous English writing clinic, called *WriteNow*, which has been designed for and implemented on the Internet. In the writing environment, we make use of a web-based user interface and provide two main functions: (1) an online essay display board with synchronous co-editing and dialog capabilities, and (2) real-time voice communication. Two algorithms are developed to provide the above functions. The *Temporal And Spatial Data conflict detection* (TASD)[1] algorithm is designed to overcome the data inconsistency problem and provide user flexibility in the co-editing process. The policy-based forward error correction voice transmission algorithm is designed to handle issues such as packet delay, jitter, and packet loss on the Internet in order to provide improved quality voice conversation. In addition, for education purposes, we provide an online comment bank for tutors in which frequently used comments can be easily stored, retrieved, and offered to students during online tutorial sessions.

The *WriteNow* system, then, differs from conventional CALL software packages in that it offers global access over the Internet, identifies the needs of learners and tutors, and provides online tools designed to meet these needs.

In the following sections, the description of the system is organized as follows. In section 2, an overview of the system is given. In section 3, we illustrate the design philosophy and building blocks of the system. The system implementation and user interface are described in section 4. The experiment results of the efficiency of voice transmission are given in section 5. Directions for future work and a conclusion are given in section 6.

2. SYSTEM OVERVIEW

WriteNow is one module in a larger web-based English learning environment designed to give learners and teachers access to each other and to online learning resources [14].

To simulate an on-site writing clinic on the Internet and to exploit the potential of the integration of computers, communication networks, and language education, the proposed writing environment is equipped with the following special tools and functions.

(1) Co-editing: This function provides an environment to mediate communication between learner and tutor or peer and peer concerning a specific portion of text written by a learner. The essay under discussion is pasted by the learner onto the essay display frame and is displayed simultaneously on the tutor's and learner's computer screens. By using the mouse to select any portion of the displayed text, both the learner and tutor are able to work on the same sentence and immediately identify the writing problem. Co-editing the same sentence, however, may result in conflicting data if a suitable co-editing mechanism is not available [1, 4]. To satisfy this requirement, a novel textual co-editing mechanism has been developed.
(2) Online conversation: To enhance communication between the participants within this environment, a real-time voice conversation channel has been provided. Since most of the current Internet environments deliver only best effort service, we have developed a voice transmission scheme to provide improved quality voice transmission for our purposes. The developed policy-based forward error correction voice transmission algorithm can overcome conditions such as packet delay [6-8], jitter [12], and packet loss [9-11, 13] on the Internet.
(3) Comment bank: This component provides users with the ability to mark a specific portion of an essay and immediately give comments concerning that portion of text. Giving comments is the basic tool that tutors have for conveying their specific suggestions to students about a piece of student writing.

Based on the above design philosophy and system requirements, the proposed system has the system framework shown in Fig. 1. With the above functions, the learner and tutor are able to work in a real-time interactive learning environment.

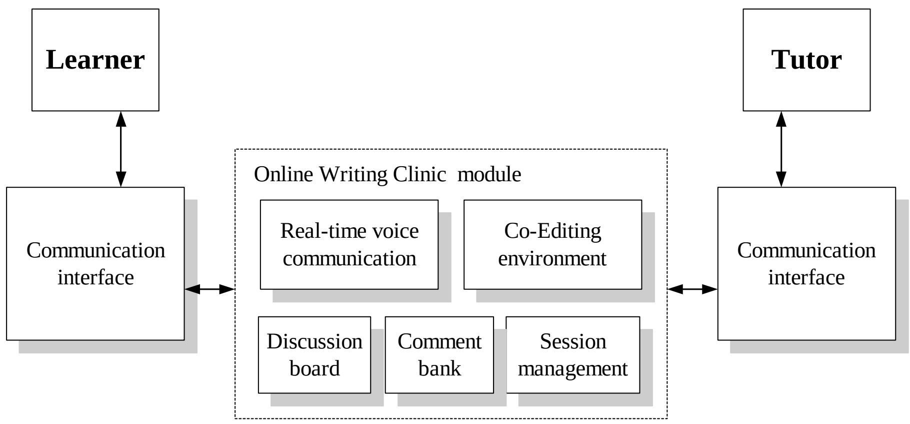


Fig. 1. The system framework of *WriteNow*.

3. DESIGN OF SYNCHRONOUS ENGLISH WRITING CLINIC

The design of the synchronous English writing clinic, *WriteNow*, is presented in this section. In the *WriteNow* module, we focus on designing the functions that provide real-time communication capability and *interactivity*. We describe below the design of the system's special features: (1) the textual co-editing environment, (2) voice transmission on the Internet, and (3) the comment bank design.

3.1 Textual Co-editing Environment

The basic purpose of the co-editing environment is to allow two users who are separated by distance) for example, a student and a tutor or two students) to discuss a piece of student writing and specifically to enable both parties to focus on the same portion of text at the same time and communicate clearly about that portion of text. In other words, the environment is intended to allow synchronous tutorial sessions for writing instruction.

