Human–Computer Interface Recording

D. MORRIS, C. J. THEAKER, R. PHILLIPS AND W. LOVE
Department of Computation, The University of Manchester Institute of Science and Technology, P.O. Box 88, Manchester M60 1QD

In 1984 IBM published a description of the 'Playback' system, which was designed to assist a human factors analyst in the evaluation of the usability of an application system, in particular in the evaluation of word-processing systems. The present paper describes the work involved in producing a similar acquisition and analysis tool, involving physical portability, multi-channel recording, synchronised video and sound recording and a synchronised system for acquisition of analysis notes, leading to greater applicability.

Received February 1988

1. INTRODUCTION

This paper relates to the creation of tools to assist human-factors analysts evaluate computer-based products. The evaluation is done either by replaying the original interaction and making subjective judgements, by using programs to analyse the information, or a combination of both. The motivation for producing the system is similar to that which led to the 'Playback' system which was developed in the IBM Human Factors Laboratory.[1] Playback was designed to assist a human factors analyst in the evaluation of the usability of application systems implemented on terminals connected to a host. An early use of the technique was in the evaluation of word-processing systems. The basic mechanism was to interpose the laboratory computer between the keyboard and the host so that all user keystrokes could be time-stamped and captured in its filestore. This capturing of keystrokes is totally non-invasive to the system under investigation and therefore does not influence its performance or behaviour. Subsequent playback of the captured keystrokes is then possible either at the original speed or an accelerated speed through periods of little interest to the analyst. Analysis of the captured data by software is also possible.

The work described here represents at attempt to produce a similar acquisition and analysis tool with a greater range of application. Its principal characteristics are as follows.

Physical portability, so that it can be taken to any test site.

Simultaneous multi-channel recording. This has several applications, for example, the capture of the multiple inputs to a single-user system, as might be found in CAD applications. Alternatively it might be used to monitor multi-operator systems as found in complex control situations. Also some channels might be used to capture the computer response in addition to the user inputs.

A closely synchronised video- and sound-recording system.

A similarly synchronised system for the incremental build-up of analysis notes relating to a captured work session.

2. THE HCI MONITOR

The HCI Monitor produced at UMIST is a special-purpose workstation which is used to capture a detailed record of the interactions of one or more people with a computer system during a complete work session. This information can then be used to evaluate the usability of particular input/output devices, software interfaces and total systems. Additionally it may be used to investigate some aspects of human behaviour.

The hardware of the system is based on VMEbus with a single-board computer supplying the main processing power, as depicted in Fig. 1. This contains a Motorola MC 68010 microprocessor, although all the software from the operating system upwards is in fact portable. The processor board is interfaced to an 80-Megabyte hard disc and an IBM PC/AT-compatible floppy drive. A second VMEbus board provides eight RS 232 serial input/output channels. The whole system is normally mounted in a five-position card cage measuring only $5 \times 13 \times 17$ inches, hence the requirement for physical portability is met. The remaining 3 card positions are available for interfaces specific to a particular use of the system. Two common examples of these are the interface to the video recorder and the interface to a video camera. Other examples are the line taps described later.

One attraction of the VMEbus organisation is its flexibility. This is particularly relevant in connection with system performance. In some monitoring applications very high data rates have to be sustained, thus the amount of RAM in the basic system and the bandwidth of the channel to the disc may be inadequate. Extensions to the basic system are therefore provided in the form of extra VMEbus boards. These may require a larger card cage, and at the price of some inconvenience in transportation, an alternative 21-slot card cage is sometimes used. Extensions to the RAM amount simply to the addition of extra memory boards. A more significant extension is the addition of a second processor board and the division of the input/output load between the two processors.

The overall system is shown in schematic form in Fig. 2. To evaluate the usability of a product the Monitor is first used to capture a detailed record of the interactions between a user and a workstation or terminal of a computer system, for a complete work session. For each interaction recorded the system also notes the time of its occurrence (i.e. its time-stamp). For example, on a graphics workstation the inputs from both the keyboard and mouse would be recorded and time-stamped. In addition to this capture of digital data, the system would normally record the user's behaviour with a video- and stereo sound-recording system in which the video frames are also time-stamped. In order for human factors

