

# Beyond Videophones: TeamWorkStation-2 for Narrowband ISDN

Hiroshi Ishii, Kazuho Arita, and Takashi Yagi  
NTT Human Interface Laboratories, Japan

**Abstract:** TeamWorkStation-2 (TWS-2) is introduced to connect two sites with a desktop overlay service using narrowband ISDN (N-ISDN) and the CCITT H.261 standard. Based on the experience gained with TWS-1 use within NTT, we radically simplified the system architecture of TWS-2. Experimental sessions confirmed that TWS-2 is useful for freehand drawing and gesture-intensive design sessions even with the basic rate interface (2B+D). Video delay and jerkiness did not prevent users from concentrating on their task. We are convinced that TWS-2 has a big advantage over ordinary videophones as a narrowband ISDN service.

## 1 Introduction

In order to provide distributed users with a "seamless shared workspace" that every member can see, point to and draw on simultaneously using various personal tools, we designed the first prototype of **TeamWorkStation-1 (TWS-1)** in 1989 (Ishii, 1990). TWS-1 integrates two existing individual workspaces: computers and desktops. Because each coworker can continue to use his/her favorite application programs or manual tools simultaneously in the virtual shared workspace, the cognitive discontinuity (seam) between the individual and shared workspaces is greatly reduced. We conducted many experiments using the TWS-1 prototype in our laboratories and confirmed the advantage of translucent video overlay approach of TWS over previous computer-oriented approaches (Ishii and Miyake, 1991).

We wanted to conduct further evaluation of TWS in real work environments using the public network. However, the bulky and complicated system architecture of the TWS-1 prototype prevented us from moving the system out of the laboratory.

To overcome this limitation, we started designing a completely new system, **TeamWorkStation-2 (TWS-2)**. TWS-2 was designed to use narrowband ISDN (N-ISDN) Basic Rate Interface (2B+D) and the Primary Rate Interface (H1/D). TWS-2 connects two sites via N-ISDN with a *desktop overlay* service and the *ClearFace* interface (translucent, movable, and resizable face windows over the shared workspace) (Ishii and Arita, 1991) using the H.261 standard for moving picture transmission.

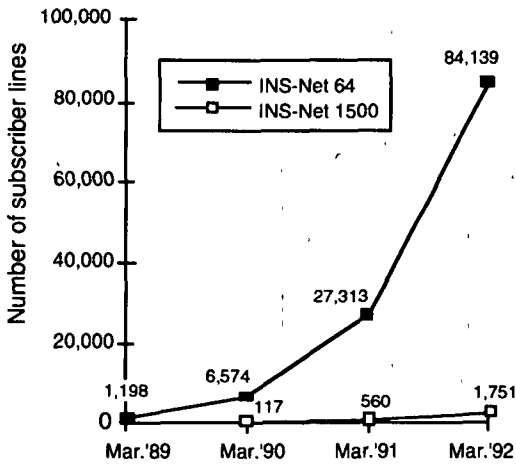


Fig. 1 Growth of Narrowband ISDN Services in Japan

In April 1988, NTT started an ISDN BRI (Basic Rate Interface) service (called INS-Net 64) in Japan using the existing metallic cables<sup>1</sup>. In June 1989, NTT started an ISDN PRI (Primary Rate Interface) service (called INS-Net 1500) using optical fiber cables<sup>2</sup>. Since the introduction of these ISDN services, the number of subscriber lines has grown steadily as shown in Fig. 1.

It is reasonable to target TWS-2 toward these narrowband ISDN services, especially INS-Net 64, because connections are available in most big cities in Japan. However, the question was if TWS-2 would really be usable under the limited 2B+D bandwidth. To find the answer to this question, we started designing TWS-2 in 1990, and completed the first working prototypes in October 1992.

1 INS-Net 64 service provides two 64kbps information channels (B channels) and one 16kbps signaling channel (D channel). Using the existing metallic cable, INS-Net 64 service requires only the addition of a compact DSU (Digital Service Unit).

2 INS-Net 1500 service offers digital communications at 64kbps, 384kbps, and 1.536Mbps. This service is accessed by means of optical fiber cable and a DSU (Digital Service Unit).

