THE UNIVERSITY
of LIVERPOOL

Videoconferencing on Unix Workstations to Support HelpDesk/Advisory Activities

A Report to the Advisory Group on Computer Graphics

Steve Morgan

Mary Thorp

Computing Services Department

The University of Liverpool

P.O. Box 147

Liverpool

L69 3BX

Contents

Acknowledgements

The authors would like to acknowledge the support of members of the Computing Services Department at Liverpool who contributed in various ways essential to the work described in this report. Particular thanks go to Chris Wooff and Allyson Matthews who took part in some of the case studies reported on in Chapter 4 and made useful comments on the contents of the report.

Funding to support the work was provided under the SIMA initiative which is administered by the Advisory Group on Computer Graphics. AGOCG is funded by the JISC and the SIMA project is funded by the JISC/NTSC under the New Technologies Initiative.

JSM/MT
April 1995

Report Summary

This report is based on work carried out to investigate the use of desktop videoconferencing products to support Helpdesk/Advisory activities in a Computing Service environment. The report provides background information on products available on Unix systems and details the use of the SUN Showme suite of videoconferencing software. Particular mention is made of the experienced gained in using the facilities to support those with hearing difficulties.

Although the report is based mainly on experience of videoconferencing over a Local Area Network, it should also have relevance to those wishing to exploit desktop technology over Wide Area Networks.

It is hoped that the report will be useful to those wishing to exploit this new technology and will provide some foresight into the possible benefits, and to the pitfalls, which may be encountered.

Chapter 1

Introduction/Background

This report is based on the findings of a project which investigated the use of the SUN-based videoconferencing product called Showme in supporting Helpdesk and Advisory activities in a Computing Service environment. In August 1994 we already had many of the basic elements required for videoconferencing (described later) and although we entered the project with a healthy scepticism we felt that it would be useful to attempt to make use of the technology in a particular working situation since this would be more enlightening than merely looking at the technology itself. We were particularly keen to see if videoconferencing could be of benefit to those with hearing difficulties who would obviously have problems in work situations which normally require telephone contact.

The report places the present project in the context of the rapid developments taking place in this field by giving general background information where appropriate and by comparing the Showme product with other software offerings available on Unix platforms. The comparison is based mainly on paper evidence since a full evaluation of competing products on Unix workstations was beyond the scope envisaged for the project. Although we concentrated the study on Unix-based products, we have had the opportunity also to compare with Intel's Proshare which is a Windows-based product.

In this chapter we look at general issues associated with setting up videoconferencing facilities and the environment in which the current project was carried out. Chapter 2 gives a fairly detailed account of the SUN products used since it is felt that this will give the reader with no previous experience of videoconferencing a good grasp for the facilities provided by a typical system.

In Chapter 3 we describe some videoconferencing software items that are available in the public domain and assess their value since this information will be of interest to those who cannot afford the currently quite expensive commercial offerings but who wish to experiment with the technology. Also in Chapter 3 we describe Silicon Graphics' Inperson software

package on which we were able to carry out a paper evaluation, and, as indicated above, Intel's Proshare package for the PC/Windows platform.

In Chapter 4 we include evaluation tables giving a tick list of features present in videoconferencing products. This will serve to give an overview of what can be expected in different products as well as information on the specific products looked at.

In Chapter 5 we describe the various experiments which were carried out to determine the usefulness of Showme in supporting Helpdesk/Advisory activities and the conclusions drawn. We also give some indication of the performance issues associated with our project. Finally Chapter 6 summarises the conclusions drawn from the project and gives some suggestions for further work.

We have included various appendices which contain useful information that we collected during the lifetime of the project and which did not fit into the logical pattern of the the report. Readers will also find the following SIMA Reports valuable sources of information:

The Dos and Don'ts of Videoconferencing in Higher Education by The HUSAT Research Institute, Loughborough University of Technology, SIMA Report Number 4;

The Dos and Don'ts of Videoconferencing - Experience from LIVE-NET by Phillip Riebold, University College, London, SIMA Report Number 7.

Video Conferencing and Learning in Higher Education by Lynne Coventry, Heriot Watt University, Edinburgh

The Range of Videoconferencing Systems

Videoconferencing can mean different things to different people. The term covers a wide range of hardware/software facilities from the use of person-to-person videophone right through to the use of full-blown videoconferencing studios with expensive cameras and possibly staffed by technicians. It is possibly more appropriate to describe many of the

products available as "systems for remote collaboration" or some such phrase since so-called videoconferencing products often provide much more than just audio/video links. However, the term *videoconferencing* is now in common use and we shall use it in its wider sense in this report.

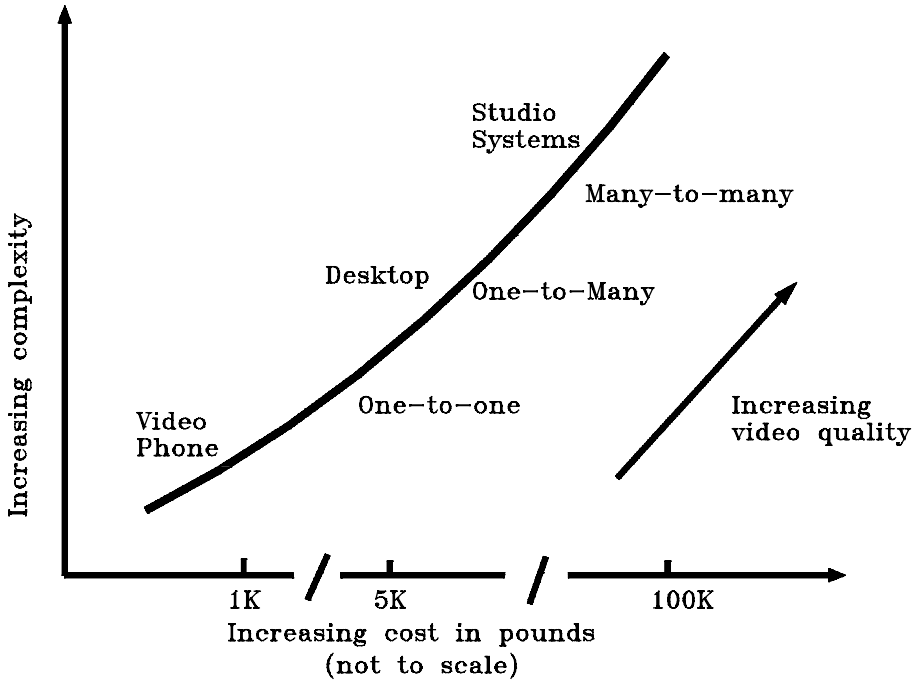


Figure 1.1 Range of videoconferencing products

Figure 1 shows roughly how the range of facilities relates in terms of the number of people involved in a conference and the cost of the facilities. At the low end, a simple videophone, providing audio and video communication, can be used for one-to-one conferencing. At the high end, high-bandwidth connections are used with digital and analogue circuits to connect videoconferencing studios where groups of people at several sites can hold meetings. Such systems are expensive to install and maintain (in the hundred thousand pound range) and have the disadvantage that they usually have to be booked some time in advance.

In this study we are interested specifically in the use of *desktop* videoconferencing products. These consist of relatively low-cost hardware/software packages which run on personal

computers or workstations and which can be connected by various means. This sort of product allows users to communicate with others from their office on a standard PC or workstation thus potentially bringing the technology within reach of a mass user base. Since PCs are present on an increasing number of desks in organisations and are becoming more and more available to students, desktop videoconferencing products which can be installed on existing equipment have the potential for spreading videoconferencing to a very large user base.

At present pricing is at far too high a level to allow such widespread use but costs will inevitably decrease as the technology develops. Within five years it is quite likely that most members of staff in HE and large numbers of students will have access to a multimedia PC with many of the components of a videoconferencing system built-in. It is also quite likely (and crucial) that the underlying means of communication (i.e. the 'network') will be transparently capable of coping with the volumes of traffic generated by multimedia applications of this sort. If the desktop videoconferencing technology delivers on its promise and the products do indeed enhance people's working lives and increase the level of 'communication at a distance' then we could see dramatic developments in working practices in the next few years. However, one of the important issues addressed in this study is whether the technology is sufficiently advanced to make it useful in working environments NOW!

Videoconferencing equipment

Current commercially available products for desktop videoconferencing usually consist of a package consisting of various hardware components and a set of software items.

Hardware

The hardware components required will vary depending on how well equipped the target workstation or PC is, but in general will consist of the components illustrated in Figure 2.

Camera — A small video camera which can be attached (usually) to the top of the workstation/PC monitor. The cameras usually have a lens shutter which allows the picture to be switched off manually. Closing the lens shutter switches off

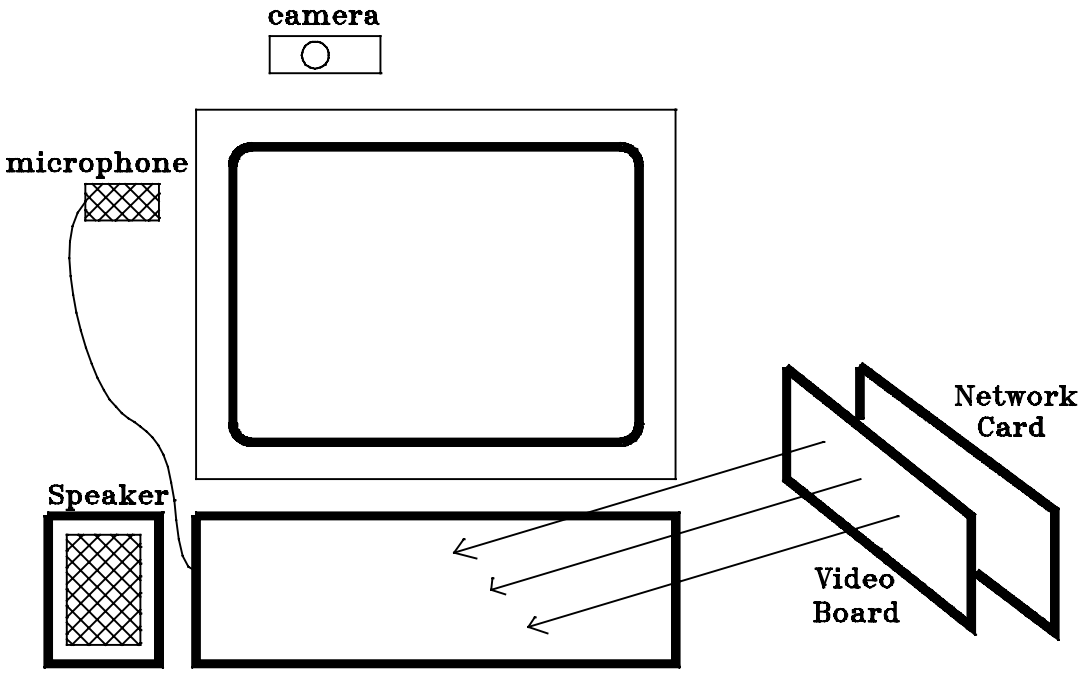


Figure 1.2 Videoconferencing hardware

the power to the camera on some models. The cameras we have seen come with a DC power supply which irritatingly must be plugged in to their own 13 Amp socket. Multimedia capable workstations with integral cameras will not suffer from this problem.

Microphone

The ones we have seen tend to be rather fragile and/or compact looking. The SUN IPX microphones that we used were an inch and a quarter square and had their own internal batteries.

Speaker(s) Speakers are often built in to today's workstations but they can be of poor quality. External speaker(s) will often provide better quality sound. Sometimes an external amplifier is required.

Earpiece/earphones

These can be useful if the speaker quality is poor or if the equipment is to be used in noisy environments or in situations where several staff are working side-by-side.

Video capture board

This piece of specialised hardware which must be installed in the workstation/PC is essential for converting the analogue video signal from the camera to (possibly compressed) digital data at a high enough rate to maintain a moving picture.

Network board

This might be an ISDN card (if videoconferencing is taking place over ISDN circuits) or an Ethernet card if TCP/IP is used.

As mentioned above, products for different platforms will vary according to how many of the above components are already built in to the target workstation/PC. Currently many vendors are offering upgrade kits which consist of one or more of the above components which add in the capability for videoconferencing to existing under-equipped workstations or PCs. This is a reflection of the immature state of the market in that currently installed workstations/PCs are not yet, in general, multimedia capable.

Software

The software components to some extent provide a means for controlling the various hardware components described above but also provide additional facilities. In general, the following components are presented in the user interface. The description given here is brief. A more detailed account of a typical system is given in Chapter 2.

Conference manager

This provides overall management of the conference.

Address book

This is a tool for managing a list of addresses of potential conference participants via a set of aliases.

Video This component controls the video settings and allows the user to vary different operating parameters.

Audio This component controls the audio settings and allows the user to vary different operating parameters.

Shared whiteboard

This component provides the conference participants with a screen area which is displayed at each location and which is simultaneously kept up-to-date. Various tools are provided to allow participants to share text, drawings, images, documents etc.

Shared application

This is a facility which allows one participant to launch an application in a window which appears on all workstations in the conference. Typically, tools are provided to allow participants to take control of the application where appropriate and where the initiator allows.

In a well-designed system all the above components would be expected to be integrated so that the user sees a single interface with the various components controlled by the conference manager. In practice several systems we have seen seem to fall short of this ideal.

Carriers

An important aspect of running desktop videoconferencing software is that of the medium over which which the basic communication is to take place. Where videoconferencing is to be used on a site such as a university campus (local area), the two main possibilities at present are to use a Local Area Network (LAN) or to use ISDN which is an emerging

technology based on telephone connections which is becoming more widespread.

Full screen broadcast quality video in its raw state (at 25 frames a second) requires transmission speeds in the 100-200 Megabit/second range. This is far beyond the capacity of current LANs (up to 10 Mbit/sec) or basic ISDN connections (up to 128 Kbit/sec). The only way to reduce the bandwidth requirements to reasonable proportions is to use a *codec* (coder/decoder) to compress the video stream in some way before transmission and to decompress at the receiving end. Compression ratios of well over 100:1 can be achieved by a combination of image compression and frame-change coding but this still leaves bandwidth requirements at quite a high level. Typical codecs tend to introduce artifacts (features that are not in the original picture) and perform less well when there is a lot of movement between frames. A description of some relevant compression methods is given in Appendix 2.

Further reduction in bandwidth requirements, beyond that gained by compression, requires reduction in the size of the video picture and/or the frame rate. Two standard formats that are widely used are: FCIF (Full Common Intermediate Format) which uses a 352 x 288 pixel rectangle at 15 frames per second; and QCIF (Quarter CIF) which uses a 176 x 144 pixel rectangle. It can be seen therefore that at current communications speeds the quality of transmitted video is going to be relatively poor. This is illustrated vividly in Appendix 5 where we have included a table of videoconferencing bandwidth requirements for various parameter values.

LANs

LAN technology uses a broadcast network. Its main advantages over ISDN are that it is potentially fast (up to 10 Mbit/sec), and it is usually already available in many organisations and so no cost is required either in terms of manpower or money in order to provide the basic carrier. It also has the advantage of being able to support inexpensive multiway conferencing (see below). However, because it is a broadcast network technology, it cannot guarantee constant speed of communication, and indeed when a LAN becomes heavily loaded with other traffic, videoconferencing can become unusable. As will be seen later, though, current LANs can be used with some effectiveness under favourable conditions.

