Session IX: Projects II

Chair: Hideyuki Tokuda, Carnegie-Mellon University

Session IX continued the presentation and discussion of multimedia research projects from Session VI. It showed a very diverse set of approaches, but integration of new media in the existing workstation environment was a common theme to all of them. Only some fraction of the session can be captured in this summary because three of the four speakers made use of video presentations (Andy Hopper did not even have a paper) or even plugged their PC into the projector (Klaus Meissner).

Jim Hanks from SUN Microsystems started the session with a talk about "Integrated Multimedia at SUN Microsystems," joint work with David Berry, Thomas Jacobs, and Daniel Steinberg. SUN has a wide range of multimedia product, advanced development, and research efforts underway. These efforts are focused on developing basic technologies to be used as a fundamental platform for multimedia applications. This platform shall enhance work-group collaboration. Jim talked briefly about SUN's Audio Platform, VideoPix, and the Advanced Multimedia Platforms.

The Audio Platform provides tools to edit audio annotations to mail and other documents. Audio data is portrayed graphically as a set of bars and lines, representing segments of sounds and silence. Regions of audio may be selected and manipulated using an interface paradigm similar to text editing. Another subproject of the Audio Platform is Radio Free Ethernet where audio is distributed across the Ethernet. A source node collects audio information and then uses IP multicast to transmit it across the network.

VideoPix is a frame grabber for NTSC or PAL video, realized as an S-Bus card. It can also be used for low-quality motion video and was applied by SUN for first experiments in workstation conferencing.

In the Advanced Multimedia Platforms program SUN has two projects. DIME (the Digital Integrated Multimedia Environment, not to be confused with DiME, the Distributed Multimedia Environment developed at the IBM European Networking Center) investigates the integration of multimedia in all layers of system hardware and software. As timing constraints often make it impossible for an application to directly handle multimedia data, the project concentrates on investigating ways in which the application can control such manipulation, but all data handling takes place at the lowest possible level in the operating system. The COCO (Conferencing and Collaboration) project is concerned with support for collaborative work. It uses the DIME platform.

For most of his talk, Jim concentrated on his own project, the High-Resolution Video Workstation. This DARPA-funded research project is developing a testbed for the investigation of new hardware and software architectures that will enable the use of HDTV-quality video for workstations. The group has developed their own hardware for a SUN 4/470 consisting of a video input processor, algorithm accelerator processors, bulk frame memory, and a video output processor, all connected by a

high-speed bus. SUN sees this project as a technology driver. As Jim put it: "There is so much data, you can't play tricks."

During the discussion, the use of HDTV resolution in a workstation environment was questioned. The point was made that too many applications will not require high resolution, for the remaining others it will be too costly, and for some applications (e.g., in medicine) HDTV resolution may not be sufficient.

Klaus Meissner from Philips-Kommunikations-Industrie in Siegen, Germany, gave the second talk of the session titled "Architectural Aspects of Multimedia CD-I Integration in UNIX/X-Window Workstations." His talk was different from most of the other presentations in that he focused on a system built around a storage utility. The findings presented come from the ESPRIT project MultiWorks which is concerned with developing a multimedia workstation based on high-end PC technology and CD-I. On the PC side, the X Window System with a Motif interface was used on top of the UNIX operating system.

Key to the effort is to use an off-the-shelf CD-I system and to integrate it into a UNIX workstation. The advantage of such an approach is that inexpensive CD-I equipment can be used (marketed at \$799 since October 1991 in the U.S.) and that regular CD-I titles can be played by the system. The system connects CD-I system and the workstation with a SCSI interface. Digital overlay functions for video also had to be provided: The CD-I system plays back analog video which is clipped onto the workstation display. At the highest level, CD-I system and workstation communicate through regular SUN RPC.

Bernd Lamparter of the University of Mannheim reported on "X-MOVIE: Transmission and Presentation of Digital Movies under X," a topic he has work on together with Wolfgang Effelsberg. The general idea of the system is to store, transmit, and present digital films in a computer network. The hardware used in the system is standard hardware as found in typical workstations today. The movies are displayed in a window of the X Window System. This allows full integration with the classical components of computer applications such as text, color graphics, menus, and icons.