This paper describes the design evolution from TWS-1 to TWS-2, and discusses the rationale of the new design. We also report the initial findings of preliminary experiments of TWS-2 on the basic rate interface (2B+D).

## 2 TWS-1 Architecture and Problems

TWS-1 was designed in 1989 to provide small work groups (2~4 members) with a seamless shared workspace. The TWS-1 prototype utilized networked Macintosh™ computers. The system architecture of the TWS prototype is illustrated in Fig. 2. In order to connect distributed workstations, a video network (NTSC and RGB) and an input device network were developed, and integrated with an existing data network (LocalTalk™ network) and a voice (telephone) network.

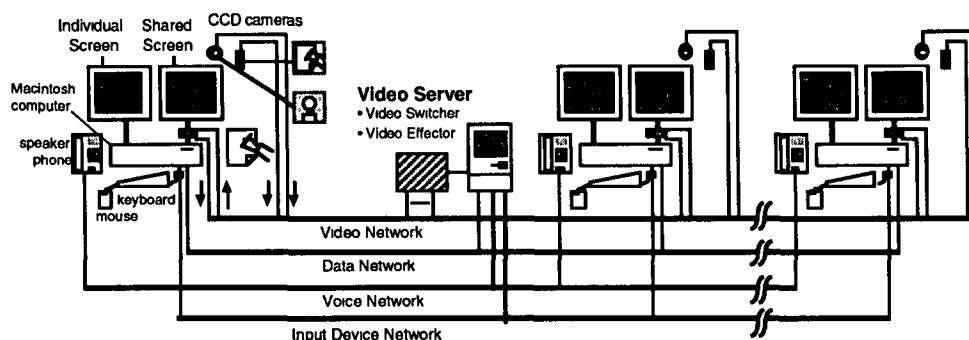


Fig. 2 System Architecture of TWS-1 Prototype (Ishii and Miyake, 1991)

The video network is controlled by a *video server* that is based on a computer-controllable video switcher and video effector. The video server gathers, processes, and distributes the shared computer screen images, desktop images, and face images. Overlay of video images is done by the video server. The results of overlaying are redistributed to the shared screens via the video network.

Each TWS-1 terminal provides a shared screen and an individual screen. The shared screen supports (1) a shared drawing window for concurrent pointing, writing, drawing, and (2) live face windows for face-to-face conversation. Fig. 3 shows the appearance of TWS-1, and Fig. 4 shows an example of the shared screen in a design session.

The individual and shared screens of TWS-1 are contiguous in video memory. Therefore, just by moving the window of any application program from the individual to the shared screen, a user can transmit the application's window to all participants for remote collaboration.

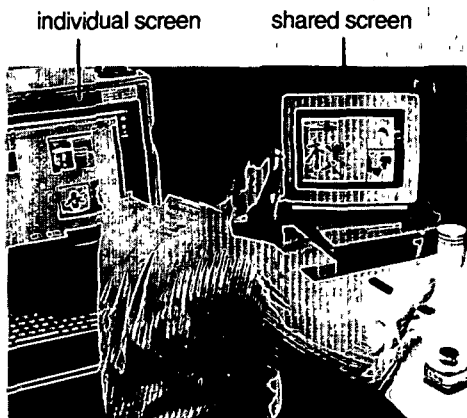


Fig. 3 Appearance of TWS-1

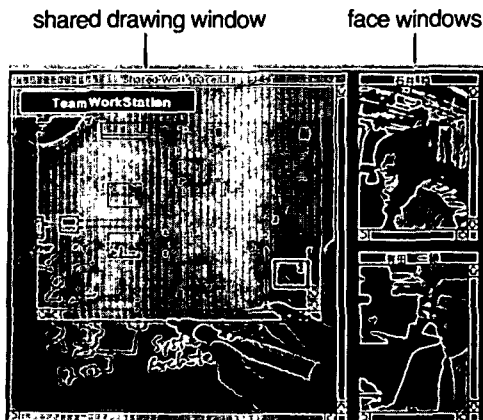


Fig. 4 An Example of Shared Screen of TWS-

### Problems of TWS-1 Architecture

TWS-1 was designed to provide a flexible environment for *laboratory experiments* using a variety of video equipment. We made it easy to change the network configuration and to add new video effect functions to the video server.