ISDN

ISDN comes in two main forms called Basic Rate and Primary Rate. Basic Rate provides two 64 Kbit/sec channels for data and a control channel. Primary Rate provides thirty 64 Kbit/sec channels for data and a control channel. Present desktop systems are geared (because of their price range) to Basic Rate ISDN and so we restrict our comments to this in the present report.

The overriding advantage of using ISDN over a LAN is that it provides a constant speed of communication which is dedicated to the conference participants. Typically, one 64 Kbit/sec channel would be used for video and the other for audio communications. The main disadvantage of ISDN, apart from its lower speed, is that, usually, current telephone switches and possibly wiring cannot be used and special connections have to be installed by a telecommunications supplier (e.g. BT, Mercury). These connections are relatively expensive (at the time of writing, from BT, they are £400 installation and £84/quarter rental plus the cost of the calls).

The lower 64 Kbit/sec speed of ISDN relative to a LAN means that video transmission frame rates tend to be around 1-5 frames/second giving very poor motion reproduction. On the positive side is the fact that the dedicated sound channel gives excellent voice communication.

A Basic ISDN connection only supports one-to-one conferencing with another similar connection. Multi-way conferencing requires connection to extra equipment (Multipoint Control Units) which is likely to be expensive (we have not been able to ascertain how expensive at the time of writing).

Ideally it would be possible for an institution to use its existing internal telephone system (PBX) but, in our case, enquiries indicate that our internal system is not capable of supporting ISDN and an expensive upgrade to our exchange would be required costing tens of thousands of pounds. It seems at present that the only option for us (and probably other similar sites), if ISDN is required, is to install single connections via BT.

[Note: The MICE (Multimedia Integrated Conferencing for European Researchers) project

(URL: http://www.cs.ucl.ac.uk/mice/mice.html) has been developing facilities to interface ISDN users to the Internet. Readers are recommended to consult the MICE WWW server for further information. Dan Kegets' ISDN page
(URL:http://alumni.caltech.edu/~dank/isdn/) contains an abundance of ISDN information and includes sections on videoconferencing and ISDN shareware. "The joys of obtaining ISDN in the UK" (URL://134.220.198.66:8000/isdnjoys.html) is worth exploring. This site is connected to the Internet via ISDN.]

For the present study we used our local Ethernet TCP/IP based network which the Showme software had been designed to run on. Part of the study was directed at seeing how the software would run in this environment.

Multiway Conferencing

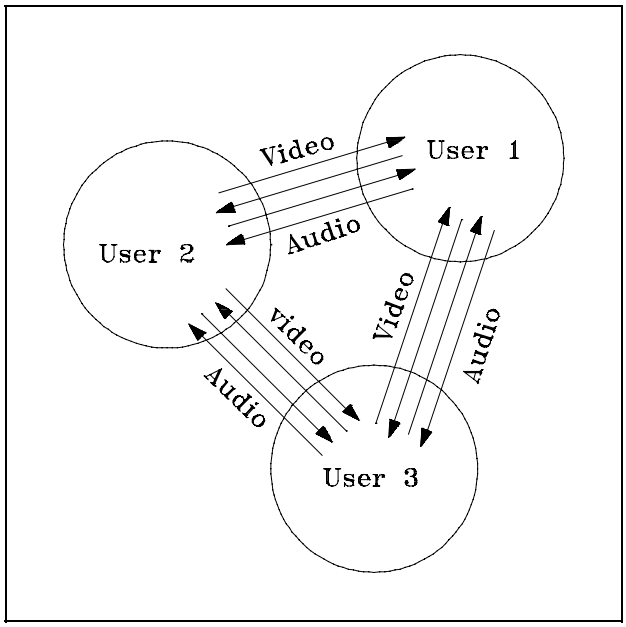


Figure 1.3 Three-way communication

Videoconferences can be set up for one-to-one communication, one-to-many, or many-to-many. As stated above, using ISDN for anything but one-to-one is relatively expensive at

present especially for a project such as this one. However, using LAN technology it is possible at no cost to create many-to-many conferences (and hence also one-to-many) using the so-called *multicast* technology.

Multicasting is a networking solution to the problem created when several people need to communicate. For example, as illustrated in Figure 1.3, if three people needed to communicate via video, each person would have to open two channels to talk to each of the other two participants. Thus six channels would be required for the conference. This, of course, would double to twelve if audio communication was also required. If there were four people in the conference then 12 video and 12 audio channels would be required. In general if n people were in the conference then n(n-1) channels would be required for each medium of communication. It can be seen that multiway conferencing would soon totally clog up a network if implemented in this way. The solution devised for this problem is called multicasting and it simply means that each participant in a conference opens only one channel of communication for each medium and the network nodes take care of the distribution of the data to all other participants.

It should be stressed that multicasting merely reduces the problems since the network still has to carry a huge amount of traffic where there are large numbers of users communicating with each other. A more detailed description of how multicasting is used with the ShowMe software is given in Appendix 3.

Multicast operating system patches are available for all major Unix workstation platforms and are delivered as standard on some platforms (e.g. SUN Solaris 2.3 and later). We did carry out some experiments with the multicasting facilities supported by Showme, LBL and IVS but these were not central to our study since the Helpdesk interactions we looked at tended to be one-to-one. However, we could envisage scenarios where three-way conferencing would be very useful. For example, a user may have contacted an advisor who then wished to bring one or more specialists in on the consultation.

Standards

At present, most videoconferencing must take place between participants who are each using the same videoconferencing software package since packages from different vendors tend to use their own proprietary compression schemes and control protocols. This may be acceptable for internal organisational communication where a de facto standard hardware/software policy has been adopted but is obviously fairly prohibitive where different organisations, possibly in different countries, or participants on different platforms, wish to conference.

The answer is, of course, an international standard for videoconferencing communication. The relevant standard in this area, which is currently being adopted by the computer industry, is called H.320. It is a ITU-T (International Telecommunications Union - Telecommunications Standards Section), formerly CCITT, standard which encompasses several subsidiary standards for: communications protocol (H.242), audio/video synchronisation (H.221), video compression (H261) and audio (G.728, G.722, G.711) transmissions. The H.261 standard was developed to work over Px64 digital channels (64 Kbits/sec to 2.048 Megabits/sec). H.320 also includes standards to handle multi-way conferencing with ISDN (H.231 and H.243) and will also eventually include a standard covering document sharing or whiteboard (T.120).

H320 is a relatively recent development and products which support it are only just beginning to appear. In fact, at the time of writing, we have seen several glossy brochures claiming support for it, but we have not seen a delivered product. If other new standards are anything to go by it is probably premature to assume that H320 will provide a short-term answer to cross-product and cross-platform videoconferencing.

In fact vendors have been developing their own ’standards’ and it is possible that one or more of these could develop into a *de facto* standard if one of these vendors begins to dominate the market. CELLB for video compression from SUN and Indeo from Intel are two such candidates. We found in our study that video performance using Showme which uses CELLB was noticeably better than that using the public domain item IVS which uses H261. This may be an indication that the H320 group of standards could become outdated quickly and will increase the likelihood that a dominant vendors product (likely to be in the Windows market)

will become a *de facto* standard.

Software Available for Unix Workstations

As mentioned previously there are several products available for the main Unix workstation platforms as shown in the following table. The table is intended only to give a quick overview and overall comparison of the products on offer. The most well-known public domain items are also included for comparison.

Product	Platform	Whiteboard	Shared App	Standards/Encodings
Communique	Unix	Yes	Yes	CellB, JPEG, Indeo
DECSpin	DEC Ultrix	No	No	JPEG, G711
InPerson	SGI Irix	Yes	No	HDCC, G711
IVS (Public Domain)	Unix	wb	No	H261, PCM...
LBL (nv,vat,sd) (Public Domain)	Unix	wb	No	
Picture Window	SUN	Yes	Yes	
PSVC (Paradise)	SUN, HP	Yes	Yes	Motion- JPEG
ShowMe	SUN, Other?	Yes	Yes	CellB, G711

Source: WWW pages on MICE server ("Desktop Videoconferencing Products")

The table includes an indication of which of the videoconferencing components described

previously are present. As can be seen most of the commercially available products include whiteboard and shared application facilities. The public domain package IVS supports video and audio communications and can be used with *wb*, a public domain whiteboard facility. LBL is also a public domain offering. It includes video, audio, and whiteboard support (via *wb*). Neither of these public domain items have a shared application component. Note that at the time of writing there is no product available from IBM for AIX. IBM have indicated that a product will be available in future.

The last column gives an indication of the video (and possibly audio) compression standards used by the various products. It can be seen that most use proprietary standards at present but some components of H320 are being used. The above table should clearly be read with caution since it will become out of date rapidly.

In the present project we used Showme and therefore can comment on it in some detail. We also obtained copies of IVS and the LBL group and were able to assess them in use to some extent. We were able to carry out a paper evaluation of InPerson from a full set of documentation. We also obtained two copies of Proshare for the PC during the lifetime of the project and were able to assess this relative to the other Unix-based software.

The Project Environment

It is useful here to describe the environment in which the project was carried out since readers can then relate it to their own situations.

The Computing Services Department at Liverpool is housed in two buildings. The Helpdesk and Advisory desk are situated in one building with several members of staff on two floors and the rest of the programming staff and management are housed on the ground, 4th and 5th floors of a tower block which is 5 to 10 minutes walk away. The staff in the tower block are frequently contacted by the Helpdesk and Advisory desk via the telephone to answer various queries. Whilst many queries can be dealt with in this way it is quite common for wrong advice to be given due to the lack of communication of the problem in some way. This can be particularly true when visual information such as an incorrect graph drawn on a plotter

or incorrect formatting of a document is the basis of the problem. In such cases bad advice often results in wasted effort on the part of the end-user who eventually returns to the Helpdesk and is then referred to the expert for face-to-face advice. Referring the user for such advice, of course, frequently involves the 5-10 minute walk between the buildings. There is an obvious waste of time here for the advisor and the user.

The tools provided by desktop videoconferencing have clear potential for easing some of the problems alluded to above. The increased communication provided by video transmission can not only improve a consultation on the personal level but also provide a means whereby visual information pertaining to a particular problem can be given instantly. A shared whiteboard facility can be used to provide instant visual information and the ability for the advisor or the user to launch a particular application and share it with the other has obvious potential. On the face of it videoconferencing is the answer to the Helpdesk Staff's/Advisor's dreams! If desktop videoconferencing eventually becomes a routine way of communicating then there will rarely be a need for people with computing problems to visit a formal Helpdesk or Advisory Desk.

Of particular interest to us at Liverpool is the potential of videoconferencing for helping those with hearing difficulties who cannot use the telephone. One of our local expert graphics advisors who is severely, though not totally, deaf took part in the project and jointly authored this report. An expert who is at a location remote from the Helpdesk and also has hearing difficulties can only be contacted via a third party or via email (both of which are unsatisfactory). However, videoconferencing can potentially provide good communication for the hard of hearing via the video, whiteboard and shared application components. We thus carried out some experiments which sought to assess the true value of these features in this special case. These are reported on in Chapter 5.

The computer service at Liverpool is based partly on a large network of some 200 SUN IPX workstations connected via an Ethernet LAN. Five multimedia upgrade kits for the workstations consisting of a video camera and a video capture board were available and (in August 1994) SUN had just released their Showme videoconferencing software. This seemed to be an ideal base on which to investigate the use of videoconferencing in the work of the

Computing Services Department. As can be seen above (and will shown more clearly in Chapter 2) the Showme software suite is typical of the the sort of tools provided by typical products in this area. We therefore felt that our experiences would have general relevance to anyone contemplating applying videoconferencing technology.

Chapter 2

SUN Hardware and Software for Videoconferencing

In this chapter, we describe the videoconferencing tools used in the project in some detail since we feel that this will give the reader a good idea of what constitutes a desktop videoconferencing system and will provide a reference system for comparing with other products

For this project, we used four SUN Sparcstation IPX systems, which are Sparc 2 based, and one SUN Sparcstation ZX system, which is Sparc 10 based and contains a graphics accelerator. Four of these SUN workstations were located in offices of members of staff and one on the HelpDesk. We upgraded these workstations using multimedia upgrade kits from SUN. The kits consisted of:

SUN camera	This can be attached to the top of the workstation and can swivel up and down, left and right. It has a lens shutter - it is more reassuring for visitors if this is closed when the camera is being used!!
SUN video card	This takes up a single SBus slot. It is used to capture and compress and transmit video data. More details are given later in this section and in Appendix 2.

The workstations we possessed already had audio support in the form of:

SUN microphone	This could be attached to the SUN workstation. However, it was recommended that the microphone should be as far away from the speaker as possible to prevent feedback. Even if there is a separate speaker box, the microphone should be detached from the workstation since it may pick

	up the background hum of the workstation.
SUN built-in speaker or box	The IPX workstations have built-in speaker hardware but the ZX workstations have a separate speaker box.

The videoconferencing software required was:

SUN XIL library	This has low level functions to control applications which take advantage of the SunVideo system.
SUN ShowMe Version 2.0	More details on ShowMe are given later in this section.

The SUNVIDEO System and the XIL Programming Support

SunVideo is a real-time video capture and compression subsystem for Sun SPARCstations. It includes a SunVideo card which supports the capture, compression, storage and transmission of video images. SunVideo is designed to work with the facilities of the SUN XIL 1.1 Imaging Library which provides functions for image processing and image compression/decompression. XIL programs when used with SunVideo take advantage of the compression engine of the video card. So the compression rates are a function of the performance of the SunVideo card.

Decompression of video on a receiving workstation is done totally in software since this particular video card has no decompression functions. Applications such as Showme can access the decompression functions in the XIL library. This has the advantage that a workstation does not need a video card installed in order to receive video but the disadvantage that video performance is partly dependent on the CPU performance of the receiving host. As we found, this can be significant when the receiving workstation is a PC running an X emulator connected to a multi-user host.

Please refer to Appendix 2 if you are interested in the more technical details of the SunVideo card and the XIL library.

SHOWME

Introduction to the ShowMe Software

The ShowMe suite of software has been installed on the SUN system at Liverpool under Solaris 2.3. It allows users on SUN workstations to communicate via live video, live audio, shared whiteboard and shared applications. We used ShowMe mostly on five workstations on which the multimedia kits had been installed.

Documentation

There is one manual for ShowMe and it includes sections on all the different conferencing tools. It is very easy to read and to follow.

The manual states:

"ShowMe is a suite of network conferencing tools designed to help you work effectively with other ShowMe users at other worksites."

Standards

ShowMe Audio is encoded using CCITT standard G.711 but ShowMe Video uses SUN's proprietary CellB compression algorithm. A future version of ShowMe which supports H.320 has been mooted. See Appendix 3 on Compression Methods for further details.