One major problem in displaying movies under X is to properly deal with X's color lookup tables (CLUTs). If two subsequent frames are coded with independent CLUTs, the CLUT of the second image has to be loaded into the video adapter card just before the second image is displayed. This causes the first image to be displayed briefly with the colors of the second image. This is visually disturbing even if the images have similar colors because similar colors do not necessarily use the same CLUT entries. The problem is solved if all images of the movie use the same CLUT. This restricts the colors of the entire movie to 256, which is insufficient.

Bernd has developed an algorithm which overcomes this problem by smoothly adapting CLUTs to the changing colors of the movie. In any CLUT, 32 entries are left free. These can be loaded with new colors before a new image is displayed. The second image can use the new colors and all old colors except 32 which are left for loading the next CLUT update. This algorithm is implemented and results in good video quality.

In X-MOVIE, a movie consists of a header, an initial CLUT, and a sequence of images with their CLUT updates. The Movie Server operates on such movie data structures. On request from the Movie Client (= X Server) it can store or retrieve information about a movie, play, stop, move faster or slower, etc. If Movie Server and Movie Client run on different machines they communicate with MTP, the Movie Transmission Protocol, which provides jitter control at the transport interface. Bernd also briefly discussed the transmission protocols used for X-MOVIE.

Last speaker of the session was Andy Hopper of Olivetti Research Center and the University of Cambridge. Andy showed a new video demonstrating his multimedia system, Pandora, in use. Pandora has previously been described in the literature and is one of the most prominent multimedia system today. It is used for audiovisual conferencing and multimedia mail. Pandora relies on special hardware that is added to each multimedia workstation in the form of a "Pandora's Box" which also connects to the network (the Cambridge Fast Ring in ATM mode).

Andy presented some findings from the use of Pandora. He noticed that most people using the system initially complain that the image they see of themselves (in a local loop-back) is not a mirror and that a slight delay occurs which makes the system appear to not lip-sync properly. Users, however, quickly get used to these things.

Pandora users mostly use the system for video mail. Videophone conversations are less popular, the majority of conversations involves two people only. Permanent video links between two offices are rarely used; a browsing feature that allows to look into other people's offices (without listening) is more popular. Live television and radio are not popular, but people frequently access a recording of the latest news which the system automatically provides. This changes somewhat during major news events (earthquakes, wars, etc.). The most interesting finding was that hardly anyone uses video in combination with text – mail remains mono-media.

Note: Andy Hopper's paper is not contained in these proceedings. For information about his work contact ah@cl.cam.ac.uk.

Integrated Multimedia at Sun Microsystems

James Hanko, David Berry, Thomas Jacobs, and Daniel Steinberg

Sun Microsystems has a wide range of multimedia product, advanced development, and research efforts underway. These efforts are focused on developing fundamental platform-level technologies for enabling applications that use multimedia capabilities to enhance work-group collaboration. This paper describes some of Sun's multimedia work: VideoPix, Sun Audio, the Advanced Multimedia Platforms project, and the High Resolution Video Workstation project. It also discusses some lessons derived from them that can be applied to future hardware and software architectures.

Introduction

The current generation of personal computer-based multimedia products focuses primarily on playback technology. This technology is useful in the areas of information archival (e.g. video encyclopedias) and in many instructional applications. However, the playback model is but a subset of what is needed in a workstation environment.

Workstations are frequently used in a work-group environment, where information is shared among a physically dispersed set of collaborating users. For multimedia to achieve its full potential in such an environment, it must support the interactions between a set of users. Therefore, we are developing enabling technologies for interactive real-time multimedia. Furthermore, we assert that in order to be successful, multimedia technologies must provide capabilities beyond the mere duplication of services provided by existing, discrete products (e.g. televisions, telephones, video conferencing devices, or videotape recorders).

In order to properly integrate multimedia capabilities into the workstation, it is essential that media data, such as audio and video streams, be available for real-time manipulation by application programs. Although such manipulation may seem infeasible at the current time, advances in processor and I/O architectures will make it possible in the future. It is important that a foundation for these capabilities be designed into multimedia architectures, so that applications can make use of them as soon as these operations become feasible. Otherwise, the computational element will be forever relegated to function merely as a controller, unable to add real value. Furthermore, it must not be necessary to sacrifice essential workstation qualities, such as interactive multitasking, in order to provide the user with multimedia capabilities.

