Pandora – an experimental system for multimedia applications

Andy Hopper

ORL–90–1

Published in 'Operating Systems Review' 12 January 1990

Pandora - an experimental system for multimedia applications

Andy Hopper

Olivetti Research Laboratory
Trumpington St., Cambridge, UK, CB2 1QA
&
University of Cambridge, Computer Laboratory
Downing Place, Cambridge, UK, CB2 3QG
ah@uk.ac.cam.cl

Pandora is a joint project between Olivetti Research Cambridge and the University of Cambridge Computer Laboratory. The project is investigating the use of multimedia workstations in a working environment with particular emphasis on digital video. It endeavours to place a camera on the desktop to make generation of multimedia documents as easy as producing text. We are aiming to produce a number of new applications as well as to provide insights into the way computer systems should be designed.

The project is in three stages. In the first a peripheral, Pandora's Box, has been designed. This box can be attached to any one of a range of workstations and provides multimedia features. In the second stage a number of such systems are being deployed amongst a community of systems developers and application writers. Finally we will use the experimental system in our daily work to evaluate new applications. From the users point of view the normal workstation environment is maintained but additional features are available.

This paper describes the design decisions which must be taken when incorporating video in a workstation. A description of the Pandora's Box peripheral which provides multimedia features is given. Finally the distributed system under construction is discussed.

1 Video Features on a Workstation

Choices

There are a number of ways of providing moving images on a workstation. In the simplest and most general approach the output of a video source, a camera for example, is digitised and interfaced to the workstation bus. Writing at high speed to the workstation graphics card can then be attempted thus generating successive frames as shown in Fig 1a. At present this is not possible for large images and it may be impossible to mix some workstation graphics modes with the colour requirements of moving images.

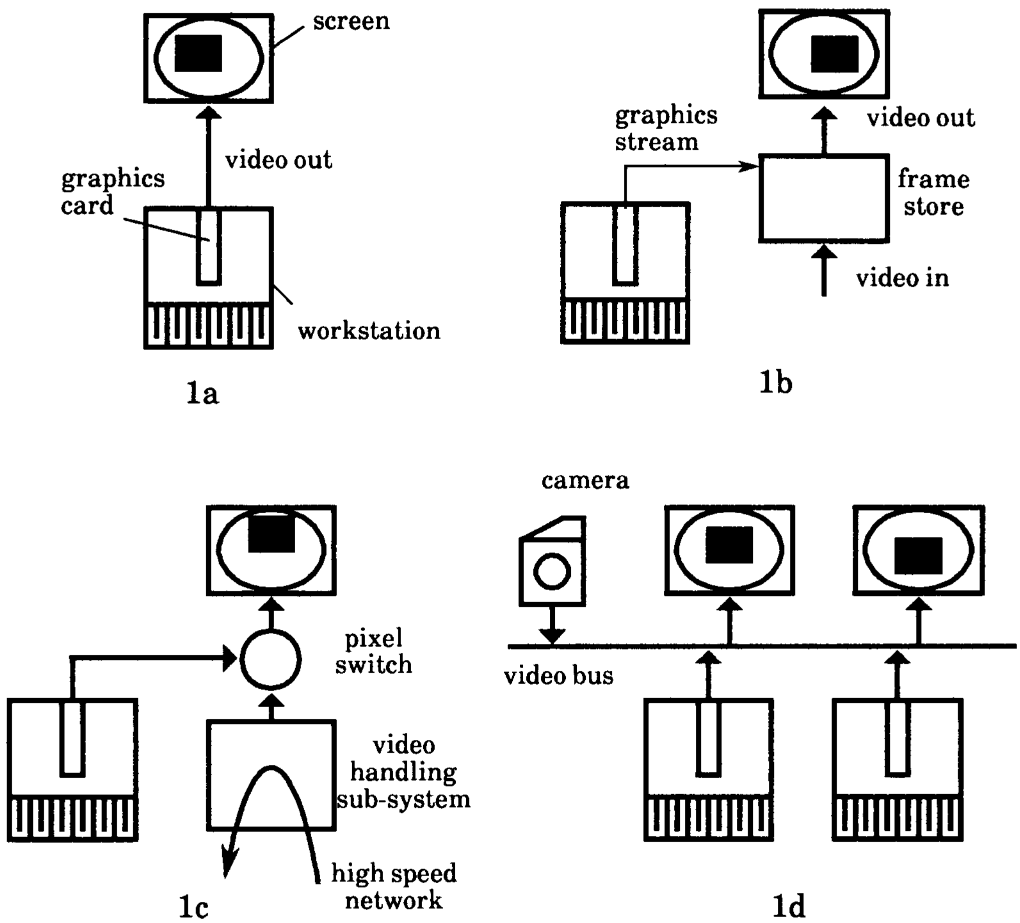

Fig 1 Incorporating video in a workstation

A second approach is to digitise the analogue video stream into a special frame store [1]. In this approach, as shown in Fig 1b, the operating system is required to use the new frame store as its graphics card. The digital bandwidth of the incoming video source is very high and the full speed moving data is not available to the workstation. However control of the analogue source is possible and static images can be transferred into the workstation for processing.

Another scheme shown in Fig 1c is to provide an analogue mixing circuit which chooses for each pixel whether to display the workstation graphics stream or some other analogue video source. With this approach the properties of the workstation graphics system are separated from the video system. It can be extended so that instead of sourcing directly from a camera a sub-system is built which itself is networked and has other input and output devices available. This approach has the disadvantage that it is more complex, but it is potentially more flexible, particularly if the handling of the video stream is digital. As speeds increase it may be possible for more and more of the video data to be processed within the workstation. For the

Pandora's Box peripheral we have decided to adopt the mixing approach of 1c. This allows us to use current workstations and graphics controllers while permitting unrestricted handling of the digital video paths.