TWS-1 was originally conceived as a B-ISDN (Broadband ISDN) service so that the video bandwidth was not a problem. We did not pay much attention to the number of cables or the bandwidth. Hubris has its price, and now the system configuration is so complex that only an expert can maintain it.

This complexity of the TWS-1 architecture prevented us from extending its scale and portability and conducting field tests using the public network. The problems are summarized as follows: 1) TWS-1 requires a special video server, and its operation needs special skills and knowledge, 2) The hybrid network (NTSC+RGB analog video network, voice network, input device network, and LAN) is very clumsy and hard to extend over long-distances, and 3) The system is too bulky and expensive. These problems motivated the authors to design the completely new architecture introduced in the following section.

## 3 Architecture of TWS-2

### INS-Net 64 (or 1500)

TWS-2 is designed to run on standard INS-Net 64 and 1500. The main target is INS-Net 64 rather than 1500 because INS-Net 64 is much more widespread in Japan. One advantage of INS-Net 64 is that it uses existing metallic cables, and requires only the addition of a compact DSU (Digital Service Unit) at each site. On

the other hand, INS-Net 1500 requires the installation of optical fiber cable in addition to the DSU.

From the HCI (Human-Computer Interaction) research point of view, we are interested in understanding the effect of limited video bandwidth on shared workspace activities. We wanted to know if TWS-2 could be effective or not within the limited 2B+D bandwidth of INS-Net 64.

## H.261 Video Compression Standard

TWS-2 uses a video CODEC that supports the video compression standard H.261. H.261 is recommended by CCITT<sup>3</sup> as the compression algorithm for transmitting moving pictures. TWS-2 uses CIF (Common Intermediate Format) whose resolution is 352 pixels/line x 288 lines/picture. Because of the adoption of H.261, TWS-2 can be used with any video CODEC supporting the H.261 standard.

## Dyadic Architecture

The centralized video server made TWS-1 network configuration complex, and hard to move out of the laboratory. To make the video server unnecessary, we needed to radically simplify the system architecture and the service functions.

We decided to limit the number of sites (TWS-2 terminals) to be connected at any one time to *two*<sup>4</sup>. This is a critical design decision. We know that there is a need for multi-point connections in video conferencing services. Although the technology of multi-point connection is already available, it definitely increases system complexity and cost.

The main argument for this decision, beyond technical and economic arguments, is our observation that most daily realtime informal communication happen between two persons. The target of *dyadic communication*<sup>5</sup> seems to be most important and appropriate as the first target of TWS-2 usage.

Another reason is that most telephone calls are between *two* people, and most video conferences connect *two* distributed meeting rooms. Although some pre-planned formal meetings are held between more than two points, the main target of TWS-2 use is the informal design sessions established between a small number of users. When the appeal of multi-point connection becomes irresistible, we may extend the TWS-2 architecture.

---

<sup>3</sup> CCITT stands for "International Telegraph and Telephone Consultative Committee".

<sup>4</sup> Although TWS-2 limits the number of terminals to be connected to two, it is naturally possible that more than two users use each terminal simultaneously. Therefore, multi-user/two location meeting using TWS-2 is possible.

<sup>5</sup> Panko and Kenney extensively discussed the importance of dyadic communication in (Panko, 1991). They pointed out that "roughly half of all organizational communication is dyadic."

## TWS-2 Architecture

We completed the first working TWS-2 prototype in October 1992. Fig. 5 shows the system architecture which does not use a video server. The two TWS-2 terminals are connected by one ISDN link. Each terminal is composed of three major components: a TWS-2 Box, a PC-9801™ personal computer, and a video CODEC<sup>6</sup>. All video processing functions (e.g. translucent overlay, picture-in-picture) are supported at each terminal. All the hardware for video processing, camera control units, audio amplifiers, and power unit were encapsulated into a single "TWS-2 Box." Therefore the transportability of TWS-2 is much better than that of TWS-1. An MS-DOS™ program running on the PC-9801™ was written to control the TWS-2 Box and the video CODEC.