Sun Microsystems has an interdependent set of multimedia product, advanced development, and research efforts underway. Current products, such as Sun Audio and VideoPix were developed to meet specific user needs, and later integrated to provide limited multimedia capabilities on standard desktop platforms. Present advanced development work demonstrates support for the integration of many aspects of the multimedia problem (such as those used to support video conferencing), by means of add-on cards. Ongoing research efforts are investigating those features that are required to truly integrate multimedia into the workstation environment. A key aspect of the approach taken by Sun's research groups is the introduction of a new data type — Time Critical Media; it differs from the normal data found in workstations (e.g., text or

graphics) in that the data needs to be manipulated and presented correctly with regard to time. For example, in order to preserve the fidelity of a movie, it is necessary to present the visual and audio data back to the viewer in sequence, and with regard to the temporal relationship of one value to the next.

Sun Audio Platform

Sun workstations are currently equipped with voice-quality audio I/O capabilities. Voice messages can be recorded and used for electronic mail messages, document annotation, and sound-tracks for presentations. Audio digitization is performed by an AMD79C30 Coder/Decoder (CODEC). The CODEC samples incoming analog audio data at 8000Hz, with approximately 13-bit precision, and encodes the digital audio data using the CCITT μ -law standard. Simultaneous input and output of μ -law data is supported. All SPARCstation desktop units include a built-in speaker and a break-out cable for plugging in a microphone and headphones.

Audio can be integrated with desktop applications in a number of ways. Authoring applications allow direct manipulation of audio data by providing record, playback, and editing features. Presentation applications, including document viewers and hypermedia tools, often use visual objects (e.g., glyphs) to indicate the presence of an audio attachment. These objects can be activated, by a mouse click for instance, in order to hear the underlying audio data. Other applications may use audio for the delivery of alerts or background status information. Audio can also be used to enhance traditional user interfaces, providing audible feedback that is synchronized with mouse events, for example.

Audio Interfaces

The many ways in which audio data is integrated into applications has some interesting implications in the area of human interface. Consider, for example, the problem of output volume control. A common instinct is to provide a volume slider as part of the graphical interface to programs that use audio. This fits in with the model of the transistor radio: if the volume is too soft, turn it up. This approach, however, cannot support applications that produce audio output without direct user actions (like an appointment announcer or print queue monitor).

One model is that of the component audio system: all audio data is routed through a single control amplifier and therefore share a single volume control, speaker switch, headphone interface, etc. This model can be realized on the desktop by providing an Audio Control Panel application that acts as a configuration controller for desktop audio capabilities. Users then learn to go to one specific place to adjust the output volume, rather than require that every application present its own volume control. One exception to this approach might be found in the control of audio recording levels. Because audio recording generally requires more active user participation, it is usually more convenient to locate the volume control for record level adjustment in the audio recorder application itself.

Another important human interface consideration is that of uniform recording levels. The audio production and broadcast industries go to great lengths to ensure that the volume levels of all audio data conform to a set of conventions. If this were not the case, you would probably have to adjust the volume of your radio for every new song, commercial, or live announcement. In the present anarchy of workstation audio, it is difficult to enforce a standard recording level for all voice messages and annotations. It is important, therefore, that audio recording utilities make some effort to help users obtain recordings at suitable signal levels.

AudioTool

AudioTool is a desktop utility for audio recording, playback, and simple editing. It is integrated with Multimedia MailTool to provide voicemail capabilities. Audio data is portrayed graphically as a set of bars and lines, representing segments of sound and silence. Regions of audio data may be selected and manipulated using an interface paradigm similar to text editing. A selected region may be inserted, deleted or dragged out to another application. The AudioTool controls include buttons for Play and Record, as well as Fast Forward and Rewind. An animated marker indicates the current position during playback operations and a level meter displays instantaneous volume information.