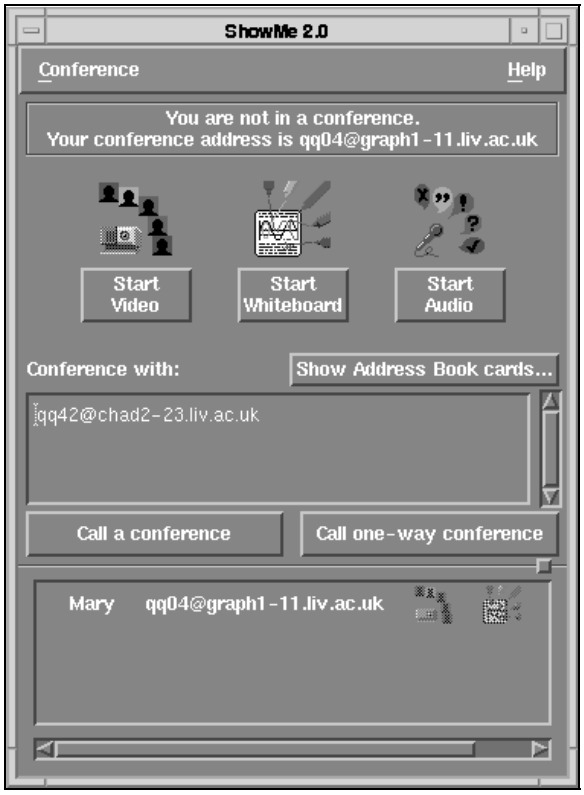
Features

The ShowMe software comes with a conference manager which allows users to set up a conference with other users of ShowMe software. This conference manager can initiate one or more of the tools, ShowMe Video, ShowMe Audio, ShowMe Whiteboard or ShowMe SharedApp. There is also an Address Book feature which is described briefly later. Although it is possible to have multiple participants in a conference, most of our investigations dealt

with the case where there would usually be an initiator of a conference (help desk) and one other participant (consultant dealing with the query).

Calling a Conference

When ShowMe is started, a conference manager window appears as shown in Figure 2.1. To begin a ShowMe conference, one or more of the video, audio or whiteboard icons can be picked to start the different components and then the addresses of users are submitted. Either the full name and address for each participant can be typed in the conference box, for example, jbloggs@yew1.liv.ac.uk, or addresses can be selected from an address book list which can be popped up if required, as in Figure 2.2 below.



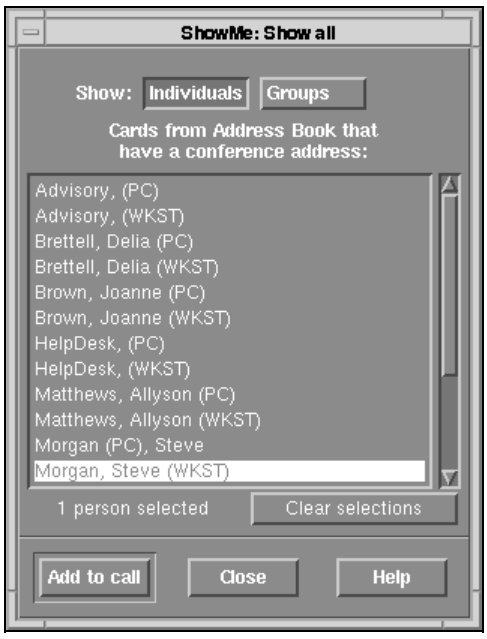


Figure 2.1 ShowMe Conference Manager **Figure 2.2 ShowMe Address Book**

In practice it is useful to store various address books in a public area so that users can copy these to their private area and adapt them for their own use.

Provided that they are running ShowMe in an X window on their workstations or PCs, an invitation panel appears on the screens of the people being invited to the conference notifying them that a conference has been requested. They can either accept or decline the invitation. If they accept, then, presuming all the tools have been selected, a whiteboard window, an audio window and one video window for each participant in the conference appear on the screen. Video windows only appear for those participants who have a camera and video board installed on their workstation.

ShowMe Video

The ShowMe video tool allows conference participants to simultaneously view each other.

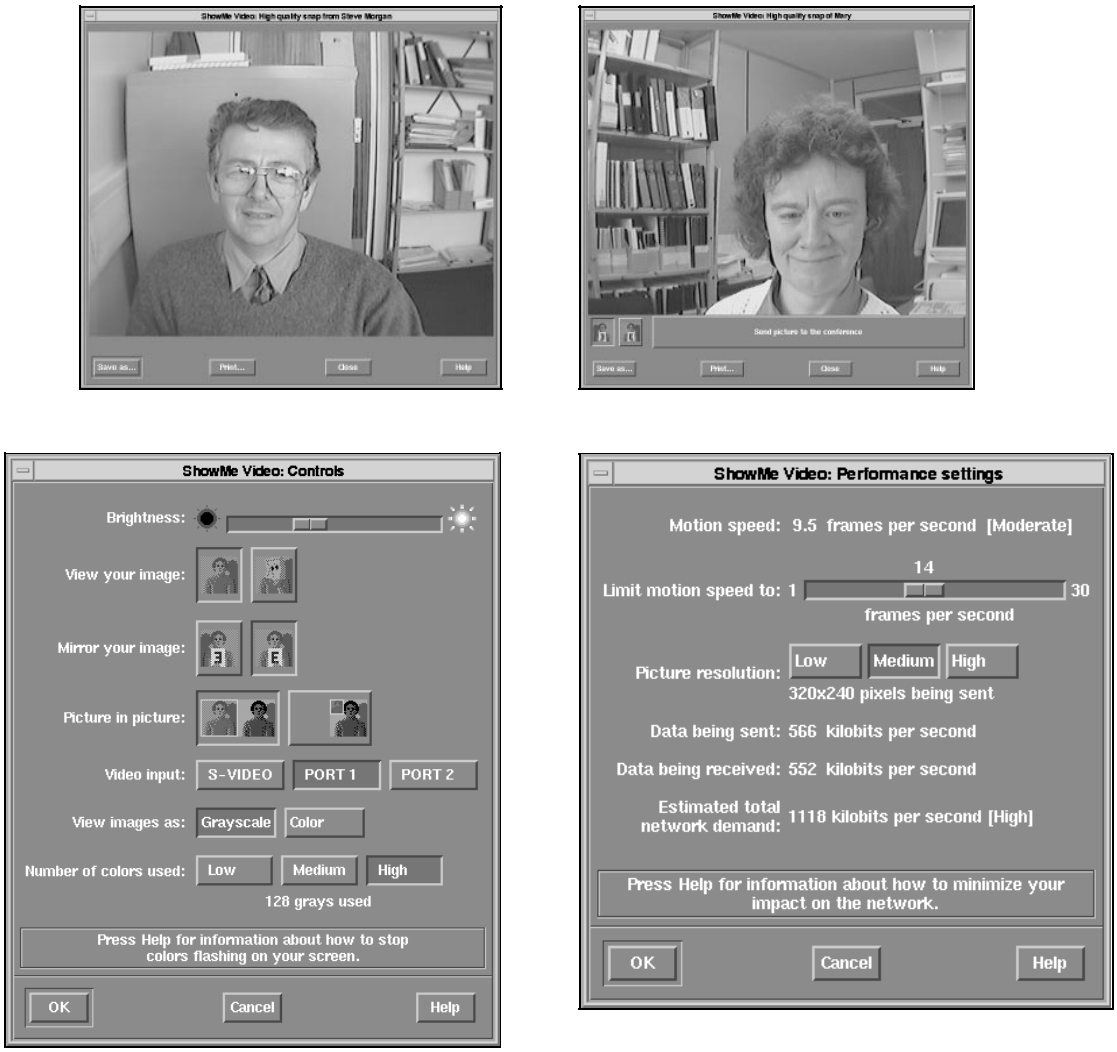


Figure 2.3 ShowMe Video Windows and menus

The ShowMe video window has a status area and this allows the user to keep track of other participants - thus helping to reduce confusion. It is possible, by selecting menu items on the video window, to monitor how the conference is affecting the network. To reduce the load on the network, there are performance options such as the resolution and the motion speed (number of frames per second) that can be adjusted (see Figure 2.3) - these are important facilities in a video conferencing package since the video transmissions can require a substantial portion of the bandwidth available in a network. There are also colour controls in ShowMe video. If the number of colours is reduced, then the required bandwidth is reduced and also colour flash on the screen is prevented. If the images are viewed in grayscale instead of colour, the number of colours used is reduced further. Zoom controls exist to reduce or increase the size of the video windows. Another control available (which we did not use ourselves in this project) is to switch to a different input port so one could use a VCR or even another camera instead of the current camera.

ShowMe video has a 'snapshot' facility; this means that the camera can be used to either capture an image and then send that image to the other conference participants or take a picture of another conference participant which is showing in a video window. The pictures can be saved and printed.

More control options include mirroring of images, hiding images of others and muting outgoing video. Hiding images of others reduces the load on the local CPU which does not have to decompress the images. Muting outgoing video reduces the impact of the conference on the network and also provides privacy (you may not want the boss to see your visitors!).

ShowMe Audio

The Showme Audio tool is designed to help users talk to one another using the microphones and speakers attached to workstations. It provides controls for adjusting the incoming volume on the speakers and the outgoing volume on the microphone. For privacy reasons, it is essential to be able to prevent other users from hearing outgoing speech and ShowMe audio has a 'mute' button which provides this facility. Another facility in ShowMe audio is the ability to switch to full or half duplex - this means that either everyone in the conference can

be heard all the time (full duplex) or that only the person who is shouting the loudest can be heard (half duplex). Using half duplex reduces the load on the network and removes the problem of feedback.

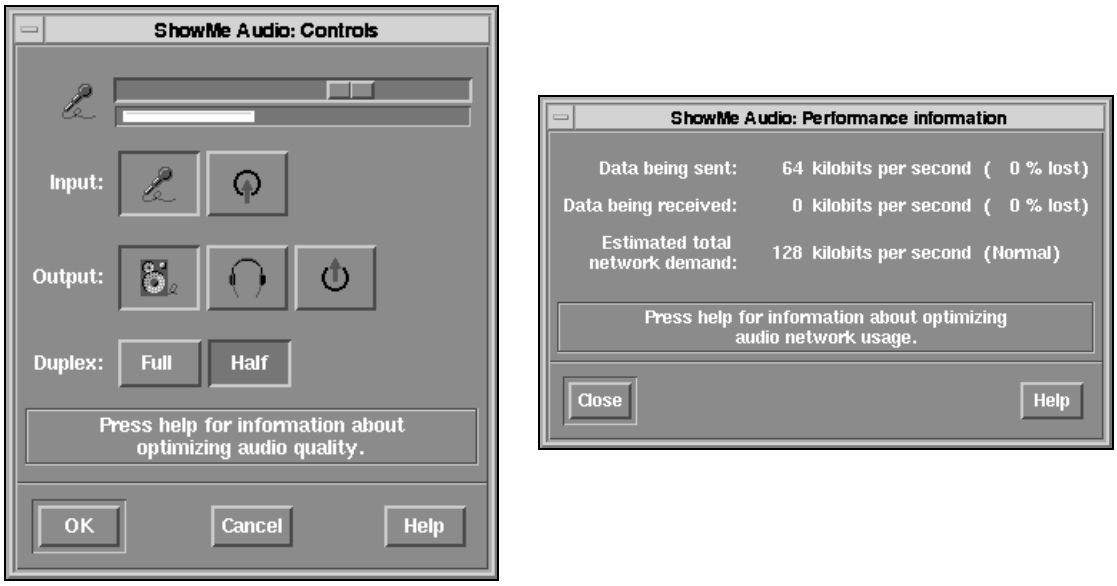


Figure 2.4 ShowMe Audio menus

It is possible to switch to different output and input ports. This means one can connect a device to a port and play a recording instead of using a microphone and one can use earphones instead of relying on the speaker or speakerbox.

The impact of audio on the network is substantial and, as for the video component, ShowMe provides monitoring information in a window and some menu options to allow users to minimise this impact. Other information provided includes the status of other participants. We found this useful since it told us when someone had switched off their audio component, which could be done intentionally, but could also be done by accident. Good practice would dictate that conference participants inform the other users if they intended to close down various tools but of course this would not always happen.

ShowMe Whiteboard

All the participants can share a common drawing board. There are tools to enable drawing

of lines, text, boxes, and circles, freehand drawing, erasure of part or all of the drawing, highlighting, importing of images etc. It operates as if the participants are facing a real whiteboard armed with pens and pictures available to illustrate ideas etc. See Figure 2.5 for an example whiteboard.

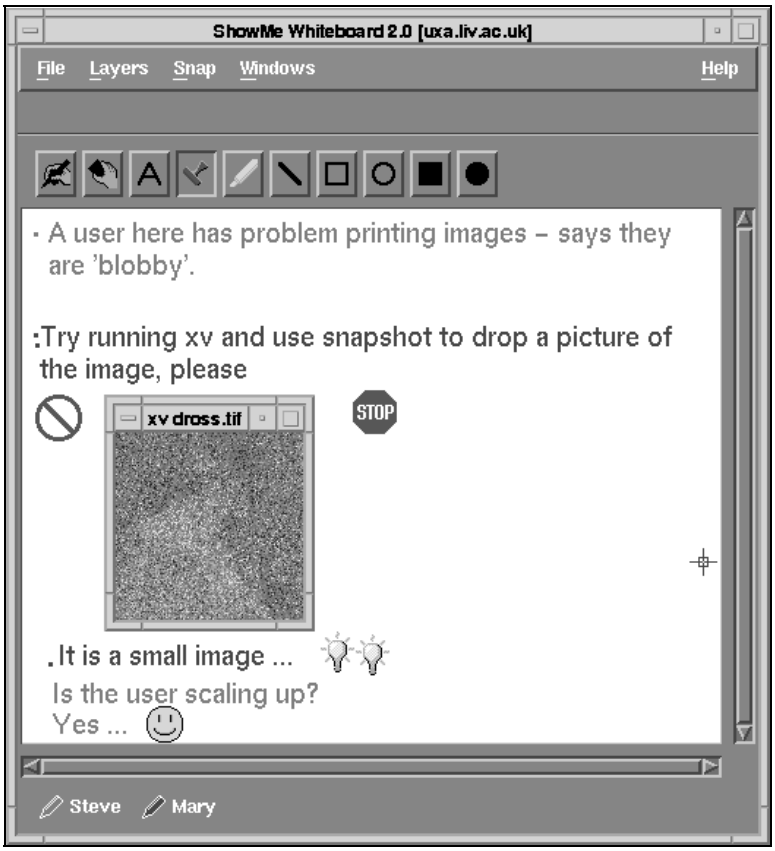


Figure 2.5 ShowMe Whiteboard

Another feature of this tool is the ability to snap regions or windows and make copies for everyone in the conference. The contents of the whiteboard can be saved in a JPEG or SUN Raster file and sent to a printer or to a PostScript file. This could be useful in keeping records of Helpdesk queries.

One of the particularly useful features of ShowMe whiteboard is the 'stamp' facility. The set of stamps included signals, "STOP", "GO", arrows and smiley faces. We found some of the stamps quite useful for quick communication when audio could not be used.

Figure 2.6 Whiteboard Stamps

ShowMe SharedApp

This feature allows a conference participant to start up any application available on a system so that other participants can follow progress. The initiator can drive the application whilst the other participants watch on their own screens. There is a cursor control mechanism which allows other participants to take control of the application and, if desired, to keep control. Participants have different coloured cursors.

In ShowMe, to start up the shared application tool, another conference has to be initiated via a sub-menu of the conference manager. The initiator has to go through the complete process of specifying who should be in the conference and then calling the separate conference. This means that the screen quickly becomes cluttered with the extra windows this process creates. In addition, the address book facility has not been integrated into this part and so a separate list of addresses has to be constructed and maintained.

Applications are chosen from a list of applications which would usually be set up by the system manager. It is possible for users to have their own private lists and also for them to type in a command to initiate an application which is not in the list. Adding extra applications to the list is fairly straightforward.