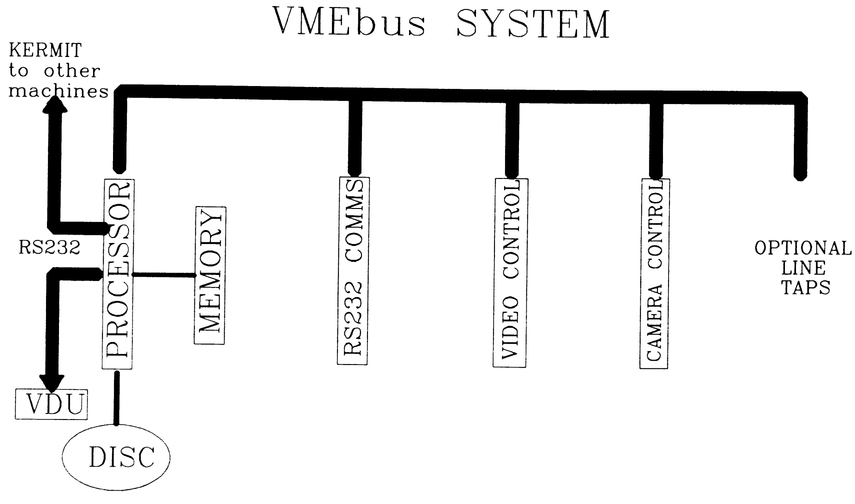

Figure 1. The Monitor hardware.

specialists to analyse usability, a captured work session can later be replayed on the test system where it was recorded along with a synchronised replay of the video.

A key feature of the system is that during replay the human factors specialist can view the user terminal screen with the same fidelity that the original user saw. A video recording alone does not provide this. The video system is mainly used to record the user's behaviour. Another key feature provides for the derivation of a journal of analyst notes for each work session recorded. This journal is normally created during the initial capture of a session, by the observer adding notes regarding the user's behaviour. It may be updated during subsequent replays of the session, building up incrementally a full analysis. The notes in the journal also relate to the time-stamps, to allow their presentation to be synchronised with the replay of a session. They can be printed to form the basis of a usability report.

Because the replay of a session involves sending the captured user input to the test computer system in order to re-create the original behaviour, facilities for moving quickly through a replayed session are limited. Besides replaying at normal speed, the only other provision is for an accelerated form of replay where the thinking time between user inputs is reduced. In order to expedite analysis, the video recording only can be replayed, so that places of interest can be identified more quickly and marked in the journal. During a full replay, periods of little interest can be replayed at the accelerated rate.

In addition to the subjective analysis, the captured user interactions for a session may be processed by software to produce a statistical analysis of the human–computer interactions. There are some basic statistical facilities local to the system which are designed to assist the preparation of reports, but data for analysis can also be transmitted to other computer systems. Quite often only selected intervals of a session need to be analysed. Such periods can be associated with particular types of events or markers in the journal.

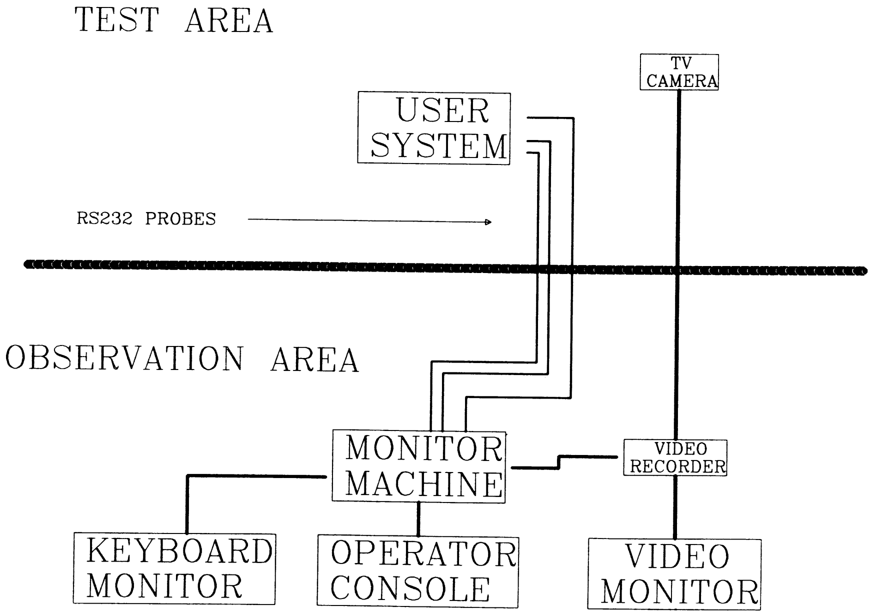

Figure 2. The logical structure of the Monitor.

3. OVERVIEW OF AN EVALUATION

At the outset of an evaluation, the human factors specialist will define the tasks that the users are to carry out on the test system. As some usability evaluations will be of a comparative nature, the subjects will often have been selected by a screening process which will normally involve questionnaires, interviews and tests.

The capture of a subject performing a task is termed a 'session.' The information for a monitored session is maintained by the system in a session database, which consists of the following.