One sub-panel of AudioTool acts as the desktop audio controller, and includes an output volume slider and a speaker/headphone switch. Another sub-panel contains controls for record level configuration, including an Auto-Adjust feature that samples the audio input to adjust the record gain to achieve a normalized signal level.

Radio Free Ethernet

Radio Free Ethernet (RFE) provides a demonstration of one way in which the network can be used to distribute audio information. Based on unpublished work by Steven Uhler and Peter Langston at Bellcore, this facility enables packets of audio data to be broadcast around a local area network, using Internet Protocol (IP) Multicast technology. A workstation can act as a radio transmitter by collecting audio data from its input port and broadcasting RFE packets of data onto the network. Any workstation on the network may then tune in to a particular station by extracting the RFE packets from that station and queueing the audio data to its output device. The network load amounts to roughly 8100 bytes/sec. per active station, though this figure can be reduced by using simple compression and by squelching the broadcast of silent data.

Radio Free Ethernet has been a useful prototype for testing multimedia networking implementations. In addition, it has been used sporadically for the broadcast of radio programs and other music sources, and extensively during significant events such as the Gulf War, U.S. Senate hearings, and the Super Bowl.

VideoPix

VideoPix is a single slot SBus frame grabber for NTSC or PAL video [Sun91a]. The software provided with VideoPix allows users to view the digitized images on their workstation screen locally, or across a network via a video server program. The

principal goal for the VideoPix project was to produce a low cost still-frame video digitizer (rather than continuous video). Therefore, the workstation CPU performs most of the processing necessary to produce displayable images.

Hardware and Basic Software

The VideoPix card is an SBus slave device. The hardware digitizes the incoming composite video and places it into two serial access memory buffers, one for each field of the video data. (This operation can be performed at 30 frames per second.) Once the image data has been placed in the VideoPix memory, the host reads it and performs the transformations necessary to generate an image displayable on a frame buffer. To date, the highest rate achieved for this step is approximately 15 fields per second.

A loadable Unix device driver is provided, along with a software library to aid in programming the VideoPix hardware and two applications: a viewer program, *vfctool*, and a (network) video server program, *nvserver* (similar in concept to [Arons 89]).

The viewer program has two modes: in the first it controls the VideoPix hardware directly through the VideoPix library; in the second it acts as a client viewer for *nvserver*, (taking the image data over the network). The tool can display images in true color, index color (pseudo-color), grayscale, or monochrome (bitmap) depending on the host frame buffer's capabilities. The tool also provides mechanisms to save digitized images as Sun Raster, TIFF or Postscript files.

The Video Server

The video server allows multiple users to independently view images from the VideoPix hardware at different resolutions and image types. For example, user A can decide to view images in grayscale at a resolution of 360×240 pixels, while user B chooses to view the images in pseudo-color at a resolution of 180×120 pixels. In this case, the video server converts the data into both formats and sends the data to the clients. The video server also allows clients to remotely control the VideoPix hardware (e.g., a client may select a different input port or adjust the hue of an image).

The video server is organized in three layers. The top layer handles client connections, using separate control and data channels (sockets) for each client. Asynchronous I/O facilities are used to optimize the data movement. The middle layer transforms the image data from a generic format (4:1:1 YUV) to the form requested by each client. Caching is used, so that an identical transformation is not repeated. The bottom layer is responsible for obtaining the generic image data and managing the hardware. If only one image format is needed by clients, the VideoPix library provides a means for accelerating the computation by combining the capture and conversion process.

Although VideoPix was intended for still-frame capture only, it is capable of reasonable performance on limited-motion video over Ethernet connections. The results in Table 1 show the maximum rates currently achievable on uncompressed grayscale frames using the VideoPix hardware. (The amount of data per frame varies with the size and type of image requested.) The two rows show the performance (frames per second) measured for various frame sizes when two SPARCstation 2 systems were on the same network and when they separated by one gateway machine.

The VideoPix hardware and software has provided the opportunity to experiment inexpensively with networked digital video. Experiments that have been performed include the use of the technology for limited motion video conferences. For example one VideoPix customer held a video conference between sites in Vancouver and Toronto, Canada over a T1 link. When used in conjunction with the Radio Free Ethernet program described in the Sun Audio section, VideoPix provides a means for broadcasting digital video and audio across Ethernet. The combination of the two has been used to broadcast live television news within the Sun campus.