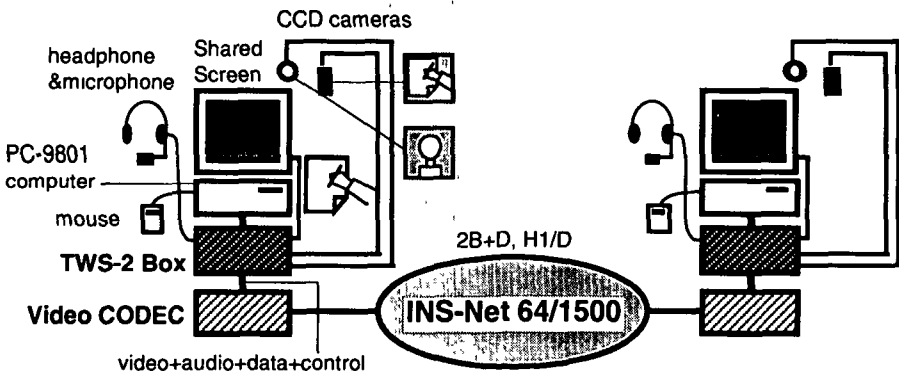


Fig. 5 System Architecture of TWS-2

Fig. 6 shows the appearance of a TWS-2 terminal in use. A headphone with a small microphone is provided for voice communication. Like TWS-1, TWS-2 provides two CCD cameras, one to capture the user's face image and another to capture the desktop image. The TWS-2 Box provides an *external video input port* that can be used to show recorded video clips by connecting a video player. This port is activated by switching the *video source* from the desktop CCD camera to this external video input port.

<sup>6</sup> As a video CODEC, we are using NTT's product, FaceMate™ FM-C700. FM-C700 provides B, 2B, H0, and H1 channel interfaces and interfaces with INS-Net 64 and 1500.

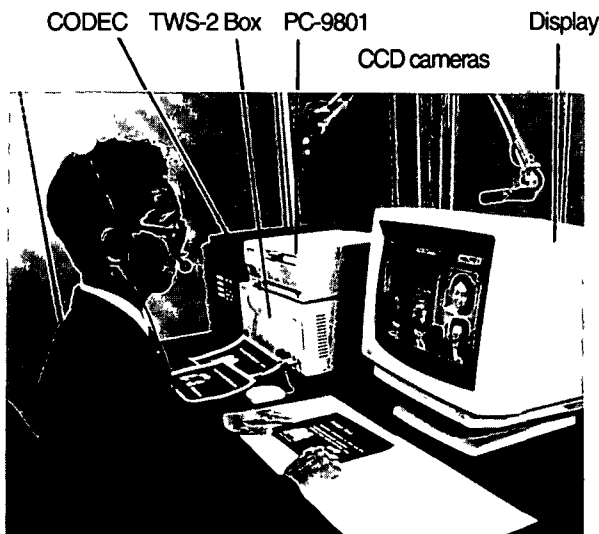


Fig. 6 Appearance of TWS-2 Terminal

The most important design requirement of TWS-2 was to implement the desktop overlay service with the ClearFace interface using only *one video channel* and *one video CODEC* for each terminal. It is not realistic to transmit more than one video channel across the 2B+D interface of INS-Net 64 because the speed and quality of each video image would be significantly degraded. The idea of connecting TWS-2 terminals with more than one INS-Net 64 line was turned down because it is not economically reasonable. How to support the full functions of ClearFace with only one video channel and one data channel was a severe design challenge. To solve this problem we devised a new video processing technique.

Face and desktop video processing flows at each terminal are as follows:

- (1) Insert own face image into own desktop image using the picture-in-picture function. If desired, the face image can be set to "translucent" mode instead of Picture-in-Picture.
- (2) Exchange the resulting video image of step (1) with the remote terminal.
- (3) Translucently overlay own image (1) and the image received from remote terminal. In this step, if the face image is set "opaque" mode, mask the face image and translucently overlay to the rest of the image.