(*a*) A session journal, which contains analyst notes for a session as well as markers to control replay and statistical analysis. Each note and marker is time-stamped and is referred to as a journal entry.

(*b*) A session video tape, which contains a time-stamped video- and sound-recording of a session.

(*c*) A session data file for each captured input or output, which contains the data captured by the Monitor along with timing information for the data.

An example of a system being monitored is given in Fig. 3. In this example the system records the input from both the keyboard and mouse to a user's machine. These connections are very often not RS 232, and the line taps will need to be of a type that convert the signals to RS 232 (see later). In both cases the monitoring system records each character input to the computer together with the time at which it was received. Normally the characters pass straight through the tap to the user machine at capture time, but in some cases it is necessary to process them first in the Monitor system. Section 6 discusses the reasons for his.

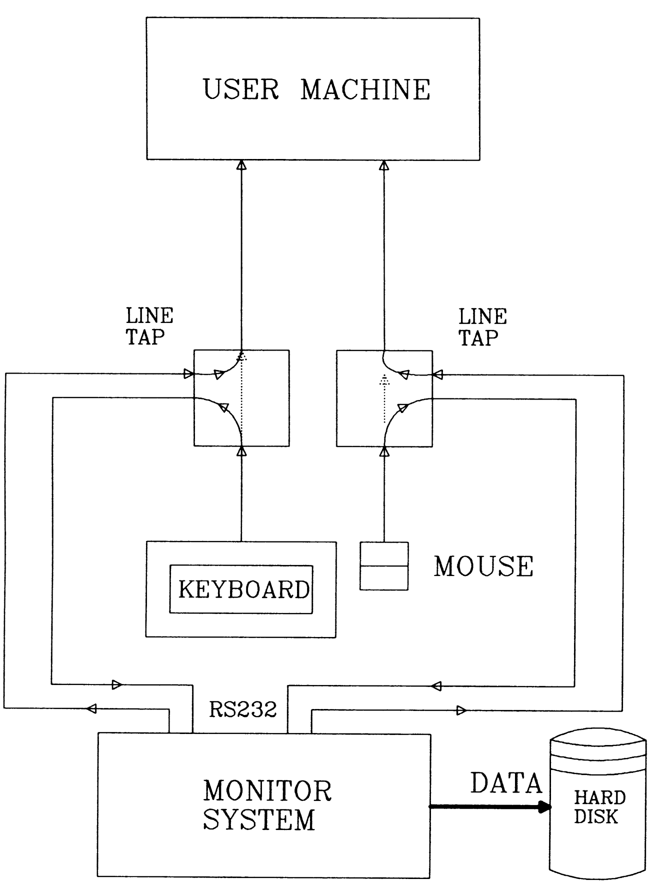

Figure 3. An example of monitoring.

In order to replay a captured work session, the software of the system re-transmits the user input to the computer. Again there may be a need for special hardware to translate from the RS 232 of the Monitor to some other form for the test machine. This is done with the same timing as the original input, in the expectation that the same system responses will be produced. It can repeatedly be replayed and analysis notes and markers added incrementally to the session journal.

A summary of the phases involved in a typical evaluation follows.

(*a*) The human factors specialist defines a set of tasks for the test subjects.
(*b*) The test subjects are selected.
(*c*) The hardware is configured.
(*d*) The system records work sessions of test subjects performing their tasks while the observer creates a journal for each session.
(*e*) The sessions are replayed, observed and the journal of analysts' notes updated.
(*f*) The captured data are processed by software to give a statistical analysis of the interactions.
(*g*) A report on product usability is produced based on session journals and statistical results.

4. LINE TAPS

The general form of a line tap was illustrated in Fig. 3.It is a hardware device which can be placed in series with an input mechanism, such as a keyboard or a mouse, in order to capture the signals which flow to and from it.

The connections from a line tape to the Monitor are via RS 232 lines. If the tap is viewed as a cut through a bi-directional cable, the flow in each direction maps on to the input and output signals on the RS 232 cable. Thus by echoing the input on to the output, the connection can be restored. During replays the input might be ignored and the output will mimic previously captured actions. It may also echo a modification of the input, for example, to apparently slow down the transmission rate of a pointing device.