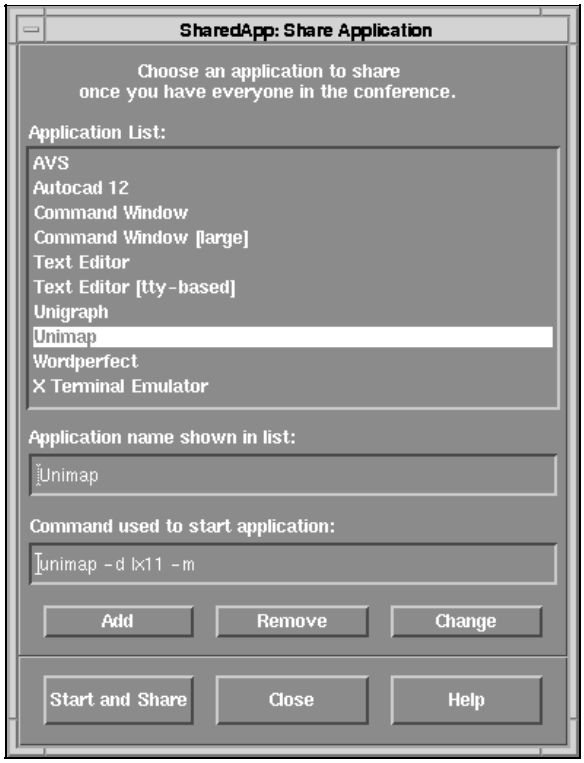


Figure 2.7 ShowMe Shared Application List

Improving Performance

Included in this section are some hints from the ShowMe manual on improving the performance of the ShowMe tools and how to reduce the demand on the network. This will give a flavour of what is involved in using the software in practice and will serve to underpin some of the comments made in Chapter 5.

Video

Here are some suggestions that can be enacted by using the Video controls menu.

- engage picture in picture mode. This means viewing ones own image as a small image in the window showing the video of another participant and it eliminates a window
- hide ones own image and reduce the load on local cpu

- close other video windows if not required - other participants can still see any video being transmitted if still part of the conference but the demand on one's own workstation is lessened
- press the video mute button (camera with red cross) to mute outgoing video - this reduces load on the network but the other participants should be informed first
- use greyscale instead of colours to reduce colour flash. Also reduce number of colours used in the display - this helps to prevent colour flash.

On the Performance Settings menu, there are options to

- limit motion speed to a few frames per second to reduce demand on network
- decrease the picture resolution to low or medium - again to reduce demand on network

Audio

ShowMe Audio can result in 64Kb/sec for each audio stream in a conference. If the network traffic is high, this means the sound output could be choppy. Another problem is echo - to reduce this, the following suggestions should be tried:

- make sure the microphone and speaker are well separated
- turn down the volume on microphone and speaker
- use half-duplex mode which is only to be used when echo is very bad or if there are more than 2 or 3 participants. Protocols have to be set up to signal each time one participant has finished speaking and who is to speak next.
- when not speaking, use the mute button but tell the other participants of your intention beforehand

The last three of these suggestions can be implemented from the Audio Controls menu.

Problems with Shared Applications

The following problems can arise when using the Shared Applications tool:

- there may be flicker or poor performance. This is caused by people moving their cursors which makes some applications refresh and redisplay their windows. This problem can be solved easily by either requesting people who are not currently in control of the application to move their cursors outside the shared application window or switching off cursors
- fonts may look strange - some X11 servers produce variable-spaced fonts with different sizes so the character placement is not correct in some windows.
- there may be a lot of colour flash - this is because the SharedApp tool tries to allocate applications colours on all the machines it is running on. If it is unable to allocate colours on any of these machines, it devises its own map. If the number of colours in the colour table for each workstation is reduced, this problem can be solved.
- some applications don't appear to work - some applications have to be started in a shelltool if the application requires shell input or output or if it is tty-based and not X-based. A shelltool is one of the entries in the list of applications that can be shared.

Some of the above seem to indicate that the current X-windows environment is not designed to cope with the demands imposed by application-sharing

Chapter 3

Other Videoconferencing Software

Introduction

In this chapter, we look in somewhat less detail at some of the public domain software for Unix systems which is available for video conferencing. We also look at Silicon Graphics' InPerson suite for which we were able to obtain the full documentation. During the lifetime of the project we were also able to assess Intel's ProShare videoconferencing package for the PC/Windows platform. Although this was not strictly part of the project remit we felt that a comparison with it would provide useful information.

We installed the public domain software on our SUN system and so our comments on this are based on experience of running it. However, it should be stressed that our information on InPerson was obtained totally from the manual and although we did have two copies of Proshare installed on two PCs we did not run it over ISDN for which it was intended. We did obtain a beta release copy for running over TCP/IP but this was not compatible with our Winsock software and so we were unable to evaluate it fully. Further details are given in the Proshare section.

IVS

Introduction to IVS Software

The INRIA VIDEOCONFERENCING SYSTEM (IVS) suite of software is from the RODEO project in Sophia Antipolis, France and has been installed on our SUN system under Solaris 2.3. It is public domain software and it is available to run on a number of different platforms. See the tables in Chapter 4 for more details. IVS allows users on Unix workstations to communicate via live video and live audio and to have multipoint conferences. It requires a workstation with a screen with 1, 4, 8 or 24 bits depth. It is possible to invoke IVS through the LBL Session Directory which is described in the next section.

On our system, no special audio hardware is required apart from the microphone attachments to transmit speech and the built in hardware in the SUN workstations to decode and encode sound signals.

On video input, frames are "grabbed" by a variety of possible boards, including the SunVideo board.

Documentation

As is often the case with public domain software, the documentation is not of a high standard and initially rather off-putting. It is highly technical in some sections and comes online with the package in the form of some man pages and a couple of ReadMe files about the installation. [In the last month or so, a new version of IVS has appeared and there is now a User Guide which should provide more help.] For comments about the documentation for the whiteboard tool, please read the next section on the LBL software.

The following quote is taken from the IVS documentation:

"IVS allows users to conduct multi-host audio and video conferences over the Internet."

Standards

IVS has tried to implement standard Internet technology to transmit video/audio data. The audio stream can be encoded with either PCM or ADPCM or VADPCM compression techniques. Video encoding is done according to the H.261 standard (see Appendix 3). The image size in the encoded video stream can be either Super CIF format (704 x 576 pixels) or CIF (352 x 288) or QCIF (176 x 144). The default size chosen by IVS is CIF.

Features

The IVS software consists of

ivs	X11 video and audio tool
ivsd	daemon server for IVS
ivs_record	write to a file the audio and video data from participants
ivs_replay	play back a recorded IVS session or play an IVS clip.

The new version, announced about February 1995, has another tool, called ivs_gw, which we did not try out. It has support for RTP (Real time Transport Protocol) which is a developing Internet standard for video and audio communications.

There is no shared application tool and no whiteboard tool included in the IVS suite of software. However, wb, which is also public domain software, can be used as a whiteboard tool. See the next section on LBL for details of obtaining wb. Note that we have not gone into detail about the video and audio components of IVS since the tables in Chapter 4 can be used for a comparision of features.

All these tools are obtainable as public domain software. The software can be transferred using the following commands on a unix system.

```
mkdir ~/ivs
cd ~/ivs
ftp
open zenon.inria.fr

.... give anonymous as your username
.... give your email address as your password

cd rodeo/ivs/last_version
ls
binary
get (whichever binary suits your machine type/system)
quit
```

You will need to uncompress the software and to separate out the various components.

Calling a Conference

There are several ways of calling a conference in IVS. One is to invoke the ivs program itself - this assumes a multicast conference is to be called and tries to find other participants

within the TTL (’time to live’) range that has been set. See Appendix 4 for details of the TTL range. By default, IVS sets this to have the value of 16 which means only the users on workstations connected to the local network and who are also running ivs can be part of the conference. Another way of starting a conference is to supply the address of the other participant on the ivs command line. The third method to start a conference, and one which we used, is to invoke the ivs daemon, the icon of which appears below.

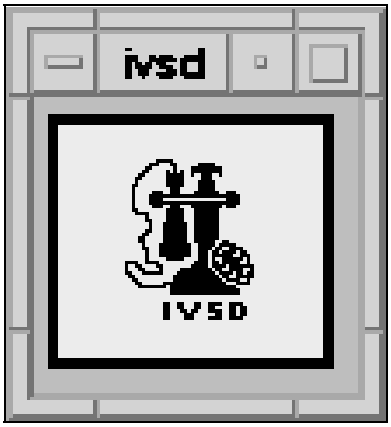


Figure 3.1 Icon of the IVS Daemon

On clicking on this icon, one is invited to opt for a unicast or multicast conference. If a unicast conference is called, one has to supply the address of the other party and then the menu shown in Figure 3.2 appears.

First, it has to be decided whether the video or audio or both tools are to be used and then, the conference is called. If the other parties do not respond, there is an Abort call button. A window with messages will notify the other participant of the request to join a conference if he/she is running an ivs daemon. It is possible to either accept the invitation or refuse to join the conference.

IVS displays the list of active audio/video conference sites with the local site appearing first in color. Sites which are sending audio/video are displayed first, and passive sites below.

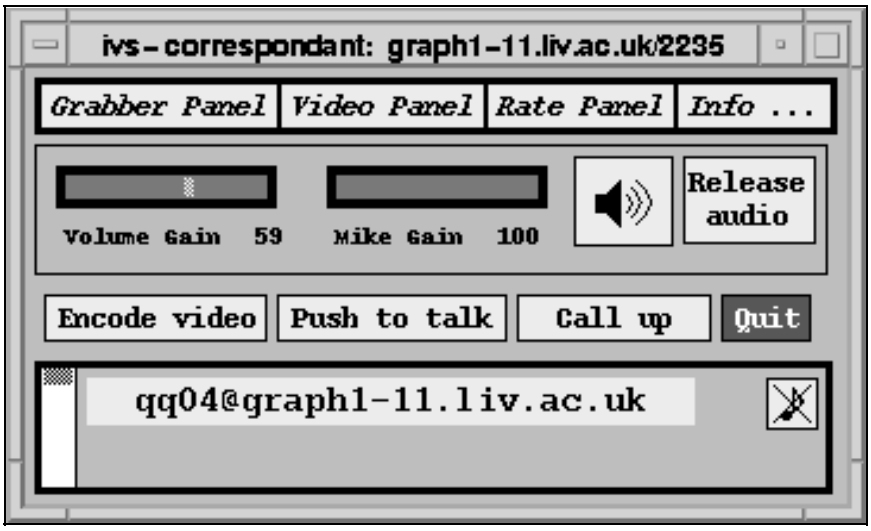


Figure 3.2 The IVS Main Menu

Improving Performance

Included in this section are some hints on improving the performance of the IVS tools and how to reduce the demand on the network.

There are options in IVS to control the output frame rate and the maximum bandwidth used. It is possible also to choose whether to opt for picture quality at the expense of the frame rate or vice versa, to fit both within the constraints of the set maximum bandwidth.

There are the usual audio volume controls plus an easy way to mute conference participants which helps to reduce echo. Undesirable background noise which may be coming from workstation 'hum' or from distant external noises can be suppressed by setting a 'squelch' level (or a level for silence detection). Obviously, care has to be taken not to set this level too high!

All these different options can be adjusted by selecting items on the menu when running IVS or by setting command line parameters. The more technically oriented reader may be interested to know that these options can be set either for the individual user or system-wide by specifying the X Resources to be used and storing these in an Xdefaults file.

LBL

Introduction to the LBL Software

The LBL suite of videoconferencing software originated at Lawrence Berkeley Laboratory, University of California. It is public domain software. For details on how to obtain the programs, see the 'Features' section below.

We installed the suite on our SUN Solaris workstations and it took advantage of the SunVideo board for video capture. There are a number of different versions of the software corresponding to different platforms (see the tables in Chapter 4 for details) and the video tool allows several different compression algorithms and different frame grabbers to be used.

On our system, no special audio hardware other than a microphone is required since the sound I/O takes advantage of the built-in audio hardware.

Documentation

As with IVS, the documentation is relatively poor. It is highly technical in some sections and comes online with the package in the form of a few man pages, lists of technical changes and some notes about use. There is a User's Guide for the whiteboard tool which looked promising but was short on detail.

The following quotes are taken from the documentation:

"The video tool, nv, allows users to transmit and receive slow frame rate video via UDP/IP across an internet." (man page for nv)

"The audio tool, vat, allows users to conduct host-to-host or multicast teleconferences over an internet." (man page for vat)

"If you simply want a shared 'whiteboard' drawing surface, you don't need to do anything

but install wb" (in the ReadMe file for wb)

Standards

By default, the video tool sets the encoding to the native compression for the platform. This means that it supports CellB compression for a Sun workstation. It has decode support for a number of different video encoders but does not support the H.261 standard yet.

The audio tool supports the PCM, ADPCM, DVI, GSM and LPC encoding schemes.

Features

The LBL software consists of

wb	whiteboard
vat	audio tool
nv	network video tool (this comes from Xerox)
sd	session directory
wbimport	help with importing PostScript files into wb

We have not gone into detail about the video, whiteboard and audio components of LBL since the tables in Chapter 4 can be used for a comparision of features. Note that there is no associated shared application tool available.

The Session Directory allows a file to be created which contains a dynamically updated list of sessions (or conferences). A new session can be created, advertised, edited and deleted.

The whiteboard, audio and video tools, (wb, vat and nv), can be run independently of one another. It is more usual, however, to use the session directory tool, sd, where one or more of these tools can be requested, simultaneously.

All these tools are obtainable as public domain software. The tools, wb, vat and sd, (with

other programs), can be transferred using the following commands on a unix system.

```
mkdir ~/lbl
cd ~/lbl
ftp
open ftp.ee.lbl.gov
.... give anonymous as your username
.... give your email address as your password
cd conferencing/sd
ls
binary
get  (whichever binary corresponds to your machine type)
cd ../vat
ls
binary
get  (whichever binary corresponds to your machine type)
cd ../wb
.... etc
```

nv can be obtained by ftp from parcftp.xerox.com on the directory pub/net-research/nv-3.3beta

Just as with IVS, you will have to uncompress the software and obtain the various components.

Calling a Conference

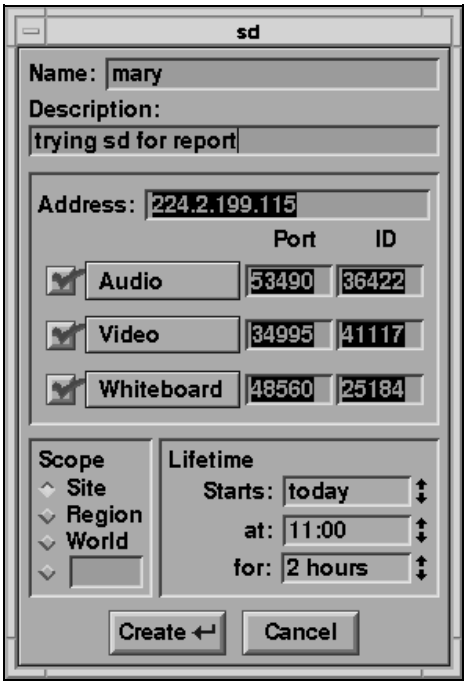


Figure 3.3 Creating a session in sd

To call a conference, the sd program is run and on asking for a new 'session', a window appears as shown in Figure 3.3. This allows the specification of the range and duration of the conference. Once a session has been created, any workstation which has sd running and which falls into the specified range (local site, region or the world) will receive a message advertising the new conference. This is the way to call a multicast conference. It is possible, also, to specify one's own TTL ('time to live') number instead of opting for the local, region or world range. Please see Appendix 4 on Multicasting for more details on the TTL number. At the time stated, the selected tools (audio, video or whiteboard) are started.