Yet another way of handling video is to separate screens from sources by a very high speed network as shown in Fig 1d. Each source and sink of video streams is put on the network so that what is shown on a particular screen is controlled by the workstation, but can be sourced anywhere. This allows all video sources to be available equally and would make good use of any special resources on the network used for manipulating images. This approach is very general because it standardises at a low level making it possible to minimise the complexity of the mixing circuitry. Although, it requires an enormous amount of bandwidth, this can be reduced in applications where one image is being used in several places if frame stores are mapped in a global address space and techniques similar to snoopy caches are applied. Because the high bandwidth path is available globally, the emphasis can shift so that most of the screens become terminal oriented and the workstations are placed in a rack.

Sub-systems

It is unlikely that moving images can be handled directly by the present generation of workstations and operating systems. Thus a sub-system is required to deal with video streams and the control of multiple real time paths. The workstation can then be used for high level control functions and for interfacing to conventional software environments while the sub-system concentrates on compression and control.

The simplest approach is to use two screens. One is driven by the workstation using its graphics card and the other is driven by the video sub-system. The two are only loosely coupled and the sub-system can be treated as a sophisticated player. The workstation can control the player and for some applications this may be sufficient.

A second approach is to mix the two video streams onto a single screen. For monochrome images a control mask plane can be used to choose the source for each pixel, whereas for colour a chromakey technique is more appropriate. If the positioning of images can be controlled the video streams can be inserted in areas which correspond to video windows coming from the workstation. This has the advantage that any windowing scheme that is used will appear to have video features.

In Pandora's Box we have decided to concentrate on a control sub-system and to make it workstation independent. A specific compression scheme is used but the design allows easy incorporation of other compression systems as required. Pandora's Box is a peripheral device to which we can attach a camera, a microphone and loud speaker, a high speed network, an output screen, a number of sensors and a PABX. It takes, as one of its inputs, the video feed from the workstation and is able to mix in moving images from other sources. It produces a combined outgoing video stream which goes to the monitor. At the highest software level an extended X system is used to control applications. The user is thus in an X environment with Pandora's Box supplying the contents of X windows which show moving images. Fig. 2 shows a fully configured Pandora's Box as seen by the user.

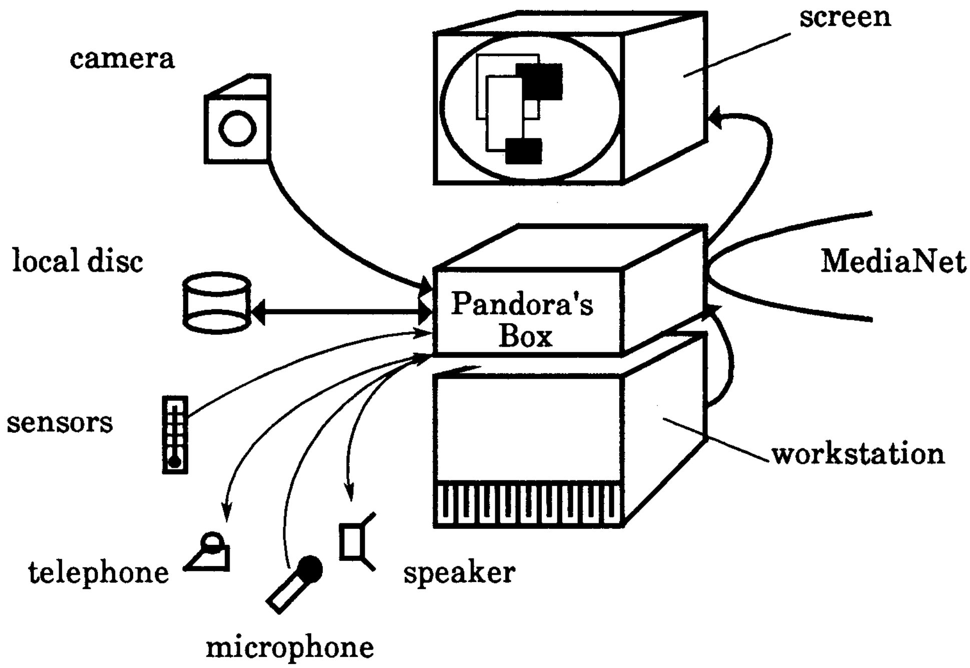

Fig 2 Pandora's Box

At present the sub-systems in use for digital video streams concentrate on either compression or control. Simplifying such sub-systems will require better performance of the host workstation in terms of response to interrupts, latencies, data delays and so on. Improvements in operating systems and the natural progress in workstation design with faster CPUs, better graphics hardware and higher internal bandwidths will make this feasible. Eventually it should be possible to attach hardware which generates digital video streams directly to the main workstation bus and to treat streams generated by these units as another type of data to be handled.

2 Compression

Moving images generate large volumes of data. By taking advantage of the properties of the human eye and the substantial intra- and inter-frame redundancy compression schemes can reduce the amount of data required by a factor of over 100. This is an area where many significant developments are to be expected and the design of workstations should take into account different approaches to this problem.

In the simplest scheme the video stream is considered as a point-to-point communication and there is no requirement to manipulate or reuse. Substantial compression removes redundancy and requires that the transmission system is error free or that errors can be corrected. A good example of this is the facsimile machine.

It would be advantageous to design a video stream representation which enables a number of compression standards to be used and which allows manipulation between these standards. The representation should encompass compression which is both information loosing and information preserving, as well as different treatments of colour. It should cater for markers in the video streams which are used for manipulation and control and the internal representation should not be restricted to any particular range of image sizes. The design of such a representation would mean that it can be considered as the master representation of the data and the transformation processes is only performed once on the incoming side, and whenever the data needs to be displayed. With moving images it may be that the mapping between the internal representation and the display is not complete in all cases. A part of the representation may even be in the form of a microprogram which could be loaded into general purpose decompression hardware to allow the proper handling of the video stream in use.