This co-editing environment includes two modes, a *control* mode and a *free* mode. By designing these two modes, we are simply offering a choice to the users. Each mode has certain advantages and limitations, and users can choose accordingly. In the control mode, a priority scheme is set in place so that the tutor can overwrite the learner's output. In the free mode, the environment allows a two-way override in order to give the users a higher degree of flexibility in modifying the essay. The flexibility of the free mode is especially useful for peer editing, where neither of the participants necessarily is the "authority" figure, and the two can engage in mutual exchange much as peers would in discussing their writing face to face.

In the free mode, however, data conflict can occur. With conventional, floor control mechanisms, such as token control, the data conflict problem is resolved at the cost of some degree of flexibility [3, 5]. In the proposed approach, a co-editing mechanism, *Temporal And Spatial Data conflict detection* (TASD) [1], is developed which takes data consistency and user flexibility into account. The designed algorithm resolves data conflict by using an *undo* process. Naturally, data conflict arises when two events occur within the same markup area. Further analysis of the events into relationships such as temporal relations, spatial relations, and event attributes makes it possible to avoid many *undo* processes.

The version of the co-editing algorithm proposed here is an extension of our previous work [1]. An event is, for example, an editing operation, such as insert, or delete. As shown in Fig. 2, the user first marks up the area of interest and then performs the desired operations. An event consists of information such as the scope of the mark up area, the operation type, and so on at the local end. When the user presses the enter key, an event is then generated and transmitted. Users need to operate the markup procedure to indicate the area of interest before any text editing operations are performed. Hereafter, we will simply refer to the markup area as the *area*. When one manipulates the characters D, E, or F, the location of the character C is called the *starting position* (*start_po*) of the *area*.

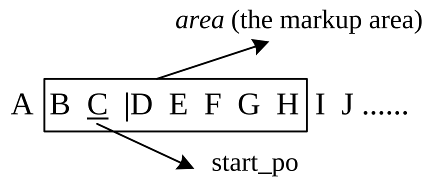


Fig. 2. The spatial notation of a markup area.

Consider the following scenarios:

(1) Both events have the same starting position:
In this case, data conflict usually occurs.

(2) Events have different starting positions:
Both events' starting positions (start_po) and lengths are extracted. Assume event A has a starting position that is ahead of event B.

(a) Both A.start_po. and B.start_po. are not in the intersection area:
As shown in Fig. 3, since A.start_po. and B.start_po. are not located in the intersection area, the delete or insert operations of A or B will not affect the other event.

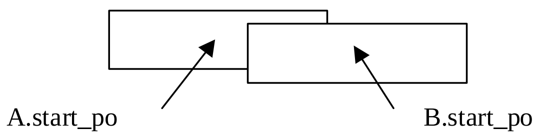


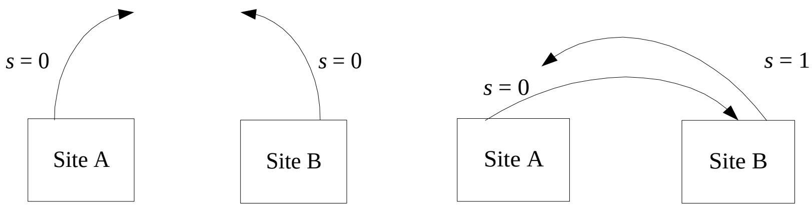
Fig. 3. A situation which may occur during co-editing.

(b) Only A.start_po is in the intersection area:
The result is the same as (1). The operations A and B do not affect each other.

(c) Only B.start_po is in the intersection area:
As the starting position of B is located inside the *area* of A, conflict may occur. If event A entails an insert operation, conflict will not occur. If event A entails a deletion, data conflict will depend on the delete length. In this case, if the content block that the user wants to delete in the markup area of A covers B.start_po, data conflict will result. On the other hand, this implies that there is no data conflict.

(d) Both A.start_po and B.start_po are in the intersection area:
If the starting position of both events is located in the intersection area, conflict may occur.

Each event has its own sequence in the co-editing process. If two events occur at the same time, data conflict may occur. The proposed co-editing algorithm, TASD, detects the event sequence by using a sequence flag *s*. A sequence flag is generated and transmitted to the other site when an event occurs. We will make use of the sequence pair (s_1, s_2) to illustrate the process. As shown in Fig. 4 (a), if both sites generate the sequence flag 0, the resulting sequence pair (0,0) indicates that data conflict may occur. Further analysis will follow to determine whether data conflict does exist or the two events can be performed simultaneously. If the events occur sequentially, however, as in (b), the first event will be indicated with sequence flag 0, and the second event will be denoted with sequence flag 1. The resulting sequence pair (0,1) indicates that these two events have no data conflict.



(a) Two events are simultaneous (b) Two events are sequential

Fig. 4. Events sequence flags.

The block diagram of the TASD algorithm is shown in Fig. 5. It determines whether the events have spatial intersection. If the two events entail the same start_po, event sequence flag comparison then follows. As mentioned before, the sequence pair (0, 1) indicates that the events are sequential rather than simultaneous, implying the no conflict condition. On the other hand, the sequence pair (0, 0) indicates a conflict condition. One of the events then must undergo the undo process. If two events do not have the same start_po, further event attribute analysis is performed in the instruction conflict-detecting block as shown in Fig. 5. If the result indicates no conflict, both events proceed and circumvent the undo process. If a spatial conflict is found, however, the temporal sequence determination operation needs to be performed as described above. The detailed event attribute analysis and implementation can be found in [1].