5. OPERATION OF THE MONITOR SYSTEM

The requirement to replay a work session on the computer system where the session was originally recorded usually imposes some conditions on the initial recording. In some instances, there is the requirement that the environment for the replayed session should be in the same state that it was at the time of the original recording. For example, this might mean that the files the session uses must be in the same state, so that a replayed session produces the same results as the original. This is generally not a problem, since files can be backed up and recovered with ease on most computer systems. Another example concerns the cursor position on some graphics workstations. A replay involving graphics might fail because the cursor positions at the start of replay and capture are different. On single-user workstations, restarting the machine is often a satisfactory remedy. Another requirement for effective replay on the user system is that the responsiveness of the user system should essentially be the same during replay as it was during the original recording. This is less easy to achieve,

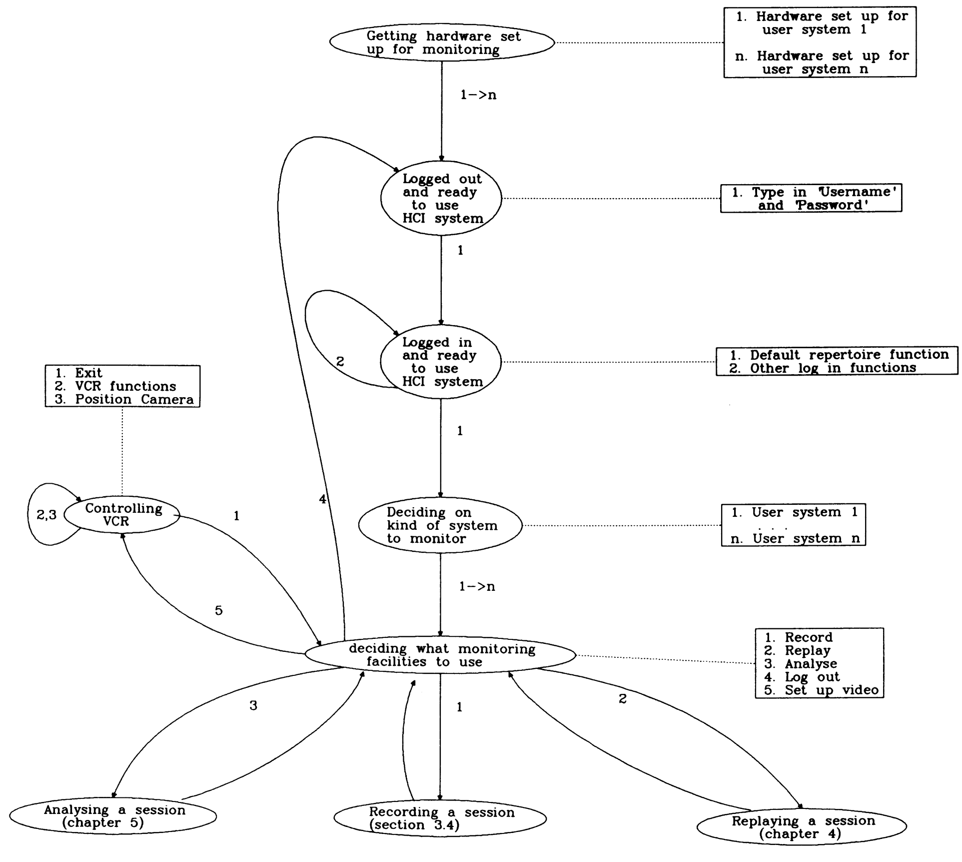

Figure 4. Summary of how to set up the MMI system.

and Section 6 discusses how the Monitor system attempts to compensate for variations in responsiveness.

A menu-driven interface on the console of the Monitor provides control for recording and replaying sessions,[2] and it provides facilities for adding and amending entries in the journal. To record, replay or analyse a work session the observer logs in, selects the kind of user system being monitored and then selects the session function required. The main states and facilities to do this are summarised in Fig. 4. This follows conventions which have been established within the project for expressing the progression through task as a series of state changes. The bubbles represent observer/monitoring states and the labelled arcs indicate the actions which cause each particular change in state. The possible choices at each state are identified by numbers and listed in a rectangular box associated with each bubble. These choices are normally made by the user from a menu.

To capture a work session the user system is first set up, then details about the session are added to the journal and the recording is started. This state is shown in Fig. 5. When the recording is complete the captured data and journal are secured in the filestore of the Monitor. During a recording the observer uses the 'Add marker' function to create markers regarding the behaviour of the test subject in the journal of analysts' notes. When this choice is made the observer is presented with a menu from which to choose the category of the observed event.

If the video-recording facilities are being used, spoken comments can be recorded on one of the stereo microphones. Functions are also provided to stop and start the video recorder during periods of user inactivity. Besides controlling the VCR, the latter functions will add a marker into the journal so that during replay the video can be synchronised automatically with the session replay. The position of the video camera is also controlled by the monitor machine with the 'Reposition Camera' function.