If the face images are moved, resized, or their modes are changed between "opaque" and "translucent", the necessary information (e.g. the new coordinates of the moved face image) are exchanged between the control programs via a data channel to keep the display the same at both sites.

## 4 Multiuser Interface of TWS-2

The multiuser interface of TWS-2 is based on the desktop-overlay function and the ClearFace technique (Ishii and Arita, 1991). Fig. 7 and 8 show typical screen images of TWS-2 in use. In Fig.7, user A and B are discussing the system architecture using hand drawing and gestures. In Fig. 8, user A is teaching calligraphy to user B using red ink to correct B's strokes made in black.

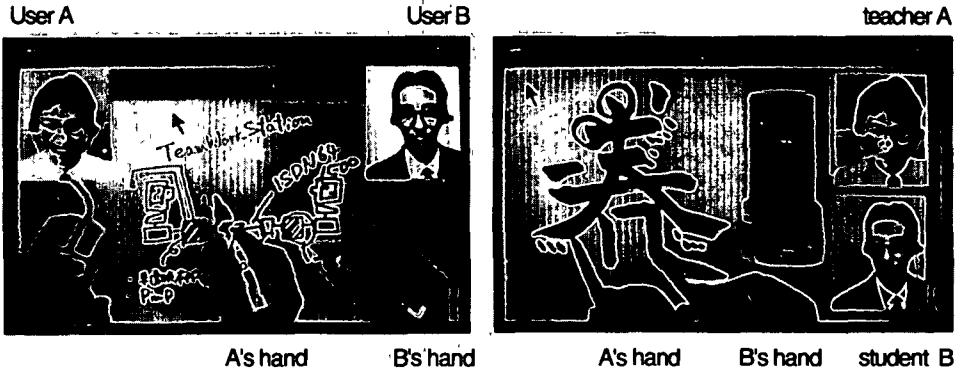


Fig. 7 Design Session via TWS-2

Fig. 8 Calligraphy Lesson via TWS-2

### Computer vs. Desktop

The design focus of original TWS-1 was on sharing the information stored in computers. The name "TeamWorkStation" reflects this computer-centered view. The TWS-1 platform is a workstation, and we extended it to incorporate the information on the physical desktop.

Although the principle of "translucent video overlay" is independent of any level of computer architecture (application, window system, operating system, hardware), the TWS-1 prototype depended on the multi-screen architecture of Macintosh to realize the smooth transition of information between individual and shared screens.

The experimental use of TWS-1 for various conceptual discussions and technical design sessions by the subjects led us to observe that computer-screen overlays were seldom used. Most work was done in the desktop-overlay mode. Pointing and marking were usually done by hand (overlying own desktop) rather than the mouse cursor (overlying own screen). Even though the documents were stored in Macintosh™ computer files, users preferred to print and share the hard copies (papers) as desktop-overlays rather than using direct computer screen overlays.

This behavior has several explanations. If the collaboration faces tough time constraints, the *speed* of information manipulation is critical. In realtime sessions, even if the users are Macintosh™ experts, computer operations take too long and



prevent smooth human-human interaction<sup>7</sup>. Generally speaking, sheets of paper can be manipulated much more quickly than the equivalent computer files. It is also quicker to point to the part of remote documents with own finger than using a mouse pointer. Marking and annotating with a pen is also quicker and easier than using computer programs. Only when data stored in computer memory is extensively manipulated with an application program, does it make sense to share the computer screen directly.

## Desktop-Centered Design of TWS-2

Based on these observations of TWS-1 use, we concluded that the support of desktop images is more important than the support of computer screens. We decided to make "desktop overlay" the basic service of TWS-2, and to make "screen overlay" an option. This decision led to the one screen architecture of TWS-2 instead of the two screen architecture (individual and shared screens) of TWS-1<sup>8</sup>.

The desktop-centered design of TWS-2 means that TWS-2 is closer to the service represented by videophones than computers. The PC-9801™ computer of TWS-2 is mainly used to control the video processing in the TWS-2 Box, and the video CODEC. Direct sharing of information stored in the PC-9801™ or any other computer is not supported as the basic TWS-2 service. However, there are two ways of directly sharing data.