3.2 Voice Transmission on the Internet

As is well known, currently, the Internet offers only point-to-point *best-effort* service, which may lead to packet delay, delay variation, and packet loss. As a result, the voice quality is degraded. We have designed a *policy-based forward error correction* voice scheme to overcome this type of transmission difficulty.

A representation of the desired mechanisms offered in the voice tool is shown in Fig. 6 [2]. We emphasize the design of the redundant packet sending mechanism and QoS monitoring mechanism. The redundant packet sending mechanism evaluates the sending packet based on a set of compression packet formats to compensate for packet loss. The QoS monitoring mechanism serves the purpose of collecting the present network traffic

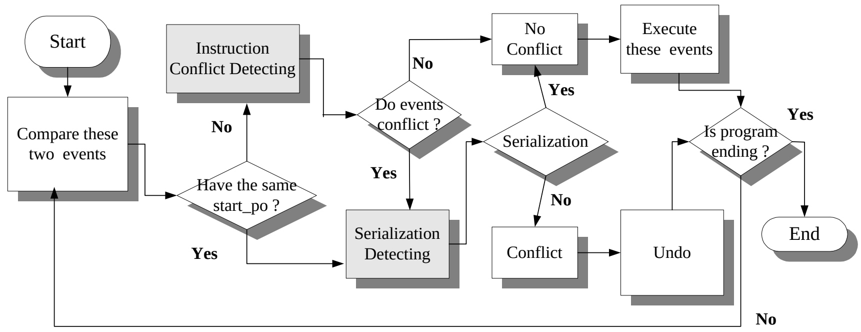


Fig. 5. The block diagram of the TASD co-editing algorithm.

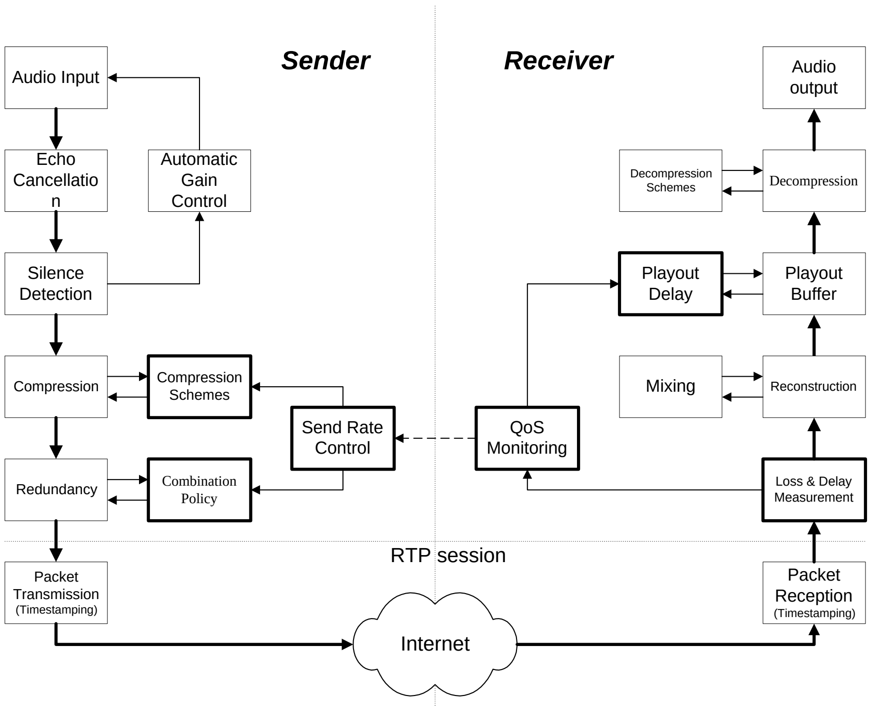


Fig. 6. The adaptive scheme for packet audio over the Internet.

conditions, such as delay, delay jitter, and packet loss. This information is then used in playout time adjustment.

3.2.1 Loss and delay measurement mechanism

The delay and loss measurement mechanism judges the current network delay state by measuring the following three parameters: the network loss rate (l_{Nw}), the smoothed loss rate (l_{smooth}), and the loss rate after reconstruction (l_{Rw}). Then, this information is

passed to the QoS adjustment mechanism to determine whether the system needs to use redundant information in the present packet to reconstruct the lost main information of the previous packets.
The network loss rate l_{Nw} is computed as follows:

$$l_{Nw} = 1 - \frac{w}{\delta_i}$$

where δ_i represents the expected received packet number in a certain period of time, and w denotes the actual number of packets received.

After this calculation, we can refresh the smoothed loss rate l_{smooth} that represents the long-term network loss rate:

$$l_{smooth} = (1-\alpha) \times l_{smooth} + \alpha \times l_{N_W}$$

where α is the linear recursive filter gain constant. Here, α equals 0.998.