At present the emphasis is on maximal compression to minimise the network traffic or storage. Special compression systems are being developed which are capable of producing TV quality images. However, these schemes either do not permit manipulation or when they do the compression factors are much reduced. If the assumption of asymmetric compression is made then the decompression hardware can be much simplified at the expense of (non real-time) CPU intensive compression such as DVI [2].

Video stream data passing through any system where resources are restricted may have to be compressed so information is lost. This may be known immediately or only when some bottleneck is encountered. It is thus advantageous to make it possible for the compression to be increased while a stream is passing through the system. Such repeatedly compressed data should retain enough structure so it can be used in the same way as earlier singly compressed versions.

Another way of dealing with different types of video stream is to assume a property of idempotency. Rather than having rules for how to use an intermediate representation it is ensured that compressing the same image many times is not disadvantageous. In a strictly idempotent scheme the first time an image is passed through a compressor the data content is reduced but on subsequent occasions it is not changed. In a slightly weaker interpretation the second and subsequent passes asymptotically degrade the image to a fixed point after which it does not change. For example, consider an incoming piece of video mail which is to be reduced in size and used as part of the screen on a new outgoing video mail message. With idempotent compression the whole outgoing message can be passed through the compressor with the new sections of the screen being compressed while the reused ones are unchanged. For simple manipulations this can be achieved by keeping track of histories, but in a distributed system it should not be necessary to keep track of how data has moved round.

The compression/expansion modules used in Pandora's Box have been placed in datapaths so as to allow experimentation with unifying representations of video streams. While the emphasis of the Pandora project has not been on advances in compression we have designed an on-the-fly compression/expansion system which gives a factor of 8 reduction in data size. The method chosen is a combination of intra-frame subsampling and DPCM with 1 bit error diffusion. The scheme is also idempotent providing manipulation maintains the left hand edge of the picture intact. As new compression systems are developed, we plan to incorporate compression and expansion cards based on those systems.

3 Networking

One way to achieve networking is to provide a separate analogue network along which the video streams can run. However, this is likely to pose problems at the interfaces where analogue is translated to digital and so we consider a totally digital networking approach. In general there is a tension between compression and ease of use. The more effort put into compression, the lower the bandwidth requirement. On the other hand the more sophisticated the compression, the more complex the hardware and manipulation of the compressed stream. It is thus attractive to consider the use of a high speed network which avoids problems by providing enough bandwidth and which reduces the need for sophisticated compression technology.

There are three classes of traffic that may be seen on a digital network handling digital video. The first has no real-time requirement but is used to transport files which contain image data. Such files are likely to be very large, containing tens of MBytes of information, and the underlying network must be capable of supporting many such transfers. In the second category there is real-time traffic which is only being transmitted in one direction. An example of this might be a remote TV camera sourcing a video stream. For such one way traffic, arrival times of packets and consequent jitter of video samples need to be controlled, but the end-to-end delays can be large. The third category of use is where real-time traffic is in two or more directions at once. This not only poses a constraint on the jitter properties of the underlying networks but also on the total round trip delay. This will consist of packetising as well as end-to-end delay and must be minimised to make two way conversations possible.

A recent development in the networking community is the Asynchronous Transfer Mode (ATM) of operation [3]. This uses small packets (cells) to transmit real-time data and the size of the packet is chosen so that delays or occasional loss do not seriously impact performance. By using small packets the delay or jitter budgets can be controlled at fine grain and thus the buffer sizes required can be minimised and managed in simple ways. By using an ATM network the complexity of bridging to ISDN or B-ISDN is much reduced and inter-working with future switch fabrics is facilitated.

We have chosen to provide high speed networking facilities for the Pandora system using the MediaNet (MNet). This network is based on hardware designed for the Cambridge Fast Ring which lends itself well to operation in ATM mode [4]. The MNet is based on interconnected rings operating at 75Mbits/sec and uses a round-robin empty slot protocol. The performance is such that for a system with 20 nodes the best case point-to-point transmission speed is about 20 Mbits/sec and the worst case is 3 Mbits/sec. The corresponding delays of a cell through the MNet are 20 µsecs and 100 µsecs. This means that delays are likely to be dominated by packetisation and buffering even for relatively uncompressed video streams. The basic cell consists of a 4 bit Access Control Field, a 16 bit Virtual Path Identifier, 16 bits used for Segmentation And Reassembly, 32 bytes of data and a 12 bit CRC. Each station is capable of receiving on any of the 2^{16} Virtual Paths and a call is set up by negotiating with a server for which a Virtual Path Identifier is always available. The structure of an MNet cell and the ring topology are shown in Fig 3.