- (1) Use *screen sharing software* while overlaying desktop video images with the shared computer screen.
- (2) Down-convert the computer screen video signal into NTSC, and input the video signal to the *external video input port* of the TWS-2 Box.

Solution (1) requires the existing screen sharing software to be modified so that it simultaneously executes with the TWS-2 control program while sharing the data channel of N-ISDN. Preliminary tests of the two solutions found that solution (1) allows users to share the computer screen at high resolution, while (2) offers only limited quality displays.

## ClearFace Interface

Fig. 9 illustrates the mouse operations of TWS-2 that move, change the size, and switch the transparency of the face windows. All the data of these mouse operations are transmitted to the control program running on the other terminal via

---

<sup>7</sup> For a similar reason, we implemented simple bitmap paint editor (TeamPaint) instead of object-oriented draw editor as the application for an electronic whiteboard (ClearBoard-2) in realtime meeting environments (Ishii, Kobayashi and Grudin, 1992).

<sup>8</sup> Another reason behind this decision is that normal desktops are too limited to support the multi-screen architecture.

the one ISDN data channel to realize the WYSIWIS (What You See Is What I See) principle (Stefik, 86) on both screens.

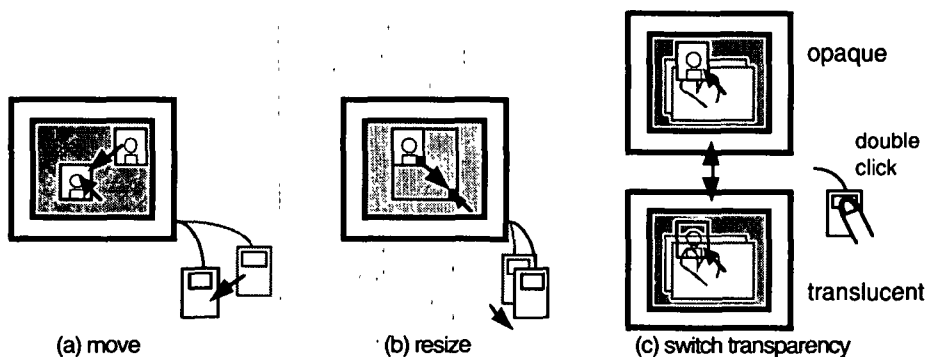


Fig. 9 Operations to Face Images (Move, Resize, and Switch)

Tests of the ClearFace interface confirmed that there is little difficulty in visually separating the translucently overlaid video layers (face and drawing surface). When a subject looked at one layer, he/she found it relatively easy to ignore the other. However, some users said that they preferred to see the partner's face clearly rather than seeing both the face and the desktop image behind it. Therefore, we provided the function that switches the face images between opaque and translucent by double clicking the mouse button (see Fig. 8 (c)).

### Who's Face Do You Want To See?

When we demonstrated TWS-1, we were often asked why the user need to see his/her own image on the shared screen. Some of them told us that just the partner's face image is enough, and some other people pointed out that the face images did not add any value to the shared workspace. In order to respond to these various comments, TWS-2 provides users with a menu offering three modes: (a) two face widows, (b) one face widow and (c) no face widow as illustrated in Fig. 10.

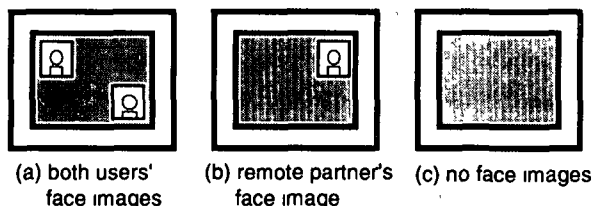


Fig. 10 Selection of Face Images to be Displayed

While mode (a) and (c) provide the WYSIWIS interface, mode (b) does not. In mode (b) each user sees the partner's face image in the face window. We are interested as to which mode will be used most frequently in future tests.