As redundant packets are transmitted, the loss rate after reconstruction l_{Rw} is shown as follows:

$$l_{Rw} = l_{Nw} \times (1 - e_{Tw})$$

where e_{Tw} represents the error correction rate.

3.2.2 QoS adjustment mechanism

The designed QoS adjustment mechanism is similar to that in [2]. Only when it receives the starting packet of the new talk spurt, the receiving end begins to run the QoS adjustment mechanism. As shown in Fig. 7, according to the smoothed network loss rate l_{smooth}, the network loss rate l_{Nw}, and the loss rate after reconstruction (l_{Rw}), the QoS adjustment mechanism queries the table of the corresponding network congestion state in order to determine the bandwidth usage and redundant package sending policy. The details of the above corresponding relationship are shown in Tables 1 and 2. The policy table is given in Table 3.

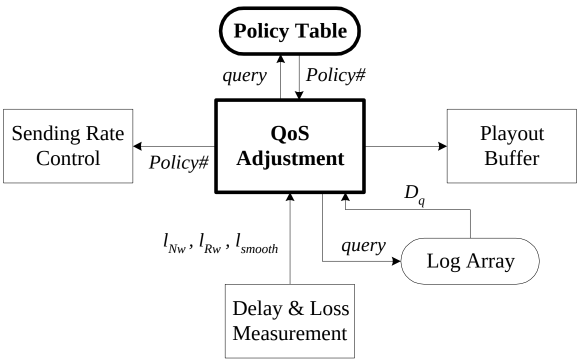


Fig. 7. The QoS adjustment mechanism.

In Table 1, L% and U% represent the lower bound and the upper bound of network loss rate, respectively. If the resulting l_{smooth} is less than L%, this implies a light traffic situation. Network bandwidth utilization is allowed to increase to a certain value, μ. If the resulting l_{smooth} is greater than U%, this implies a heavy traffic situation. The network bandwidth utilization requires a decrease by a certain percent in the current bandwidth, ν. We use μ = 2.4kbp, ν = 0.875 [15, 16], and L = 5 and U = 25 in the implementation [15].

Table 1. Network congestion state vs. bandwidth usage.

Loss Rate	State	Bandwidth	New Bandwidth
l_{smooth} < L%	Unloaded	Increase	bandwidth = bandwidth + μ
L% < l_{smooth} < U%	Loaded	Maintain	bandwidth = bandwidth
l_{smooth} > U%	Congested	Reduce	bandwidth = bandwidth * ν

Next, we compute L', the maximum of (l_{Nw}, l_{smooth}), to query the table as shown in Table 2 to obtain the corresponding number of redundancies. In the case where L' < L%, no redundant packets are added. If L' > U%, two redundant packets are transmitted with the primary packet.

Table 2. Loss rate vs. redundant information usage.

Loss Rate	Primary	1st Redundancy	2nd Redundancy
L' < L%	○		
L% < L'< U%	○	○	
L' > U%	○	○	○

Based on the bandwidth usage and redundant information as described above, the policy number is determined as indicated in Table 3. This policy number is then sent back to the sending end. The sending end also maintains the policy table in order to choose the sending packet format for the next talk spurt.

3.3 Comment Bank Design

The comment bank is designed with a standard set of comments and an optional set of comments. The standard set of comments is shared among all tutors, whereas each tutor has an optional set of comments which she or he can edit or modify according to individual preferences. The most frequently used comments can be collected in the comment bank so that tutors need not type them each time from scratch but can simply select them from the bank.

4. IMPLEMENTATION AND RESULTS

In this section, we describe the implementation of two main components in *WriteNow*: (1) voice transmission, and (2) the user interface and co-editing environment.

Table 3. Policy table.

Redundancy	Primary	1st redundant	2nd redundant	Bandwidth	Policy
0	PCM	-	-	64	1
0	ADPCM	-	-	32	2
0	GSM	-	-	13	3
1	PCM	ADPCM	-	96	4
1	PCM	GSM	-	77	5
1	PCM	LPC	-	68.8	6
1	ADPCM	GSM	-	45	7
1	ADPCM	LPC	-	36.8	8
1	GSM	LPC	-	17.8	9
2	PCM	ADPCM	GSM	109	10
2	PCM	ADPCM	LPC	100.8	11
2	PCM	GSM	GSM	90	12
2	PCM	GSM	LPC	81.8	13
2	PCM	LPC	LPC	73.6	14
2	ADPCM	GSM	GSM	58	15
2	ADPCM	GSM	LPC	49.8	16
2	ADPCM	LPC	LPC	41.6	17
2	GSM	GSM	GSM	39	18
2	GSM	GSM	LPC	30.8	19
2	GSM	LPC	LPC	22.6	20

MS VC++ is used as the programming tool in the implementation. Currently, the system runs on the Windows 98, Windows 2000 or Windows NT platforms and is available for students in the Department of English at Tamkang University [16].