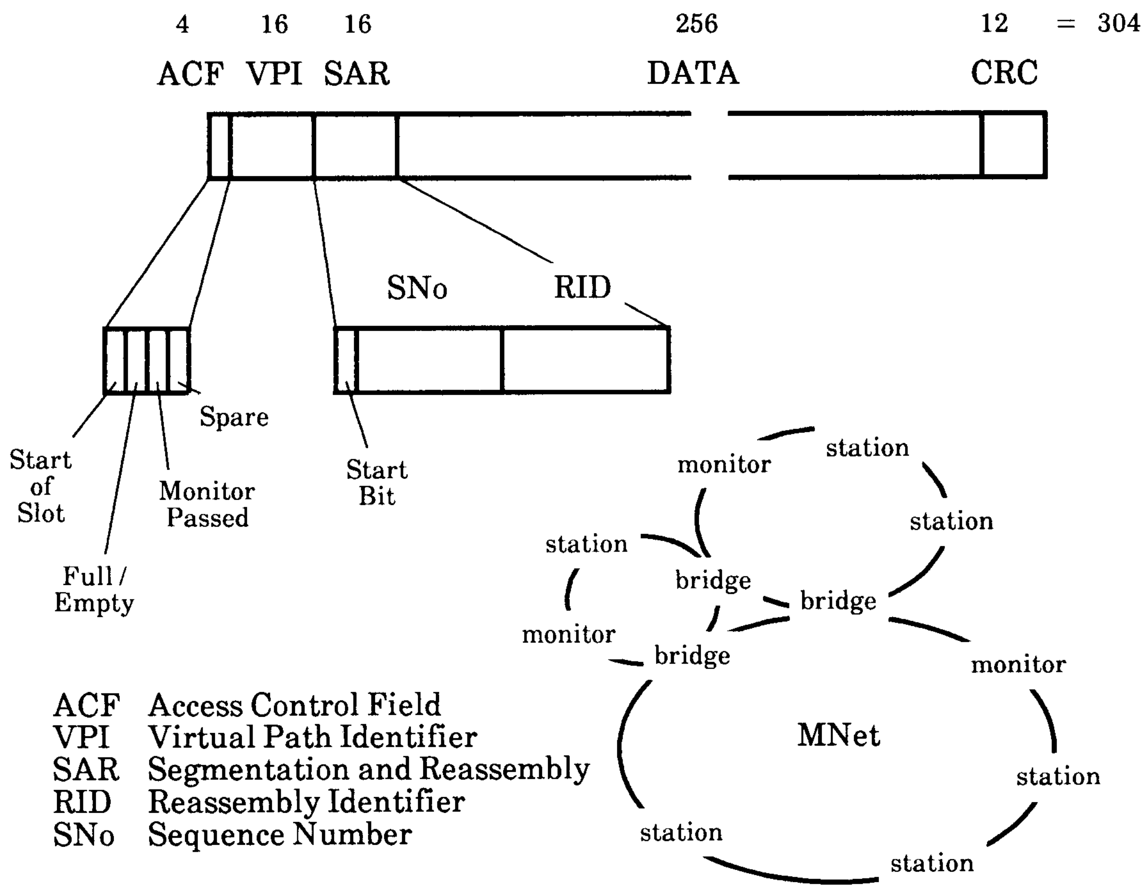

Fig 3 MNet cell format and ring structure

While it is attractive to consider special modes such as multicast and synchronous operation we hope to establish our system on a number of networking fabrics and we are thus not making assumptions about the availability of such features. The only assumption is that the underlying system is high speed and provides fine grain sharing.

4 Pandora's Box Hardware

Pandora's Box is designed to be host independent at both the hardware and software levels. At the hardware level this is achieved by the box being used only for generation of video images to the workstation screen. A switching circuit combines the video from the host with the video from Pandora's box and a map is used to rule on the source of each pixel. To ensure the host cursor is displayed correctly in a window sourced by Pandora's Box the map rule is extended. The Pandora image is only injected if the map is set and the host colour is black, and thus the non-black parts of the cursor show through.

Experience with early systems has shown that it is not necessary to cover the complete workstation screen with moving images [5]. While our emphasis is to use large screens and to allow any pixel to be sourced from Pandora's Box the performance is likely to degrade if large parts of the screen are showing images in this way.

While the basic Pandora's Box is a device for handling networked video and sound streams, we have provided facilities for interfacing it to other communications systems. In particular the voice representation uses a conventional Codec and the host/phone card implements a standard PABX interface. The Pandora's Box can thus be associated with the conventional telephone system to provide either control functions or to form part of the voice grade datapaths.

Pandora's Box is based on Inmos Transputers which were chosen because they perform well as low level embedded controllers and save complexity by using CPU power. The Box takes the form of a multi-transputer system joined by transputer links and a data bus. Some transputer links are allocated exclusively for control of video streams while others are used for general control and error reporting. The high speed data bus is used for transmitting real-time streams between sources and destinations. The bus is called the MediaBus (MBus) and traffic on it is buffered using 512 word FIFOs. The basic Pandora's Box is based on five mother cards each controlled by a transputer and is shown in Fig 4. At the lowest level each of the transputers implements a device driver. These are used for controlling the major hardware components on each card. While the lowest level code is directly associated with particular chips it is also used for controlling dataflow between the major blocks. This includes control of frame stores, compression, expansion, sound, telephone, network and host. A number of daughter modules are also used for ease of upgrade of some sub-systems.

The main datapath in the system is the bidirectional buffered MBus. This is physically identical to the normal transputer bus but the buffering can hold between 1% and 50% of a frame depending on frame size (and compression system) in use. Dataflow is unidirectional on the capture and mixer cards while it is bidirectional on the host/phone and server cards. It is possible to transfer images into the host but this has to be done at slower speeds. For this reason there is no wide path for the decompressed data back to the MBus but a transputer link can be used at slower speeds if necessary. Any size of moving image can be handled by the system up to a maximum of 512x512. A compressed video stream of this size requires approximately 6.5 Mbps bandwidth. The serial equivalent speed of the MBus is 128 Mbps and so it is capable of handling several streams in the 1-10 Mbps range.