The replay facilitity of the Monitor allows a recorded work session to be replayed repeatedly, so that an observer can build up an initial analysis on usability in a session journal and delimit sections of the recorded session which are to be subjected to further software analysis. The Monitor can present notes from the journal to the observer in synchronisation with the replay, so that the observer can peruse them and if necessary update them. Additionally, the observer can stop the replay whenever he wishes to add a new note to the journal. One way to reduce the cost of observing replaying sessions is to adopt a two-stage approach to the analysis. In the first stage of the replay of a work session the observer might identify periods of user activity which on some subsequent replays might be either ignored, for example, periods where the user is consulting online help facilities; or examined closely to generate comments in the journal, for example, periods where the user makes mistakes. Where human factors skills are scarce this first stage might be carried out by a less skilled observer who identifies the start of these periods by inserting marker entries of the appropriate category into the session journal.

In the second stage the observer would be a human factors specialist, or at least would have some human factors objectives. The replay would be done with the intention of making some initial judgements. During this

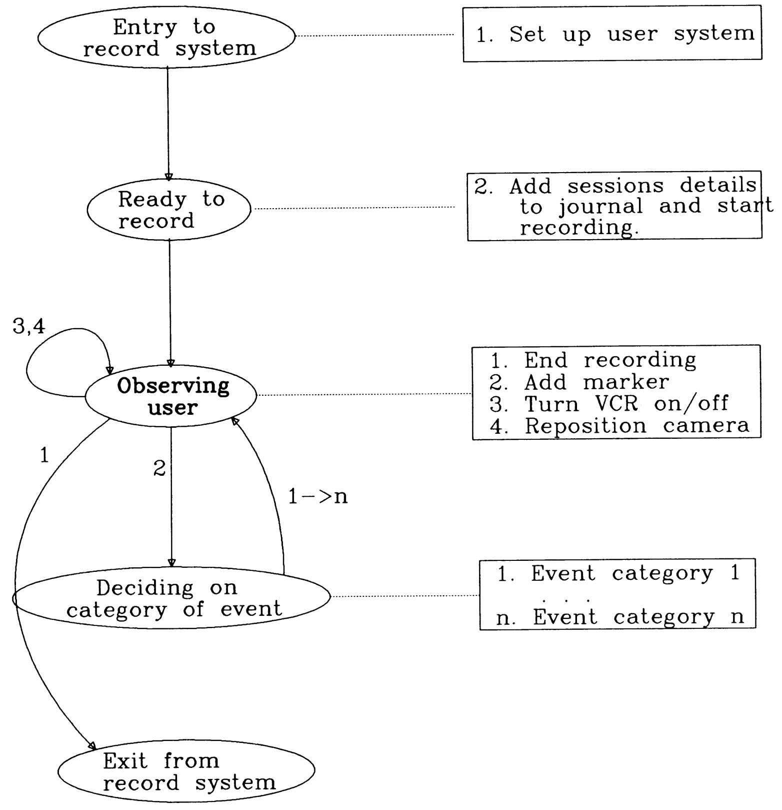

Figure 5. Summary of states and facilities during the capture of a session.

stage, the observer can nominate certain marker categories to control the replay of a session. There are four classes of replay markers, namely SKIP, STOP, PLAY and DISPLAY. The purpose of each of these classes is as follows.

SKIP These markers indicate the start of periods in a session where the observer does not wish to observe the detail. On encountering a SKIP marker the Monitor will continue replay, but at an accelerated rate.

STOP These markers specify points in the session where the replay will automatically stop, to enable an observer to view existing notes in the journal and possibly update them.

PLAY These markers allow an observer to indicate the start of periods which are of particular interest. When a session is being replayed at an accelerated rate and a PLAY marker is encountered, the Monitor will revert to capture speed, thereby ensuring that periods of interest are not passed inadvertently.

DISPLAY These markers allow an observer to set up comments for a session which the Monitor will display during replay. Typically, these comments explain the tasks being performed by the user in the work session.

During replay the observer can switch between the following modes to further reduce analysis costs.

(*a*) Synchronised replay of digital and video recording.
(*b*) Replay of just the video recording.
(*c*) Replay of just the digital recording
(*d*) Replay of just the journal notes.

For example, if a point of special interest is passed while replaying the digital and video recording of a session, the observer can switch to video only to rewind to the precise point of interest. Although the digital replay cannot be reversed, as the test system will already have received the input data, a marker can be added to the journal which will take effect on subsequent replays.

Controls are also provided to allow the observer to change the speed of replay. When replaying a digital recording there are just two speeds, namely NORMAL and FAST. The former reproduces as exactly as possible the capture timing of the original session, while for the latter the monitor machine replays the digital recording of the session with reduced time delays between user inputs. On resuming replay at NORMAL speed the Monitor automatically resynchronises the video and digital replay.