Advanced Multimedia Platforms

The Advanced Multimedia Platforms (AMP) program was formed to investigate the issues involved in integrating multimedia technologies (OS, networking, application support, conferencing, collaboration, etc.) into UNIX workstations. The two projects currently underway in AMP are DIME and COCO.

DIME

Currently, most multimedia applications must deal with data and devices directly, and must be written using intricate knowledge of the specific media. A bridge is needed between the application and the operating system (OS), because each has different views and requirements. The application is interested in dealing with abstract objects and is generally not concerned with how multimedia data is handled, while the operating system has the job of moving the data and managing the underlying hardware and network usage.

The Digital Integrated Multimedia Environment (DIME) project has been investigating paradigms to support the integration of video and audio Time Critical Media within a UNIX environment. The DIME project is approaching the problem as a whole: from the hardware up through device-independent distributed services, application toolkits, and new interfaces for the end-user. Therefore, the support necessary for multimedia applications has been considered from both the OS and application points of view. This broad approach has led to an understanding of how each layer interacts with the others. Furthermore, the DIME project has developed initial implementations at each level of abstraction to validate its architectural model (see Figure 1).

The time constraints imposed by Time Critical Media, and the amount of data associated with it, often make it difficult for the application to deal directly with the data (especially with current OS and I/O architectures). Therefore, the project has been investigating ways in which the application can control the manipulation and movement of data (that takes place within the OS) as if the media data were manipulated by the application program itself. An application can still directly access and process the data, if necessary (incurring the additional overhead).

To gain experience with the problems involved in writing applications which access multimedia, a video editor has been developed. The goal was to produce an editor that was both simple to implement and easy to use. To this end, a number of features have been assembled to allow the user to both view and edit stored video.

The DIME project has produced an experimental SBus card which contains the Intel i750 video processor chip set [Hartney 91]. The i750 provides the ability to capture, compress, and decompress NTSC video in real-time. The compressed video bandwidth (200KBytes/sec.) is manageable within current workstations and networks. This card is used as the hardware testbed for the development of the project's other components.

DIME Support for the Multimedia Application

A major goal of this project is to provide application writers with an environment that supports the following concepts: distribution and location transparency, device independence, resource sharing, and adaptability.

Distribution and Location Transparency — In trying to locate a device (such as a camera or microphone), a user should be able to specify information such as the workstation to which is connected, the user to whom it belongs, or the room in which it is located. Once an application has found a device, its location should be transparent; a remote device should be as easy to use as a local one.

Device Transparency — Similarly, it should not be necessary for the application to understand the peculiarities of each device. For example, there should be a standardized interface for a "camera" to which the handler for each type of physical camera will conform. If the application wants to use a feature of a particular camera which is not found in the standard interface, then it would have to specify the use of an enhanced interface when the device is requested.

Resource Sharing — Even when restricted to a local machine, the user will frequently be running multiple applications which will compete for access to the limited resources on the machine. For example, it is usually necessary to multiplex a speaker device over time in order to service competing applications (i.e. serial exclusive access). However, some workstations might be equipped with hardware which allows audio sources to be mixed. In that case, it may not be necessary to suspend one application to service another, since the output of the each application could be mixed together.

Adaptability — The underlying system should assist in gracefully handling overload periods when it is not able to cope with all of the Time Critical Media resource requirements. The goal is to allow the system to adapt to the changing availability of resources on both the local workstation and across networks, so that acceptable service to all applications can be provided, based upon application defined policies. An alternative system model of peak resource reservation and service denial (for the late-coming applications) was rejected because many applications have variable service requirements (e.g. video can be displayed at less than 30 video frames/sec. and still be functional). With today's powerful UNIX workstations, it is desirable to allow as many applications as possible to run simultaneously with adequate service, rather than restrict users to a few applications with "perfect" service.

To provide this functionality, **resource** and **appliance** abstractions have been developed. A resource represents any hardware device or software module that needs to be shared in any way. An appliance is an instance of a resource; it presents a standardized interface to applications. The Multimedia Data and Services Manager (MDSM) provides

the support for these abstractions. It acts as a bridge between the high level interface that the application uses (via the Multimedia Application Kit) and the actual hardware and software components corresponding to the appliances.