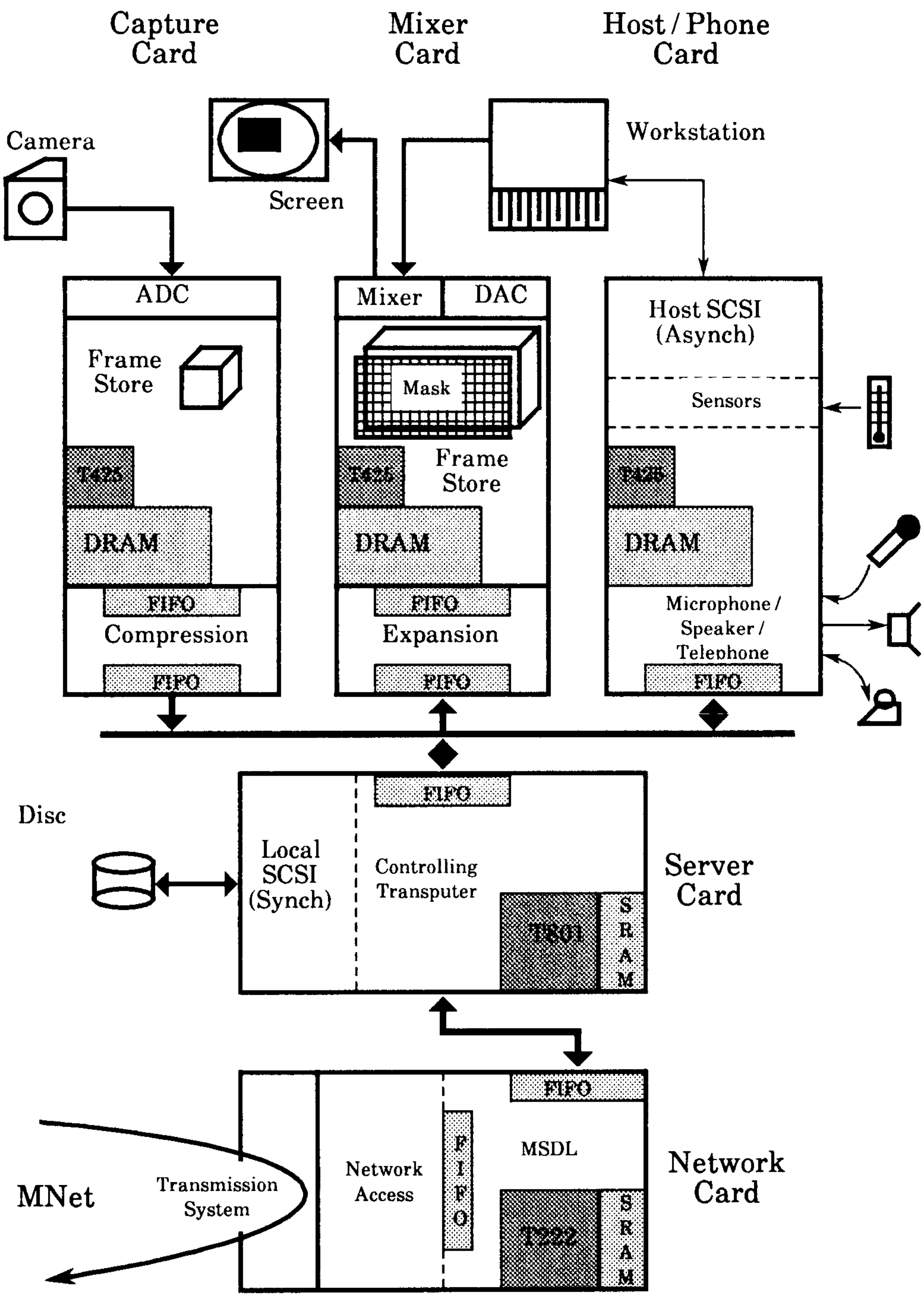

Fig 4 Pandora's Box hardware architecture

The underlying assumption in Pandora's Box is that pixels are implemented in 8 bits. This means that either images are either monochrome or colour is represented in the normal way using a Colour Look Up Table (CLUT). For movies there is no need to change the CLUT more often than once per scene and it is assumed that where colour is needed the CLUT can be updated by the mixer card transputer as frequently as necessary.

Capture card

The capture card is used for digitising images. A daughter module is used for 8 bit D/A conversion and the underlying 512x512x8 frame store can be read byte at a time to give any subsampling required. The frame store is used to decouple the incoming interlaced video stream from the way it is used in the rest of the system. A second daughter module is used for the compression system and either compressed or raw data is sent to the MBus through FIFOs.

Mixer card

The mixer card is used for driving the monitor. Data is taken from the MBus through FIFOs and is either expanded using a daughter module or is fed directly to a 1024x2048x8 frame store. The expansion hardware performs vertical and horizontal interpolation, while writing to the frame store can be used to duplicate pixels. Initially two outgoing frame stores were provided to double buffer outgoing streams but in the latest design writing of frames is sufficiently fast that only one outgoing frame store needs to be used. This card performs some of the low level clipping and cutting operations required by the windowing system. A 256x24 colour look up table (CLUT) is available for applications requiring colour. A second daughter module takes a video input from the workstation graphics card and is used for synchronisation to the workstation pixel clock and for detection of the host cursor. Because it synchronises to the host and generates its own pixel clock it can also be used for simple scaling of pixel aspect ratios.

Host/Phone card

The main purpose of this card is to provide the control interface to the host workstation. This can be done through an asynchronous SCSI bus or by using a transputer link and an adaptor on the host bus. Additionally it performs the voice/telephone functions by handling sound using a 64Kbps Codec and implementing a simple PABX interface. It is thus capable of performing the call and data functions of a telephone. It can also be used as a general input device using a small FIFO or a RS232 serial line and it is envisaged that these will be used for sensors.

Server card

The server card is used as a data switch. The server is based on a high speed transputer with SRAM which is used as the controlling agent for other transputers in the system. The capture, mixer, and voice card FIFOs appear in the address space of the server transputer and data is moved between cards by software copy operations. This card implements a general high speed network interface through which the MNet is used. The server card is also interfaced to a synchronous SCSI bus which is used for local storage. We have found that the main bottleneck in the system is the server and so we are speeding it up by using a faster type of transputer and extending the SRAM size.

Network card
The network card implements the MNet and handles a video oriented block protocol called the Multi-Service DataLink (MSDL) [6]. The interface to the server card is buffered to smooth traffic flows. A daughter module is used to implement either a twisted pair or fibre based transmission system. The control path to this card does not have to use transputer links so that network controllers not based on transputers can be accommodated.