## Limitation of Strict WYSIWIS Over Distance

In a geographically distributed environment, it is impossible to keep the WYSIWIS principle in a very strict sense because of the transmission delay caused by the distance. Especially in multimedia groupware such as TWS-2, delayed and jerky video images are inevitable because of the limitations in the communication bandwidth and the current CODEC technology<sup>9</sup>. Resolution of the remote video image is degraded from NTSC (525 lines/picture) to CIF (352 pixels/line x 288 lines/picture). Therefore, the results of overlaying the realtime local video image with the delayed, jerky, and low-resolution remote video image can never be same at the both sites. Thus strict WYSIWIS can never be achieved. One of the TWS-2 field test goals is to understand the effect of this limitation on working efficiency.

Audio is not a major problem because the delay of the remote partner's voice is negligible, and WYHIWIHN (What You Hear Is What I Hear Now) is almost perfect. In the experiments, no subject complained about voice delay. Asynchronism between voice and video can be a problem in videophone-like applications because users notice the motion of lips and the voice are not synchronized. However, in the desktop-centered application of TWS-2, we found asynchronism between hand gestures (video) and voice is not a serious problem in the experiments described in the next section. We discuss the advantages of TWS-2 over videophones in the following section.

## 5 Initial Findings of TWS-2 Experiments Using INS-Net 64

We implemented the first TWS-2 prototype in October 1992. Two TWS-2 terminals were successfully connected via INS-Net 64 on October 20, and via INS-Net 1500 on November 30. We confirmed that the TWS-2 Boxes functioned properly.

We first tested TWS-2 by connecting our offices in Tokyo and Yokosuka by INS-Net 64 at the end of December. All subjects who joined this experiment were very excited to see that a virtual desktop could be shared over a distance of 60 km. This means that TWS-2 can connect most big cities in Japan via INS-Net 64. This

---

<sup>9</sup> When we used INS-Net 64 in our experiments, we assigned 112kbps for the video. In that situation, the frame rate was about 10 frame/sec, and the transmission delay was about 200-300 msec. In the case of H1 (INS-Net 1500), the frame rate became about 30 frame/sec, and the delay was only 10 msec.

excitement could never be gained through small experiments in the same building. Shared drawing activities were much smoother than anticipated by videophone users, most of who had slightly negative impressions of the quality of realtime video communication over INS-Net 64. We believe that TWS-2 can be used for real work over distance. The success of the Tokyo/Yokosuka experiment convinced us of the potential of TWS-2 as a new N-ISDN service.

The main purpose of these experiments was to answer the question: "*Is TWS-2 usable under the bandwidth limitation of 2B+D?*" Before we started the TWS-2 experiments, many people dubious about the ability of INS-Net 64 to support realtime activities because of their previous experience with the jerky displays of videophones. To find out how subjects who were not involved in this project evaluated the usefulness of TWS-2 in a work-like setting, we conducted the following experiments.

### Experiment of Remote-Controller Design Using TWS-2

**Tasks:** Pairs of subjects were asked to design an integrated remote controller for TV and VCR (video cassette recorder), or air conditioner and gas fan heater.

**Subjects:** Three pairs of subjects (6 people in total) were recruited from other research groups in our laboratories. All had an engineering background. Although none of them had ever used TWS, they knew about TWS through paper reports and video clips.

**Conditions:** Individual TWS-2 terminals were installed in different rooms on the same floor. They were linked through actual INS-Net 64 connections, and the bandwidth of INS-Net 64 (2B+D) was allocated to each media as follows:

- Video      112 kbps
- Audio      16 kbps
- Data        9.6 kbps

Each pair of subjects spent about 20 minutes doing the design task using TWS-2. Each subject was provided with color markers, an eraser, and a whiteboard placed on the desktop surface. Actual remote controllers for the TV, VCR and air conditioner were also provided. All work activities were videotaped for later detailed analysis. After the design sessions, each subject filled in a questionnaire, and was briefly interviewed about what they liked and disliked about the task environment. A questionnaire was used to obtain a subjective view of TWS-2's usefulness.

**Results:** In the post-task interview, the subjects generally commented that they were absorbed in the task, and enjoyed interacting with the partner. Although the subjects noticed some delay and jerkiness in the remote desktop video image, these