(1) Implementation of voice transmission
The voice transmission scheme is shown in Fig. 8. It consists of silence detection, compression, and transmission, as discussed in section 3.2. We utilize APIs, such as ACM Functions (Audio Compression Manager) and Winsock II, under Windows 98, Windows 2000 and Windows NT in the implementation of voice compression and media transmission, respectively.

(2) Implementation of the user interface and co-editing
The graphic user interface of *WriteNow* is shown in Fig. 9. After initiation of the main page, a user can select the tutor from the online user list. When the connection is made between the two users, the learner and tutor can make use of the co-editing area for editing. The markup area will also appear on the screen at the remote site, allowing both users to focus simultaneously on the same portion of text though they are at distant locations. The comment bank also appears during a tutoring session. A voice control panel and text discussion board are provided as well. Thus, users may choose these communication channels for their convenience.

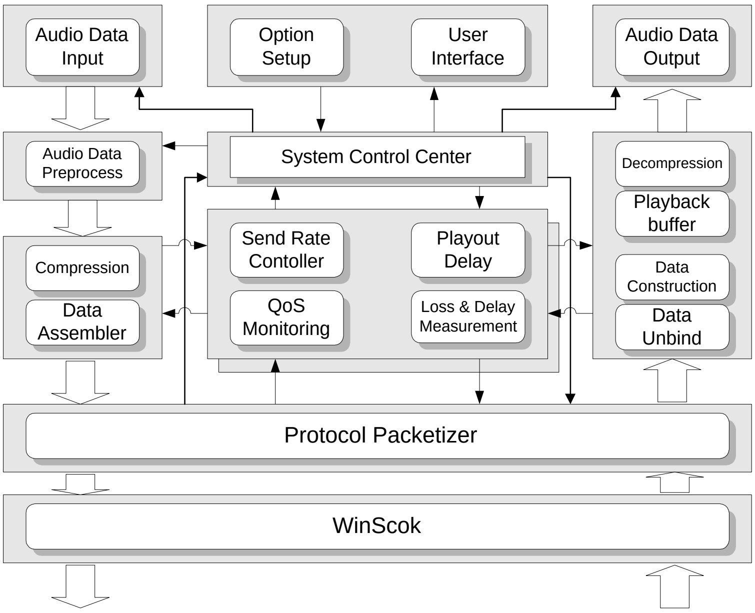


Fig. 8. The voice transmission scheme.

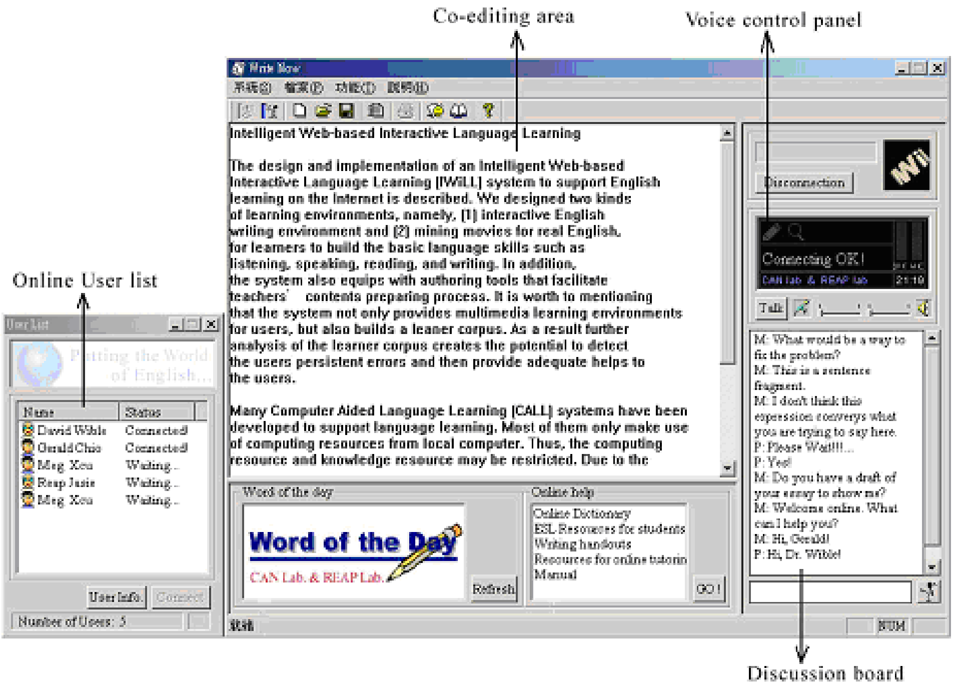


Fig. 9. The *WriteNow* synchronous virtual writing clinic GUI.

5. EXPERIMENT RESULTS

To verify the efficiency of the proposed voice transmission scheme, in the section we show two sets of experiment results: (1) results for an experiment conducted over a simulative network, and (2) results for an experiment conducted over the Internet.