5 Video Streams in Pandora

The main mechanism by which streams are transmitted in the Pandora system is as segments. At present a number of segments make up a frame, but there is no requirement for this particular mapping. There are separate segments for stills, for video, for voice, and provision has been made for other types. The model is that of sources of segments providing streams at full speed to receivers which try to keep up. If the receiver cannot keep up, error messages are produced but at present no feedback system is implemented to slow down transmissions. Each segment has a time stamp which can be used for synchronisation between voice and video and for other purposes.

The underlying hardware makes a large number of stream connections possible through Pandora's Box, particularly if size and frame rate of streams are allowed to change. The primary datapaths are shown in Fig 5. A video sequence from the capture card will typically be passed to the mixer card. At the same time it can be recorded on the private disc and also pushed down to a remote server through the network. The image size may be changed as it moves to these destinations depending on the application. Conversely a stream can be sourced from the network or the local disc and sent to the mixer card. In Pandora's Box the sound is handled by the host/phone card and thus bidirectional datapaths exist between it and the private disc and network. The data content of these sound streams is small, but synchronisation is important. The high speed streams can co-exist with slower speed streams which manipulate low quality movies or stills. On Pandora these can be sent round the network, but more typically they would be routed into the host. If images are being transmitted to the attached workstation then the real-time requirements can be reduced by spooling through the local disc.

Different applications will require a number of these datapaths to operate in parallel. For example a video mail system would send a video stream from the capture card to the local disc and the server card. At the same time sound would be sent from the host/phone card to the local disc. In the video mail editing phase, a stream of video would be sent from the disc card to the mixer card, at the same time as sound being sent to the host/phone card. Finally when the video mail message was being sent it would be transmitted from the local disc down the MNet or into the host and its attached LAN. A videophone application requires the captured video stream to be sent to the network card and the mixer card at the same time as receiving the reverse video stream from the network card and displaying it on the mixer card. Another approach uses a server on the MNet to store and generate video streams thus

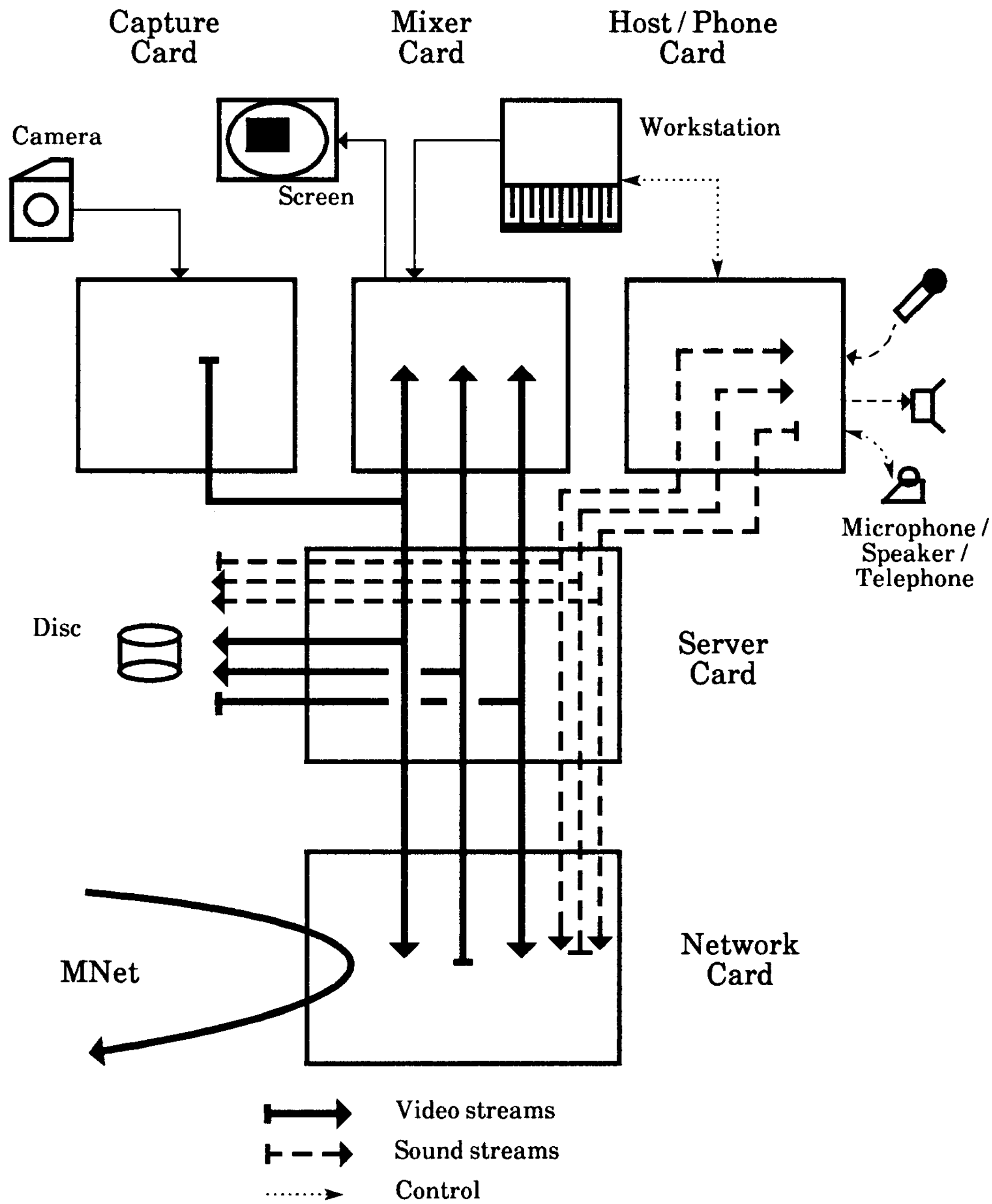

Fig. 5. Pandora's Box datapaths

dispensing with the local disc. We are concentrating on applications which use both sound and video but many others are possible.