If sd is left running, sessions from other sites may be advertised, depending on what range they have specified and a free decision can be made as to whether to join the session.

Another way of calling a conference is to invoke one or more of the tools, nv, vat and wb, specifying the destination and port on the command line for each tool.

Improving Performance

The nv tool has support for different video encoders, including SUN CellB. As usual, it is possible to trade off between quality and frame rate. The nv tool has an options panel or it can be invoked with parameters to specify limits for the bandwidth and frame rate, the size of the video picture and whether the picture should use greyscale or colour.

There is a menu associated with the vat tool to help improve the quality and transmission of the audio - this is the most comprehensive set of features that we have seen and it was a pity we did not have time to experiment. Amongst the features is the ability to suppress echo if no headphones are available, to control the volume gain and to suppress background noise.

All these different options can be adjusted by setting parameters on the vat command line as well as selecting menu items. For the more technically oriented reader, these options can be set either for the individual user or system-wide by specifying the X Resources to be used and storing them in the Xdefaults file.

A notable feature of vat is that of being able to have a private conversation with one of the participants in a multicast conference which other participants cannot hear. This would be useful, for instance, in a helpdesk environment if a consultant needed to consult privately with a colleague about the best advice to give.

INPERSON

Introduction to the Inperson Software

The Inperson software is from Silicon Graphics and runs on the Irix 6 operating system. It has been designed to partner the Indy system, which is part of the Indigo family from SGI, to take advantage of the built-in video and digital audio hardware. The dedicated video chip of the Indy system captures full resolution and full colour frames directly. The integrated audio subsystem provides two audio input channels and four independent output channels.

Inperson supports multicasting. However, the networks need a substantial amount of configuring by the network administrator of the SGI system to support multicast packets. Another job for the administrator is to monitor the possible congestion and to alter the system-wide parameters such as the rate of loss of video transmission or the compression algorithm used for audio.

Inperson is a licensed product - it only runs on the system where a licence has been installed.

Documentation

The documentation comes with the software in the form of two PostScript files; one contains the InPerson User's Guide and the other contains the InPerson Setup and Administration Guide. The User Guide has a reference section and a section containing step-by-step instructions and is easy to follow.

The following quote is taken from the User Guide:

"The Inperson desktop conferencing system lets you have face-to-face conversations and working sessions with people in other locations." (Inperson's User Guide, P1)

Standards

Inperson supports several audio compression algorithms, including the G.711 standard and also the ADPCM and GSM algorithms.

It supports the HDCC compression technique for video.

Features

The main features of Inperson are:

video tool
audio tool
whiteboard
creation of a personal phone book
creation of a group of users to call
shelf tool

Most of these features in Inperson are similar to the corresponding tools in ShowMe described previously. Detailed comparisons can be made from examining the tables in Chapter 4. The 'shelf' in Inperson, which is not an explicit feature in ShowMe, makes it easy to exchange files with other participants in the conference. All that happens is that icons representing files or directories that are to be shared with other people in the conference are put on a 'shelf'; the other participants can then copy the file or directory or just examine the file. However, there are no facilities in Inperson to share applications at present. [We believe there are third party products which do allow application sharing but these, presumably, are not integrated with Inperson.]

Calling a Conference

When Inperson is first invoked, the image of a slightly old-fashioned phone appears on the desktop. This phone rings and flashes if there is an incoming call. To answer the call, the

cursor is moved over the phone and a mouse is clicked. Incidentally, a different ringing sound may be chosen for the incoming and outgoing calls.

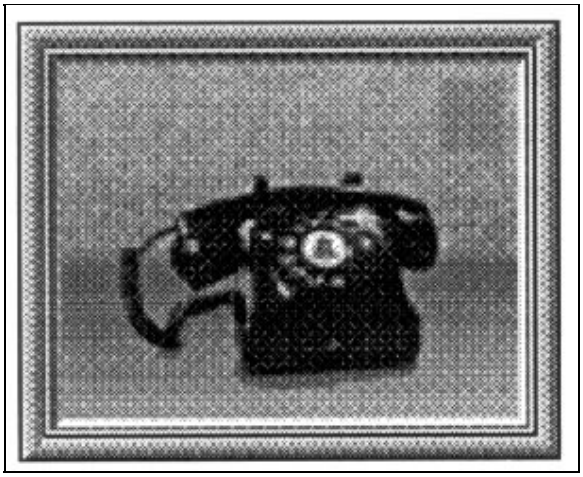


Figure 3.4 The Phone in Inperson

The best way of making calls in Inperson is to have a private or public collection of user icons or group icons. Then a user icon and/or group icon can be dragged on top of the desktop phone to make a call.

Alternatively, when the mouse button is clicked while the cursor is over the phone, a menu appears - this is the calling list. It is possible to type in a field on this menu the names and addresses of people or the names of groups who are to be invited to join the conference. Then the conference may be called.

Like the ShowMe software, Inperson has menus to adjust video and audio parameters and to draw on the whiteboard. See the tables in Chapter 4 for more details on the facilities of Inperson.

Improving Performance

The Inperson User Guide has suggestions about how to find the optimum volume of sound and how to reduce echo and bursts of noises. It gives the usual tips about reducing the size and quality of the video so that the amount of information sent across the network is limited.

It also recommends freezing the video and using static images instead.

To reduce echo, the manual recommends using unidirectional microphones (which detect sound coming from one direction only) if headphones are not available.

Intel's Proshare

Introduction to the ProShare Software

Intel's ProShare package is for videoconferencing between two people using the PC/Windows platform. As noted before, we were interested to see how it compares with the packages running on Unix platforms.

The minimum requirements for running ProShare are:

Intel 486 33MHz microprocessor
8 MB of RAM
Windows display driver running at resolution 640x480 with 256 colours.
DOS 5.0 or higher
Windows 3.1 in enhanced mode
10 Mb of free disk space

The software was reasonably straightforward to install.

The package comes with a video board, an ISDN board, a camera, a headset plus software diskettes and documentation. It should be noted that two expansion slots are needed in the PC for the video and ISDN boards.

The ProShare Video System is designed mainly for use with

- ISDN
- Analog phone circuit
- Netware
- TCP/IP

However, not all features are supported in each environment. It is possible, however, to run the whiteboard alone across TCP/IP. We did not have suitable ISDN connections available

during the lifetime of this project and although we obtained a beta version of the system which did allow connections over a LAN using TCP/IP, unfortunately this version did not support Trumpet Winsock (our local Winsock product) and so we were unable to test it. The comments on Proshare are therefore based on experience of inspecting the local features on a single PC and of using the whiteboard-only features over a LAN. This, we felt, was sufficient for us to assess the main features in order to compare with other software.

Documentation

There are three manuals;

Installation Guide
User's Guide
Quick Guide

ProShare is the only package of the ones we looked at which has an online tutorial available. We found the documentation reasonably easy to follow and use.

The ProShare documentation states:

"The Intel ProShare Personal Conferencing Video System lets you use your computer for a video conference with another ProShare Video System user. You can also share an application with the person you called." (ProShare User's Guide, P.1)

Standards

Proshare is based on Indeo (see Appendix 3) which is Intel's proprietary standard for videoconferencing communications. H.320 support was originally predicted by Intel's brochures for 1st quarter 1995 but it has not appeared yet as far as we know.

Features

The main features of ProShare are:

Video
Audio
Shared Notebook
Shared applications
Dialling List
Address Book

The facilities of the video and audio tools are similar to those in the ShowMe software. See the tables in Chapter 4 for a detailed comparison of the two products.

The shared notebook has similar features to the whiteboards described previously; i.e it allows copying of files into the notebook, text and shapes to be added, snapshots of other windows to be taken, etc.

Strictly speaking, the shared application tool in ProShare is part of the shared notebook. There are two ways of sharing an application. The first one is called Host Edit in which one person starts the Windows (or DOS) application and then asks for it to be shared. That person continues to control the application and the other person can only watch. The other way is called Host and Guest Edit in which one person starts and shares the application and both participants can provide input to the application. This means that the shared application tool in ProShare does not have quite the same functionality as Showme SharedApp. It does not allow the participant who is not the initiator to keep exclusive control of an application (which is an extremely useful requirement in a helpdesk environment). ProShare also allows only two participants whereas ShowMe allows multiple participants.

We feel that ProShare's procedure for starting a shared application is more intuitive than the cumbersome way provided by ShowMe.

A Dialling List can be created as an 'aide-memoire' to make it easy to dial people again. The Dial List contains names and corresponding phone numbers. One good feature of ProShare

is that the names and numbers of people who call are automatically added to the callee's list, unless the callee requests the system not to do this. Another feature is that the last five numbers dialled are up front in the list.

ProShare also has an address book feature - this is probably surplus to our particular needs since the Dialling List is sufficient to call people. However, the address book would be of interest to many departments, since it allows bitmaps (e.g. pictures of people) to be added and documents to be linked to the users' record, which can itself have fields that contain not only names and numbers, but also home addresses, etc.

ProShare's Windows interface makes it attractive to the vast number of people who only know Windows. As one would expect, when running the shared notebook, it is possible to use Windows features such as copying pictures from other Windows applications to the clipboard and then pasting them onto the shared notebook. Another feature is that documents that have been created in an OLE (Object Linking and Embedding) application can be linked to the shared notebook.

Calling a Conference

If ISDN is being used, to call another person, the ISDN number for that user can be just dialled on the 'Handset' (see Figure 3.5).

However, it is recommended in the manual that a Dialling List is created and then all one has to do is to select the correct name and phone number from the list.

When a call is made, the other participant has to be in 'listening' mode before the call can be answered; i.e. on the other PC, there must either be an icon for ProShare Video Listening minimised at the bottom of the screen or ProShare itself must be running.

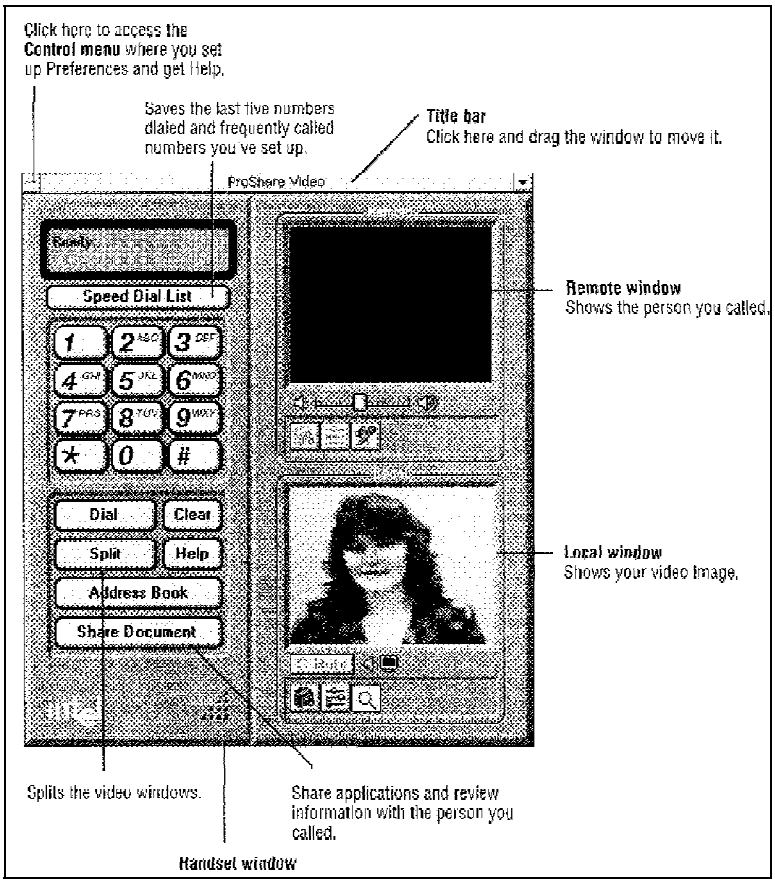


Figure 3.5 The ProShare Main Menu

Improving Performance

There are many hardware suggestions in the ProShare User's Guide for improving the performance, including the addition of more RAM (16MB or more recommended) and a change of processor to one which runs at 50MHz or greater.

Other suggestions are to close all other applications and to use plain Windows wallpaper as the background to reduce colour flash. The performance is also improved if the video windows are reduced in size and if the quality of the picture is changed from high to low.

The ProShare package contains a special headset which consists of a microphone and an earphone; using this helps to reduce the echo. It is possible to connect speakers so that other

people in the room can hear what is being said but of course this increases the chance of echoing.

Chapter 4

Evaluation Tables for Videoconferencing Software

Introduction

In this section, we convey our impressions of various video conferencing products. Evaluation tables are provided so that readers can get a feel for how typical products compare. We have, on the whole, selected features which we thought were relevant to 'desktop' video conferencing, This section will be useful to anyone carrying out a future evaluation of video conferencing products.

The feature tables were classified using the following headers.

Table headers	Summary of features
General	What top level facilities are available, on which machines ....?
Conference Manager	Invitations, selection of tools, nicknames for participants ...
Video	Adjust speed, cope with flash, zoom, snapshot ....
Audio	Adjust volume, duplex switch ....
Shared Whiteboard	Drawing tools, text, stamps ....
Shared Applications	Take control, keep control ....
Address Book	Add cards, sort, navigate ....

In the tables, the first column contains a feature and the second column contains an indication to show how relevant we feel the feature is to video-conferencing in a helpdesk environment.

**	means	Found to be very useful
*	means	Found to be desirable
?*	means	We have not been able to try this but we think it would be very useful
?	means	We have not been able to try this and we cannot assess its importance to us.

The next 5 columns show how well the features are implemented in the different items of software. (SHOWME, IVS, LBL, INPERSON and PRO-SHARE). The grades assigned are

☺	means	well implemented	
⊖	means	partially implemented	
☹	means	not implemented.	
NA	means	Not Applicable	i.e. the product is not designed to support the feature.

We have checked each item from the manuals, from the help pages and from experimenting with the package but, of course, we cannot guarantee that a feature marked as not implemented is really missing!

General

General Table	Imp	SMe	IVS	LBL	INP	PRO
SUN SUNOS	*	⊖	☺	☺	NA	NA
SUNOS Solaris	**	☺	☺	☺	NA	NA
SGI IRIX	*	NA	☺	☺	☺	NA
DEC OSF	?	NA	☺	☺	NA	NA
DEC ULTRIX	?	NA	☺	☺	NA	NA
HP 9000/700 UIX	?	NA	☺	⊖	NA	NA
IBM RS6000 AIX	?	NA	☹	⊖	NA	NA
IBM PC or compatible	**	NA	NA	NA	NA	☺
Technical Support	**	☺	⊖	⊖	☺	☺
Online Help	**	☺	⊖	⊖	☺	☺
Tutorial	?	☹	☹	☹	☹	☺
Documentation	**	☺	⊖	⊖	⊖	☺
Video boards supported	**	2 for SUN	Many	Many	SGI	PC
Standards (video)	**	CellB	H261	CellB	HDC	Indeo
Standards (audio)	**	G.711	PCM	LPC	G.711	Indeo
Conference Manager	**	☺	⊖	☺	☺	⊖
Video	**	☺	☺	☺	☺	☺
Audio	**	☺	☺	☺	☺	☺
Whiteboard	**	☺	☺	☺	☺	⊖
Shared Application	**	☺	☹	☹	⊖	☺
Address Book	**	☺	☹	☹	☺	☺
Multi-casting	?	⊖	☺	☺	☺	☹

Comments: LBL also uses PCM and GSM to transmit audio and supports variety of video compression algorithms.