To illustrate the user interface for replaying work sessions, two of the major states that an observer encounters during the replay of a session are presented.

(1) 'Replay stopped'. This occurs when a STOP marker is encountered during replay and also when the observer manually stops the replay.

(2) 'Digital replay in progress.' In this state there may also be a synchronised replay of the video recording.

The functions available to the analyst when the 'replay stopped' state occurs are shown in Fig. 6. From this state the observer can select functions to continue the replay in any of the available replay modes (digital and video, etc.), end the replay, and view or change the settings of replay markers (SKIP, etc.). Additionally, the observer can view and change information concerning the event in the journal at which the replay stopped. There are functions to edit the note of the event, change its category and remove it from the journal.

The functions available during 'digital replay in progress' are shown in Fig. 7. In this state the observer can change the speed or stop the replay. The latter allows

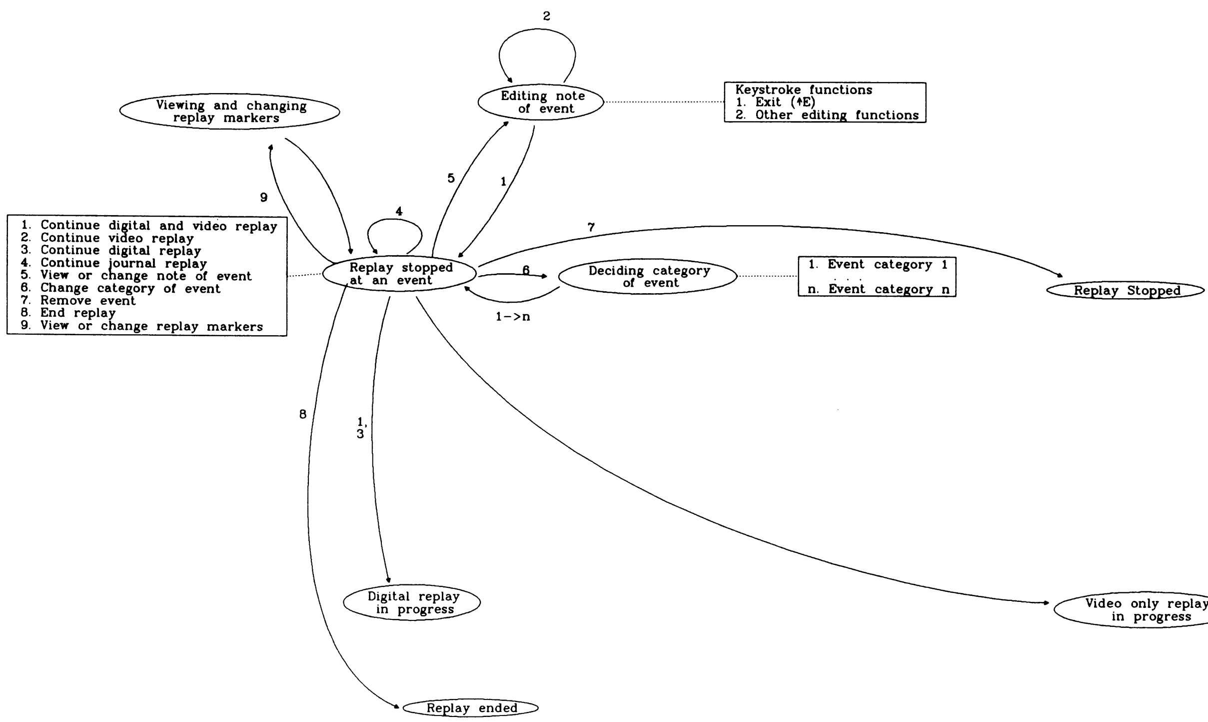

Figure 6. Action of functions when replay has stopped at an event.

closer examination of user displays and the addition of markers and notes into the journal. To expedite the insertion of markers a new marker, with the same category as the last, can be inserted into the journal without stopping the replay.

The user interface diagram in Fig. 8 summarises the states and facilities encountered by the observer during replay.

6. EXPERIENCE

At the outset of this project it was näively assumed that a replay of all inputs to a computer system with the same timing that was observed at capture time would reproduce the same results. In fact it was thought that timing to within one millisecond would be satisfactory. It soon became clear that this is not the case, particularly with high-data-rate inputs such as those from pointing devices. In fact, on multi-user systems whose internal behaviour is a function of system loading and of other features such as type ahead, even simply keyboard input might not produce identical behaviour when replayed.