Pandora's Box can use different types of hardware, video stream or host and thus a sophisticated control structure and interface to the systems programmer is required [7]. The main control paradime is that of an underlying Factory which deals with objects. The Factory can answer questions and can generate new objects as required by the user. The underlying segment manipulation structure operates in terms of control tokens being sent on links and segments being transmitted on the MBus. The control interface obscures the flow of tokens and segments and presents the user with a model of objects connected together as required. The objects that are produced show or transfer streams and do not necessarily correspond to a single activity at the level below. Consider an incoming stream from the capture card. The Factory would typically create an object for copying data from the camera to another object, the virtual frame store. This level of control does not imply a one-to-one mapping between objects and physical devices. Other objects can be created to send the data from the virtual frame store to the mixer card and elsewhere. Objects could be created for compressing and expanding data, transformation of data, measurement and control of resources and so on.

The underlying hardware makes a certain number only of object structures feasible. There are Pandora objects for specifying the resources in use at present and giving some indication of performance. In general when creating a new object, the assumption is made that enough bandwidth is available and the source transmits at the required speed. If a receiver becomes congested and cannot keep up the performance will drop. This will be reported to the user through an error control mechanism which picks up context as it moves up through the system. The user is then required to choose some compromise, for example reduce frame size, to minimise the number of overruns.

Pandora's Box is seen as a device with a number of features which can be controlled using an object interface. This can be used to write dedicated applications. However we have also provided a higher level interface which fits in with the X architecture and is suitable for providing video features on a Unix workstation. As well as handling items specific to the window system (screen, keyboard, mouse) we have implemented a number of X extensions (XV) to handle items which are specific to the video system (camera, microphone, speaker, disc and network) [8]. We have chosen the XV extensions to minimise the high speed control of the XV server so that the client can be distributed in the normal way. Applications which do not use the XV extensions run without modification. An XV library and toolkit are also being developed to enable the applications writer to be presented with a generic higher level interface.

6 Pandora Distributed System

In the previous sections the design of Pandora's Box was described and now the way we are deploying the Pandora system is shown. A picture of a fully deployed Pandora Distributed System is shown in Fig 6.

By providing local storage it is possible to deploy Pandora systems stand-alone. The camera can record to a local disc from which a transmission takes place more slowly across a network. Providing the underlying network can cope with large volumes such files can be transmitted to other similar workstations which can playback from the local disc. While it is attractive to design a private file system for Pandora's Box we plan initially to use an NFS file server for this purpose. This is being done by interfacing a PC to Pandora's Box through the SCSI bus on the server card and using

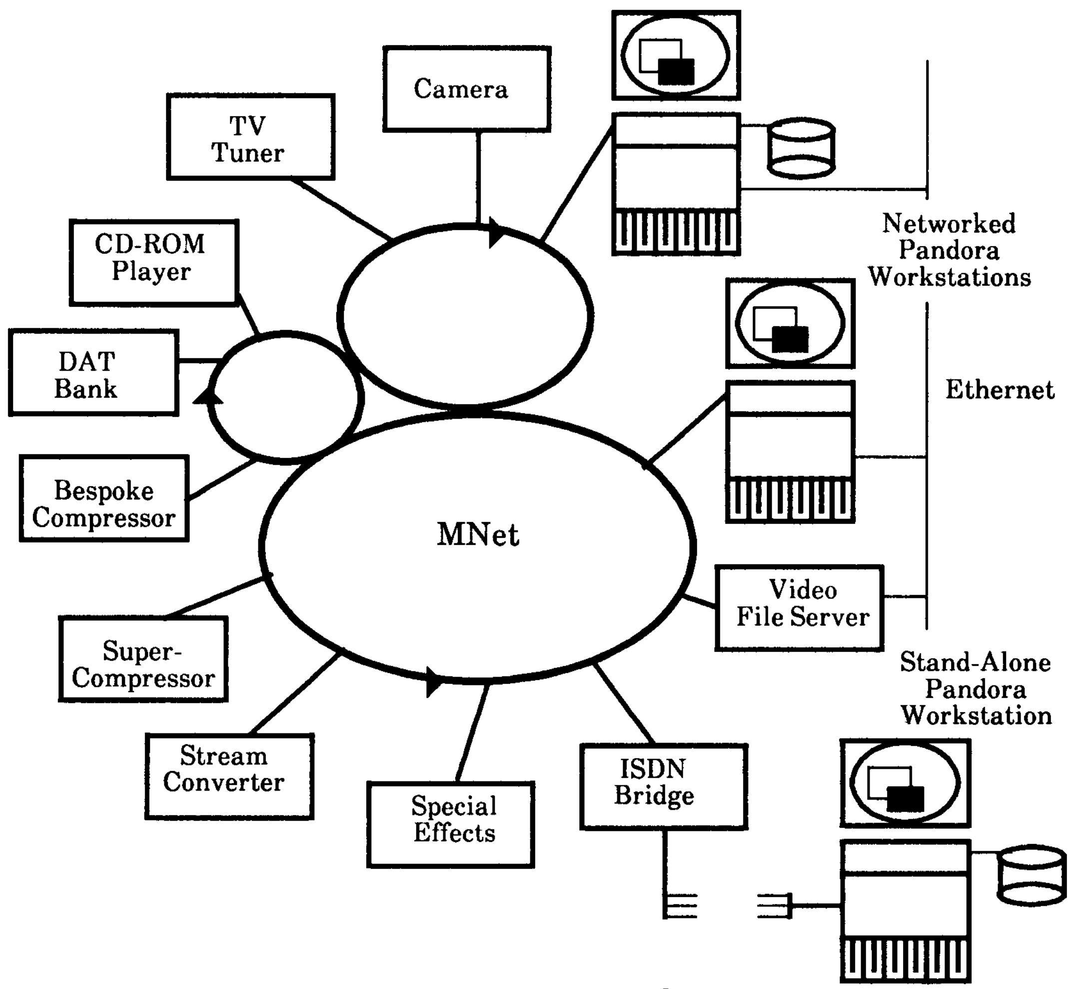

Fig 6 Pandora Distributed System

its file system. If there is enough power in the PC, it may be possible to combine the file system and the XV-server thus producing a simple host system which acts both as an XV-terminal and the local video filing system. With experience we may be able to run more and more sophisticated XV clients on a single machine or alternatively to make the filing system available across the network.