5.1 Experiment Conducts Over a Simulative Network

First, we performed an experiment based on a simulative network. The network architecture is shown in Fig. 10. It consists of two routers and four testing computers. The testing scenario is described as follow. Two of these testing computers, labeled *Sender* and *Receiver*, are responsible for voice transmission and retrieval through these two routers. The other testing computers, labeled *Traffic generator* and *Background traffic receiver*, are responsible for generating background traffic to noise the voice transmission. In the *Traffic generator*, we adopt the *mgen* program to generate two kinds of background traffic simultaneously. Among these two types of noise traffic, one is sent at a rate of 5000 packets per second, and the other is sent at a rate of 10,000 packets per second. The packet size for both types of traffic is set at 512 bytes. Generally speaking, these two types of noise traffic can cause the packet loss rate to increase by 10 % to 15 %, on average.

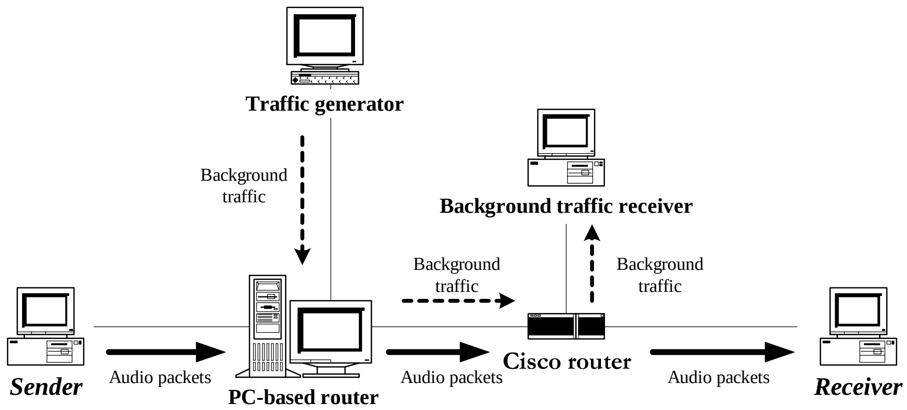


Fig. 10. Testing environment of the simulative network.

In addition, some important parameters are defined in the proposed algorithm as follow: α = 0.998, μ = 4.8, ν = 0.875, L = 5, and U = 10. Based on the above testing scenario, the experiment results are shown in Fig. 11.

In Fig. 11, the dotted line represents the native network packet loss rate (l_{Nw}), and the solid line represents the resulting packet loss rate obtained by adopting the proposed scheme (l_{Rw}). As shown in this figure, the developed QoS adjustment mechanism can reduce the packet loss rate to 8 %. In addition, when the redundant packet reconstruction scheme is employed, the packet loss rate decreases to 1 %.

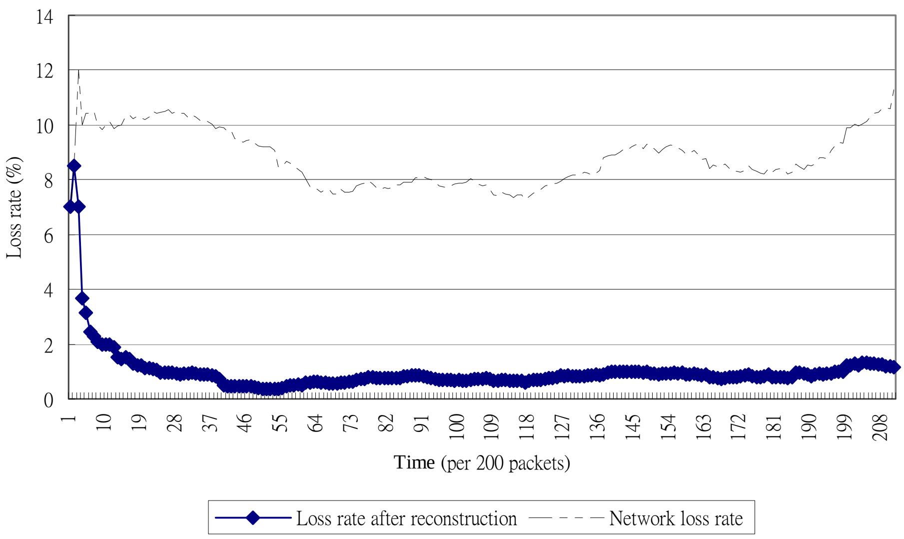


Fig. 11. Experiment results obtained over the simulative network.

5.2 Experiment Conducted Over the Internet

We performed voice tool testing from Tamkang University, which is located in northern Taiwan, to National Sun Yet-Sen University, which is located in southern Taiwan, on the Internet. The selected parameters of the proposed algorithm were as follows: α = 0.998002, L = L' = 5, U = U' = 25, μ = 4.8kbps and ν = 0.875. We used a 20 ms duration to form a packet size. The results are shown in Fig. 12 for a test which consisted of 38,000 packets.