A solution to this problem was sought through a mechanism which attempted to distort the timing of the replay by taking account of 'prompts' from the test system. This involved capturing and time-stamping a suitable output stream in addition to the user inputs. Output to the terminal is usually satisfactory for this purpose, and synchronisation with this output can be attempted in several ways. Obviously, matching it character by character at replay time can be done if the

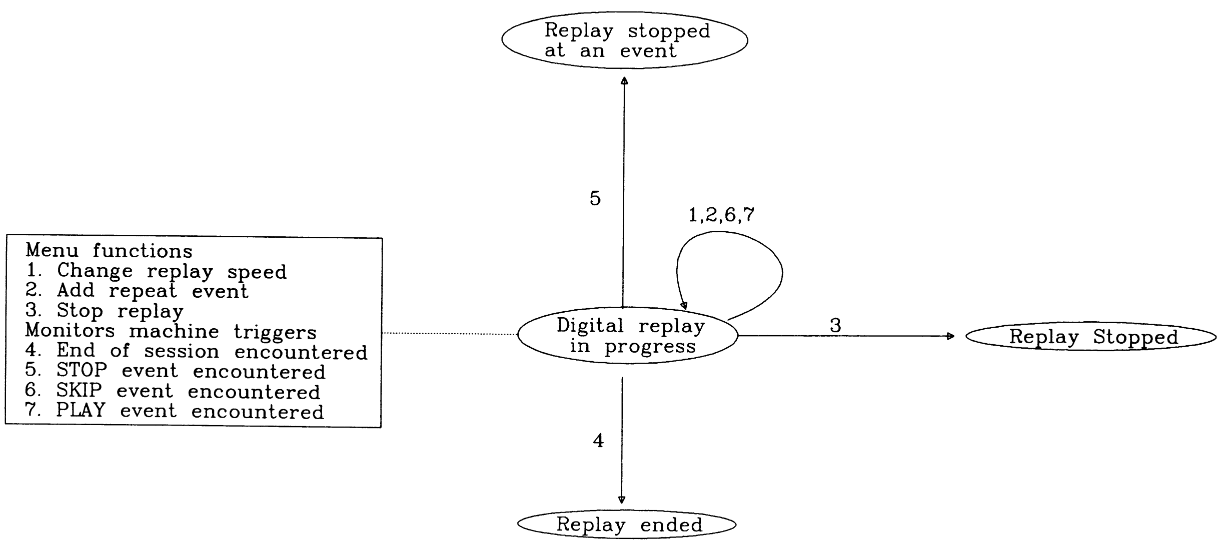

Figure 7. Action of functions when digital replay is in progress.

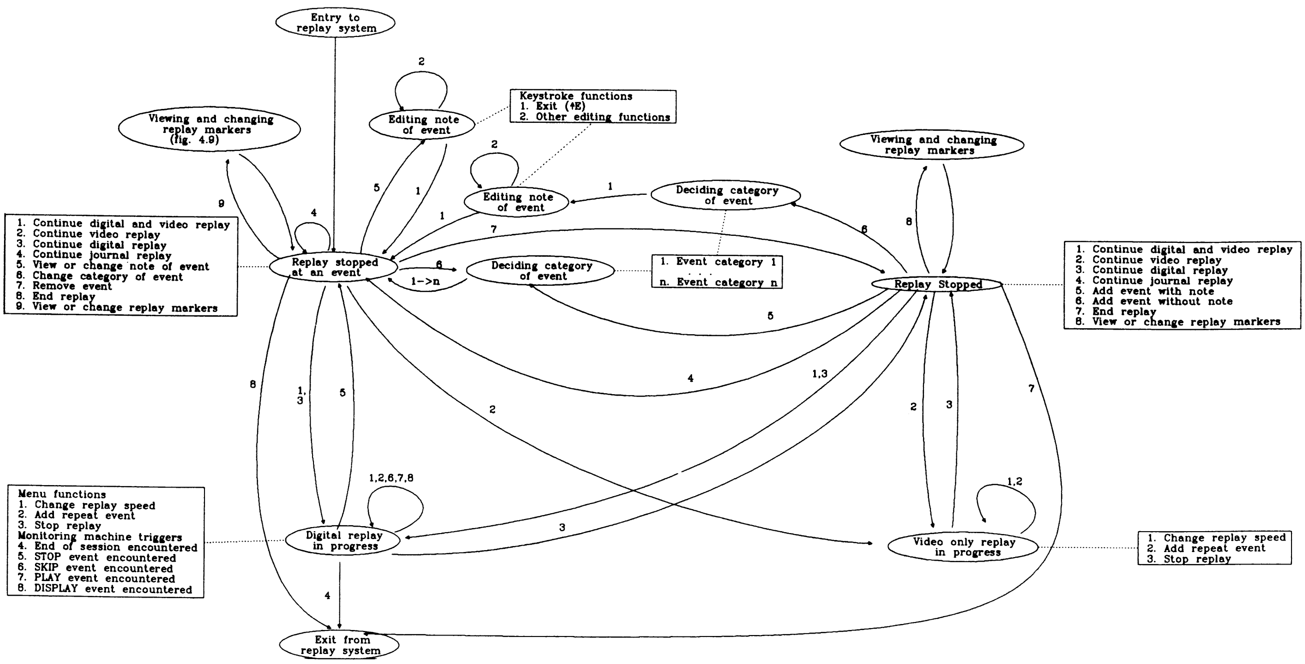

Figure 8. Summary of replay system facilities.