Conference Manager

Conference Manager	Imp	SMe	IVS	LBL	INP	PRO
Call using nicknames	**	☺	☹	☹	⊖	⊖
Call full name + address	*	☺	☺	☺	☺	☹
Call by selecting a group	*	☺	☹	☹	☺	☹
Call by selecting user icons	?*	☹	☹	☹	☺	☹
Call last number	?*	☹	☹	☹	☹	☺
Call using a public list	**	☺	☹	☹	☺	☺
Give a subject	*	☹	⊖	☺	☺	☹
Call a one-way conference	*	☺	☹	☹	☹	⊖
Join conference in progress	**	⊖	☺	☺	☺	☹
Add selected tools to conf	**	☺	⊖	☺	⊖	⊖
Reject tools selectively	*	☺	☹	☹	⊖	☹
Leave tools selectively	*	☺	☹	☹	☹	⊖
Quit conference unilaterally	*	☺	☺	☺	☺	☹
Flashes at invitation	?*	☹	☹	☹	☺	☹
Audible rings at invitation	**	☺	☹	☹	☺	☺
Invitation Panel	**	☺	☺	☺	☹	☺
Change rings	?*	⊖	☹	☹	☺	☺
Hangup if no reply	**	☺	☺	⊖	☺	☺
Set / change timeout	?	☹	⊖	☺	☺	☹
Own status in conf	**	☺	☺	☺	⊖	☹
Status of others	**	☺	☺	☺	⊖	☹

Video

Video Table	Imp	SMe	IVS	LBL	INP	PRO
Adjust speed manually	**	☺	☺	☺	☺	⊖
Adjust speed automatically	**	☺	☺	☹	☺	☺
Adjust picture resolution	**	☺	☺	⊖	⊖	⊖
Use grayscale	**	☺	☺	☺	☹	☹
Adjust number of colours	**	☺	☹	☺	☹	☹
Adjust colour balance	?	☹	☹	☺	☺	☹
Brightness control	**	☺	☺	☺	⊖	☺
Picture in picture	*	☺	☹	☹	☹	☹
Use static image instead	*	⊖	☹	☹	☺	☹
Hide one’s own image	**	☺	☺	☺	☺	☺
Hide image of others	*	☺	☺	☺	⊖	☺
Zoom image sizes	**	☺	☺	☺	☺	☺
Allow one-way video only	*	☺	☹	☹	☹	☹
Videotape	*	☺	☺	☺	☺	☹
Switch between video ports	*	☺	☹	☺	☺	☹
Mirror image	*	☺	☹	☹	☺	☺
Snapshot of oneself/others	*	☺	☹	☹	⊖	☺
Send snapshot to others	*	☺	☹	☹	⊖	⊖
Save/print snapshot	*	☺	☹	☹	⊖	☺
Actual frame rate info	**	☺	☹	☹	☺	☹
Bandwidth information	**	☺	☺	☺	☺	☹
Status area for participants	**	☺	☺	☺	☹	☹

Audio

Audio Table	Imp	SMe	IVS	LBL	INP	PRO
Adjust outgoing volume	**	☺	☺	☺	☺	☹
Automatic volume control	**	☺	☺	☺	☹	☹
Adjust incoming volume	**	☺	☺	☺	☺	☺
Audio test facilities	?	☹	☹	☺	☹	☺
Suppress background noise	?*	☹	☺	☺	☺	☹
Choose audio encoding	?*	☹	☺	☺	☺	☹
Playout delays	?	☹	☹	☺	☹	☹
Headphones	**	☺	☺	☺	☺	☺
Speakers	**	☺	☺	☺	☺	☺
Full/half duplex switch	*	☺	⊖	☺	☹	☹
Mute oneself	**	☺	☺	☺	⊖	⊖
Mute others	**	⊖	☺	☺	☺	☺
One way audio	*	☺	☹	☺	☹	☹
Side conversations	?*	☹	☹	☺	☹	☹
Record voice	?	☺	☺	☹	☹	⊖
Status messages (comms)	**	☺	☺	☺	☹	☹
Status area for particpants	**	☺	☺	☺	☹	☹
Play audio CD/tape	?	☺	☺	☹	☺	⊖
Audio acquisition	?	☹	☹	☺	☹	☹

Comments: Audio acquisition is to indicate what takes control of the audio hardware when there is a competition for the audio facilities.

Whiteboard (PART1.)

Whiteboard Table	Imp	SMe	IVS	LBL	INP	PRO
One way	*	☺	NA	☹	☹	☹
Status area for participants	**	☺	NA	☺	☺	⊖
Import images	*	☺	NA	☹	☺	☺
Import PostScript file	?*	☹	NA	☺	☹	☹
Save window contents	*	☺	MA	☹	☺	☺
Print window contents	*	☺	NA	☺	☺	☺
Clear window	**	☺	NA	⊖	☺	☺
Layers/Pages	*	⊖	NA	⊖	☺	⊖
Clear one layer/page	*	☺	NA	⊖	⊖	☹
Sort layers/pages	?*	☹	NA	☹	☹	☺
Move items to another page	*	☺	NA	☹	⊖	⊖
Snap window/screen/region	*	☺	NA	☹	☺	⊖
Erase drawing item	**	☺	NA	☺	☺	☺
Undo last operation	?	☹	NA	⊖	☹	☺
Move objects	?	☹	NA	☺	☺	⊖
Select group of objects	?	☹	NA	☹	☺	⊖
Cutting/Pasting	?	☹	NA	⊖	☺	⊖
Distinctive cursors	**	☺	NA	☹	☺	☹
Ask who drew this object?	?	☹	NA	⊖	☺	☹
Ability to point	*	⊖	NA	☹	⊖	☺
Make private notes	?	☹	NA	☺	☹	☺

Comments: The makers of wb (part of LBL) claim that distinctive cursors do not work well, according to some human factors research. A deliberate decision was made to omit them.

Whiteboard (continued)

Whiteboard Table	Imp	SMe	IVS	LBL	INP	PRO
Freehand/line	**	☺	NA	☺	⊖	⊖
Ellipse/Rectangle/Circle	**	☺	NA	☺	⊖	☺
Line thickness	*	☺	NA	☺	☺	☺
Line colour	*	☺	NA	☺	☺	☺
Arrow/arrow colour	?	☹	NA	☺	☺	☹
Filled shapes	*	☺	NA	☹	☹	☺
Patterns	*	☺	NA	☹	☹	☺
Text	**	☺	NA	☺	☺	☺
Text size/colour	*	☺	NA	⊖	☺	☺
Text fonts	?	☹	NA	☺	☺	☺
Edit text	?	☹	NA	☹	☺	☺
Import text	?	☹	NA	☺	☺	☺
Highlighter	*	☺	NA	☹	☹	☺
Highlighter colour	*	☺	NA	☹	☹	☺
Variety of stamps	**	☺	NA	☹	☹	☹

Shared Application

Shared App Table	Imp	SMe	IVS	LBL	INP	PRO
Status area for participants	**	☺	NA	NA	NA	⊖
Legend	**	☺	NA	NA	NA	⊖
Take control	**	☺	NA	NA	NA	⊖
Keep control	**	☺	NA	NA	NA	⊖
Cursor colours	*	☺	NA	NA	NA	☹
Hide other cursors	*	☺	NA	NA	NA	☹
Select application from list	**	☺	NA	NA	NA	☺
Exchange files	?*	⊖	NA	NA	☺	☺
View-only files	?	⊖	NA	NA	☹	☺
Public edit of files	**	☺	NA	NA	☹	☺

Address Book

Address Book Table	Imp	SMe	IVS	LBL	INP	PRO
Create address books	**	☺	NA	NA	☺	☺
Open old book	**	☺	NA	NA	☺	☺
Import address book	*	☺	NA	NA	☺	☺
Create a group of addresses	*	☺	NA	NA	☺	☹
Save address book	**	☺	NA	NA	☺	☺
Print address book	*	☺	NA	NA	☹	☺
Add a card	**	☺	NA	NA	☺	☺
Copy a card	*	☺	NA	NA	⊖	⊖
Delete a card	**	☺	NA	NA	☺	☺
Sort by firstname	*	☺	NA	NA	☹	☺
Sort by lastname	**	☺	NA	NA	☹	☺
Define structure of cards	*	☺	NA	NA	☹	☺
Add fields to card	*	☺	NA	NA	☹	☺
Navigate cards	*	☺	NA	NA	☺	☺

Conclusions

This section is to give a summary of the evaluation tables and will help to give an indication of how many features each package supports.

We start by assigning marks and simply finding the totals for all but one of the previous tables (the General table is not included)

2 marks for ☺
1 mark for ⊖
0 marks for ☹

Features Table	Maximum	SMe	IVS	LBL	INP	PRO
Conference Manager	42	30	17	19	29	16
Video	44	41	22	25	28	19
Audio	38	25	27	34	17	13
Whiteboard	72	46	NA	35	47	50
Shared Application	20	18	NA	NA	NA	12
Address book	28	28	NA	NA	17	25

We then construct another table where the features are weighted according to how important we feel they are in supporting desktop videoconferencing. The marks assigned are again

2 for ☺
1 for ⊖
0 for ☹

but this time having weighting factors:

2 for **

2 for ?*

1 for *

1 for ?

This is the resultant table.

Weighted Table	Maximum	SMe	IVS	LBL	INP	PRO
Conference Manager	68	48	28	30	47	30
Video	66	63	40	42	42	29
Audio	62	42	49	58	32	22
Whiteboard	92	62	NA	48	59	62
Shared Application	34	31	NA	NA	NA	22
Address book	40	40	NA	NA	27	35

These figures should clearly be used with caution. There are various reasons for recommending that the figures are treated as guidelines only. One is that neither of these final tables (with the emphasis on the **number** of features supported) has any indication about the actual **quality** of the video or audio received by a conference participant. However, it is reasonable to assume that an increase in the number of performance options, colour controls, or features to ascertain the performance for both video and audio, leads to more flexibility in obtaining higher quality video and audio. So the figures should not be dismissed out of hand!

One could conclude from the tables that ShowMe is the best package but it has to be pointed out that our experiences of desktop videoconferencing were mostly in using ShowMe. This meant that the choice of features for the tables was made with ShowMe in mind.

If the two tables are examined side by side, we can deduce that in video and audio, both IVS and LBL have a high proportion of 'weighted' features. So although they may not have as many features, they do score well on the important ones.

Therefore, the public domain products, since they are free, must be taken seriously. IVS is a good audio/video tool. It has the advantage over LBL that it supports the H.261 standard. Its main disadvantages, relative to proprietary products, are that it is comparatively poorly documented, it has a relatively technical user interface (compared, for example, with ShowMe) and is not integrated with the whiteboard tool, wb. The LBL set of tools, nv, vat, wb and sd, are better integrated but otherwise suffer from the same disadvantages as IVS.

We feel that the public domain products are more suited to IT specialists at present and would not be suitable for a general service environment. Also, we are not aware of any public domain offerings that support shared applications.

In a more extensive evaluation than the one presented here, one would expect to see a classification of features according to their importance to various types of videoconferencing. Also, there would be performance and quality tests, particularly for the video and audio components.

Page intentionally left blank

Chapter 5

Helpdesk/Advisory Case Studies

In this Chapter we report on the experiences we gained with the Showme software in the Helpdesk/Advisory environment. Our aim was to carry out a pilot study to begin with in order to assess the facilities and later to train sufficient staff so that the system could be used in ’live’ advisory sessions. In fact we did not get much beyond the pilot study for various reasons described below but we do feel, however, that we were probably able to gain as much experience as we would have under live conditions by setting up various advisory ’scenarios’ which reflected typical problems that advisors met in practice. These scenarios were acted out by the project team.

Case Studies

In order to carry out a pilot study under reasonably controlled conditions, we identified what might be termed a set of generic query types which from experience we assessed to be typical of the sort of query that is commonly brought to the Helpdesk and which would be likely to benefit from videoconferencing when dealing with remote expert advisors. We tried to come up with scenarios which would involve use of the different components of videoconferencing described earlier. Of course the basic video and audio components are usually present and provide a basic form of communication but there are also cases where the video component can be used in different ways.

We identified the following scenarios which anyone who has manned a Computing Services’ Advisory Desk will no doubt recognise. In fact the scenarios probably have analogues in any advisory situation and are not necessarily only related to computing.

User has "incorrect" plot (use of video/audio only)

This is a common kind of query where the user has a problem which needs to be identified

in some visual way which can be difficult to describe over the telephone. This can be a plot which does not 'look' correct or typically has some part inexplicably missing or blanked. It could equally be output from a wordprocessor where a special font is incorrectly spaced or the formatting is subtly incorrect. In other words any form of output for which it is easier to *show* the expert what is perceived to be wrong.

The videoconferencing facilities in this case can be used in such a way that the user shows the expert via the video link what is wrong and the advisor then uses the audio/video facilities to give advice. It may, of course, be appropriate for the expert to use the whiteboard or run the application which created the problem as a shared application so that the way the problem was created can be ascertained. However in the pilot study we wished to determine whether the basic video/audio capability was useful in its own right.

Problem using part of a package (use of shared application)

This is the sort of problem where a user has used a package in a certain way and has come up with an unexpected or incorrect result or possibly caused a bug in a package to manifest itself. Such queries are very difficult to deal with over the telephone and almost always require face-to-face consultation.

This is an obvious case for use of the shared application facility in which typically, following initial contact using the video/audio component, the user demonstrates the problem by launching the relevant application and re-creates the problem while the expert is watching. The expert then takes control of the application and shows the user the correct way of carrying out the task.

User wants to draw something via Autocad (use of whiteboard and shared application)

This is a common query which again is very difficult to communicate via the telephone. It covers frequently arising cases where the user has some drawing or model in mind which he wishes to create using a particular package. We have given AutoCad as an example application here to fix ideas since the problems associated with it occur quite frequently at

Liverpool and often involve 3D problems which are particularly difficult to describe in words. There is a requirement with this sort of problem to communicate the envisaged model accurately to the expert so that the latter can assess the best way of using the particular package to create the effect. The expert then usually wishes to demonstrate this to the user.

This sort of query can be handled by the user sketching out his problem on a shared whiteboard and the expert also using the board to then confirm/deny his/her impressions of what is required. When the participants are agreed on the requirement the expert can optionally use the shared application component to demonstrate how to achieve the effect with the appropriate application.

Results of the pilot study

The above scenarios all appear at first hand to be eminently suited to solution using the videoconferencing tools described already in this report. In the remainder of this Chapter we report on our experiences of actually trying to use the facilities in live situations to deal with the above scenarios. As we all know, things that look good in theory are often flawed in practice!

In the following we have tried to organise our findings into well-defined areas as much as possible to try to extract separate conclusions. In the study, however, all these different issues were all interacting to give an overall level of performance and ’feel’ for how useful the tools were. We therefore also give some overall impressions at the end of the section.