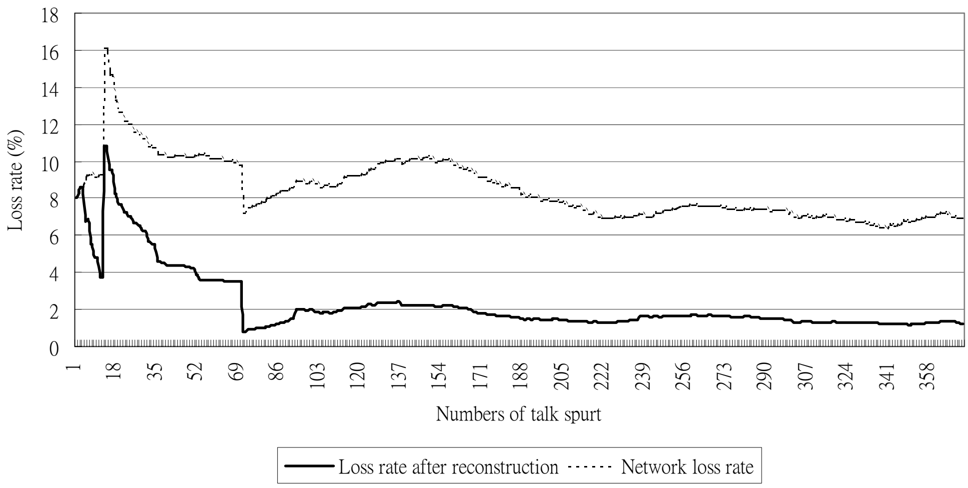


Fig. 12. Experimental results obtained over Internet.

The common network packet loss rate was around 20 % to 30 % over this link (Tamkang University to National Sun Yet-Sen University). As shown in Fig. 12, the developed QoS adjustment mechanism reduced the network packet loss rate (l_{Nw}) to 10 % (denoted by the dotted line). When the redundant packet reconstruction scheme was used, the packet loss rate (l_{Rw}) decreased to 1 % to 2 % (denoted by solid line).

6. CONCLUSIONS AND FUTURE WORK

Advances in computer and Internet technology has resulted in dramatic changes in contemporary society. At this stage, one critical challenge produced by these changes is how to use these technologies in the development of novel and effective learning environments. In this paper, we have presented our design philosophy, premised upon the integration of computer and networking technologies, and language pedagogy in the construction of a synchronous virtual writing clinic, *WriteNow*, on the Internet. The proposed system breaks down the temporal and spatial limitations.

To achieve the design goals, the developed system consists of the following features that are distinct from usual networked writing environments. First, it provides a co-editing environment with voice communication support, which allows users to attend to a common focal point though separated by great distances. Thus, users may co-edit the same sentence and then identify relevant learning problems immediately. Second, a comment bank is provided which includes comments for encouragement and reinforcement, for correction, and for practical matters of session management. Through the tutor's use of the designed comment bank, a learner can be made aware of the nature of his or her problems and strengths and can be offered immediate resources that address these.

The proposed system is currently being tested in the Department of English at Tamkang University. The corresponding software works under Windows 98/2000/NT and is available from our web site [17]. The synchronous writing clinic is currently integrated with a complementary *asynchronous* writing environment–an interactive English writing system in IWiLL used by over 200 English majors and six English professors at Tamkang University. The resulting system constitutes a novel multifaceted writing environment for second language learners and teachers.

REFERENCES

1. C. H. Kuo, C. L. Chio, C. H. Yu, and W. Y. Hsia, "In supporting of distributed co-editing environment," in *Proceeding of International Computer Symposium, Workshop on Computer Networks, Internet, and Multimedia*, 1998, pp. 17-24.
2. C. H. Kuo, T. C. Hsu, and W. C. Hsi, "Adaptive scheme for packet audio over the internet," in *Proceedings of SPIE*, Vol. 3561, 1998, pp. 242-252.
3. H. P. Dommel and J. J. Garcia-Luna-Aceves, "Floor control for multimedia conferencing and collaboration," in *Proceedings of ACM Multimedia Systems*, Vol. 5, 1997, pp. 23-38.
4. M. Handley and J. Crowcroft, "Network text editor (NTE)–A scalable shared text editor for the Mbone," in *Proceedings of the ACM SIGCOMM '97*, Vol. 27, 1997, pp. 197-208.