volume of output can be handled. However, simply counting the number of characters output works just as well in practice. In either case the regeneration of captured input can be delayed until the output state matches that at capture time.

This technique solves some but not all problems. For example, the output of prompt messages may be a variable quantity, and where type ahead can apply, the application software often consumes all the pre-typed input before issuing the next prompt. The amount by which it needs to vary need only be of the order of 1 instruction time in the limit. This is enough to displace the timing of an interrupt by 1 instruction, which might delay the response to the interrupt by a much larger time, if for example that instruction is in the operating system and it inhibits interrupts.

Thus it was concluded that the only safe approach to capturing and replaying a multi-user system is to capture both the user input and the full computer response. This is possible with a character terminal, although the volume of computer response may be quite large. If the terminal connection is full duplex it is only necessary to replay the computer responses in order to regenerate the original screen behaviour. This has worked satisfactorily, and in some respects is more convenient, since only a terminal is required to perform a replay.

Experience with single-user workstations having memory-mapped graphics output has also revealed problems. The capture of the graphics output to the user is not feasible with the present techniques. It is also necessary to capture fairly high-data-rate input from a pointing device (e.g. a mouse). Two distinct cases have been explored in detail, an Atari 1040 ST and an IBM PC-XT. In both cases there is a certain amount of unpredictability in these responses to the high data rate of the pointing devices. Since there is usually feedback it would not normally be noticed by a user, who would adapt instinctively.

In the case of the Atari 1040 ST the hardware utilises an I/O processor to handle both the keyboard and the mouse. This communicates with the host by means of an asynchronous serial protocol, which is easily tapped and converted to RS 232. Thus there is only one channel to capture and replay. The PC keyboard is connected by means of synchronous data and clock lines, and the translation of this to and from RS 232 requires a line tap which contains an I/O processor and suitable software in EPROM. In principle it seems that a replay of the captured input with precisely the original timing should work. Again, satisfatory results might be expected with the timing accurate to one millisecond, provided the correct ordering of mouse input relative to keyboard input is maintained. This was not found to be true in practice. Two situations arise which are functions of the way in which the system software is organised.

First, this software sometimes inhibits interrupts, for example to perform screen-management tasks. This occasionally causes input from the mouse to be lost, but it is still captured by the HCI Monitor. As before, timing shifts of the order of one instruction time are enough to cause different results at replay time as a result of characters previously missed being noticed. Even very accurate time-stamping would not eliminate this problem, since the user system and Monitor system clocks are not synchronised.

The second feature that produces variable behaviour stems from the internal organisation of the software. Interrupt procedures deal with the input from pointing devices and represent it in the form of 'current position variables' in memory. The application software samples these variables. As before, very small differences in timing can yield different results if the variables are sampled at about the same time as an interrupt.

Some experiments have also been carried out with intelligent graphics terminals connected to a host. This situation exhibits both classes of problems since the screen display is affected directly by user input and most output.

7. CONCLUSIONS

In spite of the difficulties and with only subjective appraisal of the replay the technique has been valuable in generating feedback to designers which makes a positive contribution to achieving a good level of usability. However, it is now clear that playback through the user system has inherent limitations, and two clear pointers for future work emerge.

First the replay and analysis is very labour-intensive. Attention is now being directed towards tools customised to assist this task.

Secondly, replay of the computer output is the only satisfactory way of guaranteeing reproducibility. This raises difficult technology problems when the output is to a high-resolution memory-mapped display, and current research is directed towards developing techniques to capture and replay the video signals from the test machine. This is only feasible with specially designed hardware to reduce the high volume of data by detecting and encoding picture changes.

Acknowledgements

The research leading to this paper was supported by the SERC, as part of Alvey Project MMI/091. This was conducted in collaboration with ICL and Husat, Loughborough.

REFERENCES

1. A. S. Neal and R. M. Simons, Playback: a method for evaluating the usability of software and its documentation. *IBM Journal* **23** (1) (1984).
2. *MMI User Manual.* Department of Computation. University of Manchester Institute of Science and Technology (1987).