The pilot study was carried out in two ways. Some experiments were carried out in adjacent offices. This helped greatly at first since we did find it necessary to run back and forth in order to sort out in person various minor problems we encountered with the software. This reflected partly our inexperience with Showme but is also a reminder of the superior efficiency of human contact. Subsequent experiments were carried out between the advisory desk and an office in a remote building. It is interesting to note that this was necessary in order to simulate real conditions since the temptation to pop one’s head around the door when

the experiments were done in adjacent offices was almost irresistible!

Conference Manager & Address Book

The conference manager is the main interface between the user and the various components. Although it seems obvious that the interface should be as easy to use as possible we realised that it was extremely important in the case of videoconferencing because of the need for it to be used by people with little computer literacy. We found the packages we used somewhat laboured in this respect although Proshare's use of the mobile phone paradigm is perhaps along the right tracks.

The address book features, when used in a LAN environment required the workstation address of the person being contacted to be coded into a file and hence would not be of much use in circumstances where users move around from workstation to workstation. This, of course, is the case in establishments where publically accessible workstations and PCs are available.

Video Performance and effects

Video performance, as expected, varied during the study from very acceptable to almost useless. The performance was unpredictable and obviously affected by other traffic on the network. It should be noted however that, for many interactions, live video, although subtly important for establishing initial personal contact, is often then not required. It is often sufficient for participants to see static images and indeed Showme, and other videoconferencing products we have seen, allow the sender to freeze the video frame and transmit it to other participants. This is a useful device for reducing the load on the network, especially where many conferences are taking place simultaneously when the potential load on the network is large. It is an essential facility of any good videoconferencing software package.

Showme also provides good monitoring and control of the parameters of video transmission, for example, the frame rate and number of colours in the picture. Adjusting these parameters can reduce the load on the network whilst maintaining an albeit inferior quality of live video.

If there is a requirement for showing documents to participants (as is the case for our first scenario) then a fixed, vertically-mounted camera with adjustable focus, under which documents could be placed for viewing, would be a very useful asset. Using the standard display-mounted camera is possible. However, such cameras have wide angle lenses and fixed focus, and are therefore of limited use for this purpose. If a second camera is to be used then the video board must accept two video inputs. This is in fact a feature of the Sunvideo board and Showme does have software controls for switching between the two inputs. For our study we did not actually set this up but it would be a fairly high-priority requirement for any Helpdesk set-up.

It should be noted that video boards associated with PC/Windows products that we have seen do not cater for two video input channels. It may be possible to overcome this limitation by some sort of hardware switch between two cameras and a workstation or by mounting the camera on a pedestal with a universal joint which would allow the camera to be easily redirected from the conferee to a fixed document.

We found that camera position was important both in terms of the framing of the person being viewed and in terms of the lighting. Some adjustments were necessary in order to cope with different times of day and lighting conditions. Lighting can vary from bright sunlight to darkness over a working day depending on both geographical orientations of different offices and weather conditions. This underlined the fact that a good system would provide flexible controls to control camera angles and contrast and brightness of video pictures. The Showme software did provide such flexibility.

Audio Performance and effects

Audio performance, although variable, was generally worse than expected during the study. It was useable but tended to be ’choppy’ and occasionally failed completely for short periods. It was possible to hear the other participant and carry on a reasonable dialogue in a quiet office but this would be impossible in a busy Helpdesk environment. It was difficult to authoritatively pinpoint the exact cause of the poor sound since it was obviously affected by the quality of the microphones and speakers used. The speakers on the SUN IPX workstations

are known to be of poor quality but we also used external speakers on the Sparcstation ZX and these although improving the sound significantly did not always give a truly adequate performance. The main factor was unlikely to be anything other than variable network performance.

The difficulties with the audio performance mean that some sort of dedicated audio link (telephone with hands-free operation or intercom) would be highly desirable for any live set-up if a LAN was being used as the base carrier for conferencing. This would obviously add to the cost of the overall set-up and is somewhat unsatisfactory since synchronisation of the video and audio streams is then not possible.

The above problems emphasise one of the advantages of using ISDN as the base carrier. With ISDN video and audio traffic can be sent on separate 64 Kbit channels which are dedicated to the conference participants. Although the video performance with ISDN is relatively poor (compared with a LAN under favourable conditions), the sound quality is muchbetter than that obtained with ordinary analogue telephone lines.

Effectiveness of whiteboard

In general we found the whiteboard component to be very useful and we frequently found ourselves using it to compensate for the poor audio quality. The tools provided were easy to use and allowed quite a good level of interaction to occur. The performance seemed to be largely unaffected by other LAN traffic. This is to be expected since changes are relatively infrequent and the amount of data being transmitted, when changes are made to the whiteboard, is small relative to video and audio.

We found that we soon developed short-cuts to communication using the various standard icons provided. The drawing tools were fairly primitive but allowed simple sketches to be drawn quickly. Of course if more complex or, say, accurate 3D models need to be shown then conference participants can cut and paste from a running application's window. For more complex interactions it is possible to launch a shared application such as, say, AutoCAD. In practice, however, we often found that this was impractical due to the excessive time it took

to launch a shared application. This, in some ways, is a reflection on the current performance of our system, but it also raises some user interface issues which are discussed later.

Effectiveness of shared applications

Since a high proportion of advisory queries require customers to demonstrate their problems to the advisor, the shared applications component is a crucial element in dealing with advisory queries from remote locations. To be of real use, however, the facility must be easy to use and it must be consistent across all applications. The Showme product component suffered from being not fully integrated into the Showme suite. In fact a separate conference had to be started in order to launch an application and this, coupled with the fact that a separate address book had to be set up for it, made it cumbersome to use.

We also found that applications were sometimes slow to start up and occasionally would not start at all. Such problems can obviously ruin a conference with a remote person. It is at times like these that the audio and video components can be very important since they give the conference participants good feedback on what is happening and, possibly, reassurance that something is being done about any problems that have arisen.

We experienced flickering and poor performance when running certain applications (AutoCAD in particular). This occurred on our SUN IPX workstations as the graphics cursor was moved over the screen. We also experienced colour flashing which is caused by re-allocation of colour table entries by the Shared Application component. These are documented problems (in the Showme manual) associated with the windowing system (in our case X11) and its use of colours from the colour palette. They seem to indicate the inadequacy of the X-Windows environment to cope with this form of working. Such problems obviously greatly reduce the effectiveness of videoconferencing and, in our opinion, mean that application sharing will not be for the masses until the problems are sorted out.

The Shared Application component is arguably the potentially most powerful feature of videoconferencing suites but it is essential that its implementation presents a smooth, seamless integration with the other components of the software if it is to be used by non-specialists.

It should be stressed here that the above comments are based only on experience with our local hardware/software set-up. Such problems will not necessarily occur in other environments. It need hardly be added that the above emphasises how important it is to evaluate any potential products for usability in the *exact* environment they are going to be used before going too far towards setting up a live system.

Training & Documentation

We found the Showme software quite easy to use in general but there was a definite requirement in the project to provide training for anyone who was going to use the software and especially to those who do not routinely use software in multi-windowed environments. In a GUI environment (X or MS/Windows) basic window management skills are needed. In fact we found, as will be explained below, that screen management i.e. the placement, resizing, iconising and restoring of windows, was very important for the software we used.

For the Showme software, we felt the need to produce crib sheets which listed procedures required frequently but which were not always obvious from the menu systems. This would be especially important in our environment where advisory experts man the advisory desk possibly once a month or less. The sheets are included in Appendix 1.

Issues for the Hard of Hearing

As has been mentioned, a particular area we were interested in was the usefulness of the videoconferencing tools in possibly enhancing the ability of those with hearing difficulties to carry out remote consultations which were impossible over the telephone. We came to several conclusions which are given below.

The video performance, both in terms of speed and clarity, was far below the level at which lip reading could be used for communication. In our environment video frames were dropped frequently by the network and this, of course, rendered lip-reading impossible. The video performance was limited not only by the network technology but also by the Sunvideo card we were using since even when the network was relatively ’empty’ the video was far below

the quality at which it would be possible to discern lip and facial movements well enough to decipher speech. In fact near broadcast quality video would be required for this. As explained in the previous chapter the Sunvideo card captures and compresses the video stream and each workstation decompresses the video stream totally in software. With today's network bandwidths and the consequent need for high compression ratios for video streams we are a long way away from seeing broadcast quality video pictures in desktop videoconferencing environments. Thus we cannot see that desktop videoconferencing products will be available even in the medium term which would provide sufficient quality of video to support lip-reading.

We did find during the study that, even with relatively poor quality video, we began to use a very primitive sign language (such as a thumbs-up and thumbs-down signs) which allowed us to speed up some communication we would otherwise have done via the whiteboard. Formal sign language however requires rapid hand movements and would thus be beyond the capabilities of present desktop videoconferencing systems.

The problems of establishing initial contact in order to start a conference are obviously increased when the remote expert being contacted cannot use the telephone. Thus it is very important to have a fixed time during which the expert is close to a workstation running the videoconferencing software in listening mode and is available for consultation. The Showme software actually pops up a window alerting a potential participant that a conference is being requested, simultaneously emitting an audible alarm. This would not be much good to a person with hearing difficulties who was not watching the screen since they would be unlikely to hear the alarm. The only partial solution to this that we could think of was to have a large brightly flashing signal on the screen which would at least be immediately noticed if the person concerned glanced at the screen occasionally.

Not having the audio component available exacerbates any problems encountered during a session such as difficulties with the shared application part mentioned above. We found in practice though that use of the whiteboard was an excellent, if somewhat slower, substitute to the telephone. In fact we made some use of the icons provided with the standard whiteboard but felt that the ability to expand the icon pallette with one's own icons would

be a valuable asset in building up protocols and devising shortcuts which could be used by those with hearing difficulties. Showme does not support this capabaility but the InPerson software does.

For those with partial hearing a set of headphones would obviously be a useful addition to the equipment as it would for anyone contemplating using the system in a noisy environment.

Other General Observations

When conditions were favourable on our LAN the facilities we used were very good and would be very useful in supporting Helpdesk/Advisory from remote locations. Training of those who are going to use the software would be essential and good aide memoires would be useful for those using the system infrequently. We tested the system with Computing Services staff operating at both ends so they had time to become familiar with the tools. If the system was intended to be used for general users contacting a manned advisory desk then, in our opinion, the robustness and friendliness of the user interface would have to be significantly improved.

Showme did not use use the mobile phone paradigm used by Proshare (Intel) which is likely to be more familiar than pop-up menu panels to less technically-oriented casual users. The controls offered in Showme for varying the performance factors (video, audio and network) were very useful but we felt that several of them were presented at a fairly technical level and would only be useful for those with technical knowledge. This gave the software a feel of being designed for *experimenting* with videoconferencing in inadequate environments (which, of course, to some extent, is what we were doing!) and again emphasises the immature state of the technology.

With the Showme product a fair mastery of screen management was required in order to ’drive’ the system. Each of the components created one or more windows and the screen soon became very cluttered. Some windows soon became redundant and could be closed, others were useful occasionally and were thus best iconised. Positioning of windows was also a problem and we found that a shared application could easily obscure the shared whiteboard

when the latter was still required. All these problems, although irritating, could be coped with by a fairly skilled X-windows user who was forewarned that such things were likely to occur. They would, however, be very off-putting for a casual or occasional user and may be beyond those with little basic training.

There is little doubt that, even in the sort of LAN environment in which we operated, videoconferencing can be used reasonably effectively by trained staff even at the present time even though the video quality was relatively poor. We felt that the video component had a subtle effect on interactions in that we often *felt* that it was superfluous but somehow found that when it wasn't there the conference was less satisfying. As mentioned previously, the video component was most useful in establishing initial personal contact and was less important subsequently unless something went wrong (e.g. an application failed to start). This supports the notion that current desktop videconferencing is really 'video-supported conferencing'. On the other hand the presence of an additional fixed video camera for showing documents and possibly other objects would always be very useful and so video support in the workstations used would nevertheless be useful.

We found that the audio was definitely not essential and that communication could be carried out quite effectively using text and icons on the whiteboard only, for many types of interaction. This is not to say that audio is not important since it is obviously more convenient than communicating speech by typing on the whiteboard, but this may be a significant observation to those wishing to use low cost videoconferencing over a LAN with public domain software or the less expensive options of commercial products which offer whiteboard (sometimes called notebook or data sharing) only. We would not, however, contemplate using such low-function facilities in a Helpdesk/Advisory environment.

The use of telephone links in LAN environments for the audio component has the disadvantage that several separate calls have to be made to set up a conference. In the case of Showme one would have to first establish the telephone link using the telephone number, then call a video conference using the IP address via the Address Book, and finally call a third conference to establish a Shared Application conference. This is obviously very cumbersome.

Because of the various problems we experienced with the video and audio we did not put the system into live service on the advisory desk. Our experiments showed that the facilities were too unpredictable to justify risking them on our customers at present. However we are optimistic that, if we set up reliable audio links and installed a variable focus camera, as mentioned previously, we should be able to introduce the facilities to Helpdesk staff in the not-too-distant future. We are also optimistic that over the next few years the basic carrier technology will be able to support an adequate level of synchronised audio/video communication which will under-pin the whiteboard and shared application facilities offered by desktop videoconferencing.

Chapter 6

Overall Conclusions

In this chapter we give a summary of the report findings in brief form so that readers can see easily the main conclusions and recommendations that can be drawn from the study.

The project looked at several issues and posed the following questions:-

- Is the performance of current desktop videoconferencing adequate?
- What software can be used on Unix platforms and how do they compare?
- How useful is videoconferencing in supporting Helpdesk/Advisory activities?
- Does videoconferencing bring any benefits for those with hearing difficulties?

The following answers are based on our experiences only and should be read with that in mind.

Is the performance of current desktop videoconferencing over LANs adequate?

The performance of the video component is generally poor at present and will only improve when adequate carriers are available. However, desktop videoconferencing is not totally dependent on video quality and use of shared whiteboard and shared applications is not so badly affected by network speeds. Audio performance is unreliable if a LAN is used and it is therefore probably advisable to allow for use of separate audio links based on the telephone system.

[For intra-site conferencing ISDN is currently an expensive option. Also note that Basic rate ISDN (64 Kbit/sec) will never give very good video performance but it may be more appropriate than TCP/IP based conferencing for inter-site (wide area) communications via the desktop.]

What software can be used on Unix platforms and how do they compare?

All major vendors of Unix workstations have products with varying degrees of support for the different components of videoconferencing. The differences between the products are likely to even out in time and the workstations themselves are likely to support audio and video as standard.

Public domain products are very useable but in our experience they are slower and have somewhat poorer performance and supporting documentation. Integration of the different elements is relatively poor and this has implications for the user interface. The items we looked at (which are the most well-known) do not support application sharing. We are not aware of any public domain packages which provide this facility.

How useful is videoconferencing in supporting Helpdesk/Advisory activities?

All the components have great potential in this area provided the performance and user interface issues are overcome.

At the present time it would not be wise to invest greatly in hardware/software but introduction of limited facilities such as shared whiteboard coupled with convenient telephone connections and a camera with variable focus for remote viewing of documents and/or objects would be useful if remote contact is a problem.

Does videoconferencing bring any benefits for those with hearing difficulties?

Although lip reading and/or sign language via video connection is a long way off, we did find that informal gesturing via video helped with communication. The whiteboard can replace conversation to some extent and has the advantage, over a plain textual communication like the Unix *talk*, that illustrations and, if available, pre-defined icons can be used to abbreviate certain kinds of communication.