5. M. O. Pendergast, " A comparative analysis of groupware application protocols," in *Proceedings of the ACM SIGCOMM '98*, Vol. 28, 1998, pp. 28-40.
6. T. C. Hsu, "Adaptive synchronization mechanism for packet audio over the Internet," Master thesis, Department of Computer Science and Engineering, Tamkang University, 1998.
7. S. B. Moon, J. Kurose, and D. Towsley, "Packet audio playout delay adjustment: performance bounds and algorithms," in *Proceedings of ACM Multimedia Systems*, 1998, pp. 17-28.
8. P. L. Tien and M. C. Yuang, "Intelligent voice smoother for VBR voice over ATM network," in *Proceedings of IEEE Infocom'98*, Vol. 2, 1998, pp. 841-848.
9. B. J. Dempsey and Y. Zhang, "Destination buffering for low-bandwidth audio transmission using redundancy-based error control," *in Proceedings of 21st IEEE Local Computer Networks Conference*, 1996, pp. 345-355.
10. M. Podolsky, C. Romer, and S. McCanne, "Simulation of FEC-based error con trol for packet audio on the Internet," in *Proceedings of IEEE Infocom'98*, Vol. 2, 1998, pp. 505-515.
11. J. C. Bolot, and A. V-Garcia, "Control Mechanisms for Packet Audio in the Internet," in *Proceedings of IEEE INFOCOM'96*, 1996, Vol. 1, pp. 232-239.
12. S. Jha and M. Fry, "Continuous media playback and jitter control," in *Proceedings of IEEE ICMCS'96*, 1996, pp. 245-252.
13. J. C. Bolot, S. Fosse-Parisis, and D. Towsley, "Adaptive FEC-based error control for Internet telephony," in *Proceedings of IEEE Infocom'99*, Vol. 3, 1999, pp. 1453-1460.
14. D. Wible, C. H. Kuo, F. Y. Chien, A. Liu, N. L. Tsao, and H. S. Chui, "A web-based EFL writing environment: Integrating information for learners, Teachers, and researchers," conditional acceptance, *Computers and Education*.
15. C. H. Kuo, C. L. Chio, W. H. Chen, and T. C. Hsu, "Delivering voice over the Internet," in *Proceedings of NCS, Workshop on Computer Network*, 1999, pp. 74-80.
16. K. V. Chin, S. C. Hui, and S. Foo, "Enhancing the quality of Internet voice communication for Internet telephony systems," *Journal of Network and Computer Applications*, Vol. 21, 1998, pp. 203-218.
17. Intelligent Web-based Interactive Language Learning System, http://www.iwillnow.org.
18. B. Skinner and R. Austin, "Computer conferencing: Does it motivate students?" *ELT Journal*, Vol. 53, 1999, pp. 270-279.
19. C. Ess, "Wag the dog? Online conferencing and teaching," *Computers and the Humanities*, Vol. 34, 2000, pp. 297-309.
20. G. G. Colomb, and J. A. Simuti, "Visible conversation and academic inquiry: CMC in a culturally diverse classroom," in Susan C. Herring (ed.), *Computer-Mediated Communication: Linguistic, Social, and Cross-Cultural Perspectives*. Amsterdam: John Benjamins, 1996, pp. 203-222.
21. A. DiMatteo, "Under erasure: A theory for interactive writing in real time," *Computers and Composition*, Vol. 7, 1990, pp. 71-84.
22. M. Warschauer, "Comparing face-to-face and electronic discussion in the second language classroom," *Calico Journal*, Vol. 13, 1996, pp. 7-26.

Kuo Chin-Hwa (郭經華) received the B.S. degree in mechanical engineering from Chung-Yuan University in 1980, the M.S. degree in electrical engineering from Marquette University, Milwaukee, WI, in 1989, and the Ph.D. degree in electrical engineering from the University of Notre Dame, South Bend, IN. in 1994. Currently, he is an Associate Professor in the Department of Computer Science and Information Engineering, and Director of the Division of Distance Education, both at Tamkang University, Taipei, Taiwan, R.O.C. His research interests include multimedia processing and applications, computer-assisted learning, and adaptive systems. Dr. Kuo is a member of the IEEE Circuits and Systems Society's Technical Committees on multimedia systems and applications.

David Wible (衛友賢) received his Ph.D. in Linguistics from the University of Illinois at Urbana-Champaign in1990. Currently, he is an associate professor in the English Department and the Graduate Institute of Western Languages and Literature at Tamkang University in Taipei, Taiwan. He has taught theoretical and applied linguistics at University of Florida at Gainesville and at Penn State University and has served on the faculty of the Department of Foreign Languages and Literature at National Taiwan University. He was a visiting scholar in the Academia Sinica's Institute of Information Science in Taiwan in the summers of 2000 and 2001. His research interests and publications cover comparative Chinese and English syntax, the syntax-lexical semantics interface, lexical representation, and the application of computational tools to second language research and web-based language learning environments.

Chen Meng-Chang (陳孟彰) received the B.S. and M.S. degrees in Computer Science from National Chiao-Tung University, Taiwan, in 1979 and 1981, respectively, and the Ph.D. degree in Computer Science from the University of California, Los Angeles, in 1989. He joined AT&T Bell Labs in 1989, and then was an Associate Professor at Department of Information Management, National Sun Yat-Sen University, Taiwan, from 1992 to 1993. Since then he has been with Institute of Information Science, Academia Sinica, Taiwan. He holds Associate Research Fellowship since July 1996, and assumed the responsibility of Deputy Director from August 1999 to July 2002. His current research interests include QoS networking, wireless networks, multimedia systems and networking, database and knowledge base systems, operating systems, knowledge discovery and knowledge representation. Dr. Chen is a member of ACM and IEICE.

Sung Li-Chun (宋立群) received the B.S. and M.S. degrees in computer science and information engineering from TamKang University, Taiwan ROC, in 1996 and 1998. Currently, he is a Ph.D. student of Department of Computer Science and Information Engineering at TamKang University. His research interests include multimedia networking, storage system, and information retrieval.

Chio Chia-Lin (周家麟) received the B.S. and M.S. degrees in computer science and information engineering from TamKang University, Taiwan ROC, in 1998 and 2000. Currently, he is an associate researcher of Chunghwa Telecommunication Laboratories. His research interests include multimedia networking.