For networked applications Pandora's Boxes are being attached to small workstations and PCs running Unix. These workstations are linked by ethernet in the normal way while the Pandora Boxes are linked by the MNet. The high speed traffic is sent on the MNet using the Multi-Service DataLink protocol while TCP/IP and NFS are used on the ethernet side. Real time traffic can be sourced on the MNet either by cameras attached to Pandora Boxes or by servers which interface remote cameras, TV tuners and devices like VCRs and CD-ROM players.

A video file server is under development which will have an interface both to the MNet and to the ethernet. The video server will be able to communicate simultaneously with a number of devices at real time rates. It is based on a PC and will be able to provide services to both networks and possibly gateway between them. It will also enable us to compare the use of packet and ATM networks for real-time streams. The video file server may keep a cache of the most recently used frames ready for delivery again. The caching algorithm may take into account the highly sequential mode of access to these frames or other patterns to facilitate fast scanning through video files. The frame cache may itself be a separate server on the communications system. Longer term and volume storage may be provided by a bank of Digital Audio Tape machines.

The underlying questions in considering servers are whether the services they provide can be done across a network with acceptable delay and whether special hardware is required. With this in mind we are considering adding servers which provide specialist services. An example may be a video stream converter. This server could provide a central facility for conversion of video streams to be used where the additional delay did not prevent two way communication or it did not matter. If delays prohibit the use of centralised servers for this purpose it may be necessary to deploy the video stream converter between the MNet and each Pandora's Box. Additionally we may provide servers built on special hardware for services like bespoke compression, super-compression and special video effects.

As the deployment of Pandora proceeds we will use multiple MNet rings linked by a fast MAC-level bridge under development. It is also intended to bridge to ISDN with on-the-fly conversion to ISDN formats. This may require dynamic subsampling of the video stream if the bandwidth provided by the ISDN link is too small. The PABX interface will allow us to connect to and enhance applications which involve the telephone system.

The Pandora system allows many types of application to be attempted. For example documents which contain moving images can be generated easily. Imagine this paper arriving electronically with a video introduction by the author. A simple video mail system would enable short clips of maybe one or two minutes to be made, to be changed using a simple editor, sent to the recipient, and viewed using the editor. Because we are equipping a group of people with these facilities we can try to video conference even for very short periods of time. It may be that the visual clues produced in this way make a short unstructured conference possible. By combining a video file server with real-time sources (eg TV) we can extend the contents of file stores to include latest versions of regularly updated items. Thus we always have the latest news, the latest weather or the latest message from head office.

We have also implemented a scheme for making it possible to determine automatically who is in the vicinity of a Pandora's Box. This uses personal badges which transmit a unique infa-red code to a receiver in Pandora's Box and which were developed with other applications in mind. Because walls and partitions are opaque to infa-red radiation and transmission range is restricted it is possible establish the position of each badge. The receiver can be positioned so that its field of view corresponds to that of the camera. This makes it possible to annotate automatically a video stream by indicating who was being recorded, who was nearby and any other measurements received by sensors. This may facilitate future searches of video files or it may just make it easier to discover the name of the person at the other end of a video conference.

The Pandora system allows us to experiment with video streams on workstations. We have constructed about 20 systems so far and we will make more in the near future. It is our intention to use them among a community of users in Cambridge both to stimulate and to try out new applications. The Pandora system when fully deployed may show us not only how to design multimedia distributed systems but also which applications are successful and which are not.

7 Acknowledgements

Many people have contributed to the design of the Pandora system. Ian Wilson, David Milway and Roger Needham helped with the initial designs and goals. Tony King, Oliver Mason, Roy Want and Dave Clarke were responsible for much of the hardware design. Stuart Wray, Alan Jones and Tim Glauert wrote most of the code. Ian Leslie, Cosmos Nicoloau, Rob McAuley, David Greaves, John Porter, Dave Garnett and others have commented on the design. Maurice Wilkes made detailed comments on early drafts of this paper.

8 References

[1] "The Parallax 1280 Series Videographic Processor", Parallax Graphics Technical Report, 1987.

[2] "Digital Video in the PC Environment", Arch C. Luther, Intertext Publications, McGraw Hill Book Co., 1989.

[3] "Layered ATM Systems and Architectural Concepts for Subscribers' Premises Networks", J.P. Vorstermans and A. De Vleeschouwer, IEEE Jour. on Sel. Areas in Comm., Vol. 8, No. 9, Dec 1988.

[4] "The Cambridge Fast Ring Networking System", A.Hopper and R.M. Needham, IEEE Trans. on Comp., Vol. 37, No. 10, Oct 1988.

[5] "Experiments with Digital Video for Workstations", I.D. Wilson, D.R. Milway and A. Hopper, Technical Report, Olivetti Research Laboratory, Cambridge, Mar 1989.

[6] "Protocol Design for High Speed Networks", D.R. McAuley, Technical Report No. 186, Univ. of Cambridge Computer Laboratory, Jan 1990.

[7] "The Interface to Pandora's Box", S.C. Wray, Technical Report, Olivetti Research Laboratory, Cambridge, Nov 1989.

[8] "The Olivetti Video Extension for X", T.H. Glauert, Draft, Olivetti Research Laboratory, Cambridge, Jul 1989.