Further Work

The following are suggestions for further work which arose from the present project and which we did not have the time or resource to address.

Investigate issues relating to widespread use of ISDN in a HE institution.

Investigate Unix/PC interworking (via H320?)

Keep track of Multicasting technology developments

Carry out a full comparative evaluation of products

Page intentionally left blank

Appendix 1

Training Documents Used in the Project

Basic Guide to the SHOWME Software for the Helpdesk

This is a set of brief notes we wrote for people manning the HelpDesk to supplement their training. Practice sessions were held and there were several copies of the ShowMe manual made available. Obviously, since some of the information in this is a duplication of the material in Chapter 2, we have removed the repeated parts from this appendix and marked them as [material omitted].

Introduction to ShowMe Software

[material omitted]

Using ShowMe

Calling MT or JSM

> You should inform either MT or JSM by phone that you want to start a Showme session. (JSM 3746, MT 3746 or 3722). You could also try typing **talk qq04@graph1-11** if phoning fails.

How to Use ShowMe Video, Audio and Whiteboard

1. Log on at a SUN workstation (preferably) or on uxa using a PC running eXceed. If you want to use a camera attachment or microphone, then make sure you are using a local xterm window on the Solaris workstation.

2. *Copy two address books from the home directory of qq04 to your home directory.*

You need do this step once only.

cp ~qq04/.addresses.ab ~/.
cp ~qq04/.sharedapp ~/.

3. Type

showme

and wait for a minute or two until the Showme window appears.

4. Click on **Show address book cards...** and click on the address you want to call then click **Add to call**. Some people have two entries; pick the one for their PC or the one for their workstation.

5. Click on **Call a conference**. The person's name will appear in the list at the bottom. Eventually, you will realise that the invitation has been accepted.

6. Pick one or more tools (**Start Video**, **Start Audio** or **Start Whiteboard**). When the whiteboard appears, we can start typing messages to each other. You may find it helpful to increase the size of the whiteboard window.

How to Use ShowMe SharedApp

1. Tell MT or JSM first that you want to use the Shared Application tool.

2. Select **Conference** on the main Showme window then click **Start SharedApp...** and a small window appears. This is the **Shared App** window.

3. To start a conference, select in this window **Conference** then **Call....**

4. Click on **Show address list** in the new window and click on the address you want to

call. (NB pick the uxa address if MT or JSM is using a PC). Select **Add to call.** Wait a few seconds ...

5. Select **Call a conference** and wait for the invitation to be accepted. Then close this window.

6. Click **Conference** on the SharedApp window then **Share Application**.

7. Click the name of the command from the list in the Shared App window. If the command is not there, MT or JSM will prompt you on the whiteboard

8. Select **Start and Share**. By this time, there will be a number of windows cluttering the screen. You should minimise all but the whiteboard and the SharedApp window and the window in which the application is running. The application should have started for all of us. You have control first, but control can be passed back and forth by clicking **Take Control** on the **Shared App** window.

How to Close Showme

1. If you have minimised the main Showme window, restore it and click **Conference** on this window. Click **Exit**.

2. If you have used **Shared Application**, then click **Conference** on the SharedApp window and then click **Exit.**

Examples to Try

1. You should try out the video and the whiteboard tools on your own, first. Just start **showme** and pick **Start Whiteboard** and **Start Video**.

2. An AutoCAD user brings a picture done on the laser printer. It has been clipped on one side. Call MT or JSM, start showme and start the whiteboard only. You can use

the whiteboard drawing tools to describe the problem! Try this again, using the video as well.

3. A Unigraph user says he wants to draw two plots on the same page. You would call MT or JSM and start **showme**. For this problem, you would need probably both the whiteboard and the shared application tool. Select **unigraph -d lx11** for the shared application. When ready, MT or JSM would take over and show you using unigraph how to do two plots on one page.

4. A user of WP wants to import a graph into his document. You would need to start at least the whiteboard and the shared application tools.

Further suggestions

Hints on Using the Conference Manager

If you want to have a beep on incoming calls, select **Conference** on the main ShowMe window and then select **Preferences/Beep** .. You may also like to change your conference name - this item is also on the Preferences menu.

Hints on Using ShowMe Video

[material omitted]

Hints on Using ShowMe Audio

[material omitted]

Hints on Using ShowMe Whiteboard

The **Windows/Controls** menu gives a chance of changing pen colours, pen widths, eraser widths and text sizes.

You can also import old images and place them on the whiteboard (use **File/Open**). You can save the contents of the whiteboard in a JPEG or a SUN Raster file and you can print to a printer or to a PostScript file.

Hints on Using ShowMe SharedApp

[material omitted]

Appendix 2

SunVideo and the XIL Programming Library

The following sequence of statements (taken from the Sunvideo documentation) show the steps taken by a simple XIL program to produce the video element of a video conference. It should help to clarify the sequence of events in the capture, compression and display of video images. It also helps to illustrate the connection between the SunVideo card and XIL. Nevertheless, it should be emphasised that the sequence shown here is specific to the XIL-SunVideo environment. Other video cards may have different features; for example, some may support decompression while others may not support compression at all..

open the XIL library
connect to the SunVideo card
select an input port for image capture
collect information about the image
capture the image (using the card)
scale the image
if necessary, download the chosen compression algorithm to the card
compress the image (using the card)
store the compressed images (using the card)
decompress and display at a local workstation
broadcast the compressed images over the network (using the card)
receive compressed images from other workstations
decompress and display

About the SunVideo card

Figure A1.1 depicts a workstation scenario. The workstation possesses a local CPU plus memory together with a SunVideo system which has four subsystems, Video Capture, Compression Engine, Frame Storage and the SBUS-Video Interface.

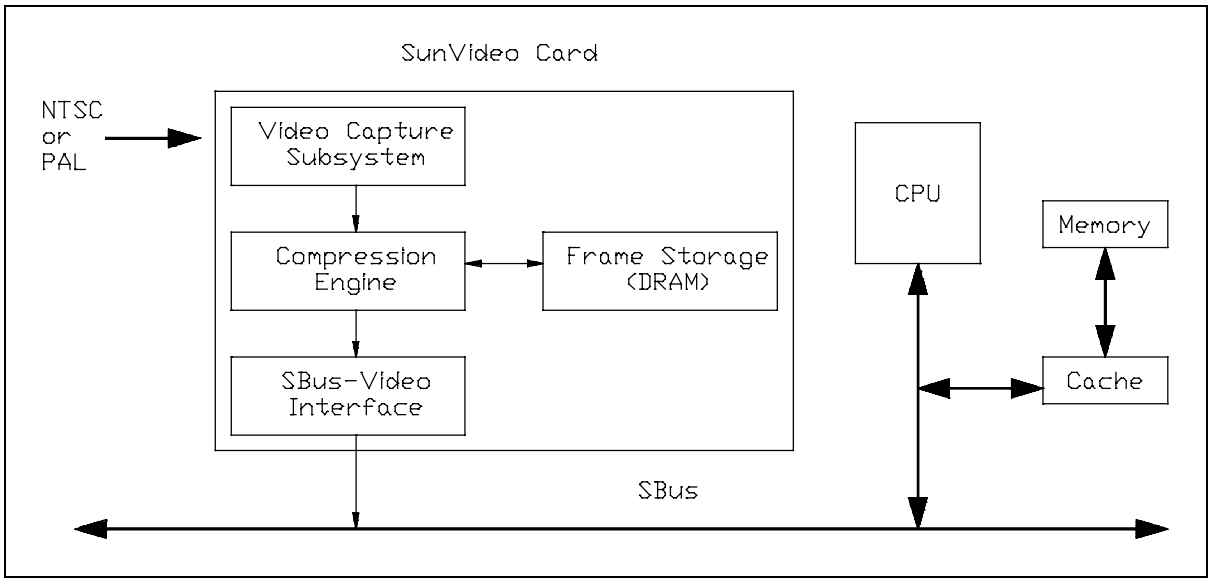


Figure A1.1 A SUN Workstation Scenario

The Video Capture subsystem accepts video in either NTSC (National Television System Committee) or PAL (Phase Alternation Line) formats from video cameras, VCRs and videodisks. It performs the video signal processing and digitizes the analogue input and then passes the digital data to the compression engine. The performance of the capture subsystem depends on the size of the image and whether it is an 8bit or 24bit colour image.

The Compression Engine is based on a CL4000 chip. Currently, this chip compresses the digital data, using either the CellB or JPEG or MPEG technique but it should be possible to use other compression algorithms since the required compression algorithm is downloaded to the compression engine by the XIL application.

The Frame Storage buffer is used by the CL4000 compression engine to store the immediate results of the compression operations. XIL programs can control the compression performance by changing the size of the buffer used.

The compressed video data is then transferred to the network via the SBus Video Interface which connects the SunVideo card and the SBus of the workstation.

For more details on Sun Video, there is a manual 'SunVideo 1.0 User's Guide'

Appendix 3

Compression Methods

In this appendix we give details of compression methods supported by the Sunvideo card and also those used by the different software packages we used in the study. The standards described all fall into the category of *lossy* compression methods which means that the compression method used discards information during the compression process which cannot then be recovered on decompression. Lossy methods allow very high compression ratios (up to 200:1) but can introduce artifacts into the decompressed frames. We also describe here well-known standards (JPEG and MPEG) that are not presently suitable for videoconferencing applications since they require too much computing resource. They are included since they may be useful for comparison purposes.

JPEG (Joint Photographic Experts Group)

This is a standard designed mainly for still image compression. Although JPEG defines a non-lossy method, of main interest here is its lossy algorithm. For video sequences each frame is compressed by transforming 8x8 or 16x16 blocks of pixels via a DCT (discrete cosine transform) to produce coefficients describing the spatial complexity of the image. The method retains the larger features in the image and discards the finer details. Standard Huffman encoding is used to compress the resulting code. The algorithm is symmetrical in that roughly the same amount of time is required to decompress the frames as it takes to compress them. The compression/decompression rates are such that video sequences of JPEG frames can be stored on disk and retrieved at 25-30 frames a second. Provided a sufficiently powerful CPU is available the frames can be decompressed in real time thus giving high quality video playback. This is currently possible only on today's high-end workstations. The compression rates provided by JPEG are not high enough to allow transmission of full video over today's communications lines (T1) or to allow playback from CD-ROM drives.

MPEG (Moving Pictures Experts Group)

There is a series of MPEG standards but we will only deal with MPEG-1 here. Basic frame compression is based on very similar principles to JPEG but in order to reduce the bandwidth

required for transmission MPEG also includes interframe coding where adjacent frames are compared in blocks of 16x16 pixels to determine areas that have changed between frames. This is a very computationally intensive operation. MPEG-1 in fact takes much more time to encode than to decode but decoding rates are still much slower than that required for JPEG. Thus only the most powerful of today's workstations are currently capable of decoding MPEG-1 at full video quality.

CELLB

This is SUN's proprietary codec which is used by the ShowMe software. It is designed for use in videoconferencing applications where real time coding/decoding of the bit stream is required. 4x4 pixel blocks are represented by a 16-bit bitmask and two 8-bit vector quantised codebook indeces. This gives a compression ratio of 12:1. Interframe compression is also used by comparing corresponding blocks in adjacent frames and using skip codes where a block has not changed or is 'close enough' to the previous frame. Quality can be traded against compression rate by modifying the definition of 'close enough'. Video frames are typically compressed at from .4 to .8 bits per pixel which is equivalent to about 10:1 to 20:1.
At these compression rates a FCIF frame (352x288 pixels at 15 frames/sec) would require bandwidth in the region of 0.5 to 1.0 Mbits/sec range which is well within the capacity of today's LANs. CELLB is such that decompression can easily be accomplished in real time by today's workstation CPUs.

H.261

As mentioned in Chapter 1, this is one of the standards included in the ITU-T standard known as H.320. H.261 is a video compression codec which is meant to be utilised on communication lines that have multiples of 64 Kbit capacity (so-called Px64 where P=1...30). H.261, like JPEG and MPEG, is based on DCT but it uses a simpler interframe algorithm than MPEG which reduces the time taken to encode and decode the video stream. The simpler algorithm, however, means that video quality is reduced. H.261 is defined for both the FCIF and QCIF standards. It is difficult to transmit full motion CIF frames with less than 384 Kbit/sec (6 x 64 Kbit/sec) communications so QCIF tends to be used on slower connections such as when Basic Rate ISDN is used. Even then, since compression is reduced when there is a lot of movement between video frames, the pictures will be of relatively poor quality.

Indeo

This is Intel's proprietary codec. We have not been able to obtain details of its algorithms in time to be included in this report. If Intel's Proshare software takes hold in the market place it could well become an important *de facto* standard.

Further information on Indeo and Intel's products can be obtained from *http://www.intel.com/IAL/indeo/indeo.html*

Page intentionally left blank

Appendix 4

Multicasting with ShowMe

The following is based mostly on information in the ShowMe manual. It should serve to make clearer the issues surrounding the use of multicasting.

ShowMe provides an option for specifying that a conference should be opened in multicast mode. When this mode is chosen a window appears inviting the user to choose the range over which the conference is to be broadcast. The user has to specify a TTL (Time To Live) number which defines the distance a given network packet travels. The larger the TTL number, the further the data packet travels. The TTL value is decremented each time the data passes through a gateway [the *precise* definition of what constitutes a 'gateway' is not given].

The manual points out that there are no standards defined for TTL numbers but there are conventions which may be followed by sites. Thus, by convention a TTL number of 1 will mean that data is restricted to the local sub-net, TTL numbers of 32, 64, 128, and 256 will resrtict the data to the site, region, continent, and world (unrestricted) respectively.

Once a TTL number has been selected, ShowMe chooses an IP multicast address for the conference (based on the host machine's IP address) and passes that address to other conference members. This allows the other conference members to send conference messages back to the originator. Current multicasting technology is such that all leaf nodes which are part of a multicast tree receive multicast data regardless of whether they have requested it. If a high TTL number is chosen then, even if only two people are in the conference, all multicast routers are required to carry the traffic if they are within the TTL range. This can obviously have a serious impact on a network.

Multicast technology is under constant development and ways of alleviating the potential problems are being addressed by new versions of the software which avoid sending data to areas of the multicast tree which have not requested data and which allow system administrators to restrict the amount of bandwidth being used on the network.

page intentionally left blank

Appendix 5

Videoconferencing Bandwidth Requirements

	24bit@25fps	**8bit@25fps**	**8bit@25fps**	**8bit@5fps**
VGA (640x480)	184 Mbit/s	61 Mbit/s	36 Mbit/s	12 Mbit/s
SCIF (704x576)	240 Mbit/s	80 Mbit/s	48 Mbit/s	16 Mbit/s
CIF (352x288)	60 Mbit/s	20 Mbit/s	12 Mbit/s	4 Mbit/s
QCIF (176x144)	15 Mbit/s	5 Mbit/s	3 Mbit/s	1 Mbit/s

Carriers

ATM	Up to Gigabit/sec range
Fast Ethernrt	100 Mbit/sec
Ethernet	10 Mbit/s
Basic Rate ISDN	64Kbit/sec (x2)
Primary Rate ISDN	64 Kbit/sec (x30) (max. 1920 Kbit/sec)

Compression Methods

JPEG	> 100
MPEG	similar to JPEG
CELLB	10 to 20 times
H261	10 and up times