The Multimedia Application Kit

The multimedia appliance abstraction is used by the Multimedia Application Kit (MAK). Applications can allocate, control, and destroy appliances, as well as connect them together to form networks over which Time Critical Media will flow. The application does not have to deal with the media itself. Instead, the OS provides the means for transporting the media data between appliances.

The MAK also enables the application to access control panels that it can present to users. For example, within an application controlling a digitizer board, the user may select video characteristics via a visual interface. The application might present sliders and buttons on a panel which correspond to the options for that board. However, the panel has to be independent of the application because options will vary from board to board. The application can ask the MAK for the proper panel once it has located the corresponding appliance, and if one is available it can be presented to the user.

Multimedia Operating System

The goal of this effort is directed toward providing system support for Time Critical Media. The general problem can be expressed as that of enabling Time Critical Media to be transferred, processed, and delivered to the consumer (either within a single system or across a network) within the media's time constraints. The DIME and High Resolution Video Workstation groups have together developed programming abstractions, called **conduits** and **transducers**, to address these issues. These abstractions are described in the High Resolution Video Workstation section of this paper.

COCO

The Conferencing and Collaboration (COCO) project has been exploring the opportunities for multimedia-equipped workstations to support and augment collaborative work. The project is investigating the needs of the people engaged in collaboration, and developing prototype technology to meet those needs. To this end, it has built a development platform and applications that demonstrate the value of multimedia technology in supporting collaborative work.

The COCO project has chosen to investigate distributed, real-time, computer-integrated, small-group (two to ten people) collaborations using extensible platforms. That is, the COCO project is focusing on augmenting the interactive aspects of the collaborations that occur when a few people come together to tackle a problem, rather than the organizational coordination that often dominates large groups.

The COCO group aims to support "immediate" interaction among collaborators, where actions taken by each participant are presented to all participants as they occur in real time, rather than the asynchronous "off-line" interaction of electronic mail. Video conferencing is a crucial component of the project as it provides some of the immediacy

of face-to-face meetings. In contrast to the traditional approach of using computers to "control" dedicated conferencing systems, the COCO project makes use DIME technology and networking capabilities to permit the integration of video, audio, and textual data within a workstation, and enable real-time conferencing between widely distributed sites. This level of integration permits new functionality including: novel multimedia, multiuser interfaces, synchronizing events in the computer application with the audio/video conference; and journaling audio, video and computer conference text for annotation, storage, and retrieval. The result of this work is an extensible platform and basic applications that allow new or different tools to be quickly prototyped and integrated into the system.

A pilot desktop video conferencing application has been produced using the DIME platform and a "shared whiteboard" application. The environment is also being extended to support generic "collaboration aware" applications. The intent is that applications will be able to support collaborative use without much additional programming effort. Finally, surveys of users (at Sun) of commercially available video teleconferencing systems have been conducted to better understand how well these systems address the users' needs and how future computer based conferencing can be improved.

High Resolution Video Workstation

The High Resolution Video (HRV) Workstation project (funded in part by the U. S. Defense Advanced Research Projects Agency) is developing a research testbed for the investigation of new hardware and software architectures that will enable the manipulation of high-bandwidth, time-critical data by standard workstations. The HRV workstation project's goal is to integrate HDTV-quality video into the workstation environment in such a way that the essence of the workstation is not compromised.

HRV Hardware

The HRV team's goal in designing its hardware platform was to provide a testbed that is sufficiently flexible so that many concepts and trade-offs of integrating high-resolution video into workstations could be explored. Therefore, the hardware is capable of supporting a wide array of video formats (NTSC, PAL, SMPTE 240M) on several types of displays (HDTV monitors, flat panel displays), and can do so at full frame rates on uncompressed high-resolution video data (e.g., 2 million pixels at 72 Hz).

The HRV workstation consists of a Sun 4/470 augmented with several special-purpose units that are connected by a high speed bus, as shown in Figure 2. These units are:

Video Input Processor — A functional unit capable of capturing, digitizing, and transporting high-resolution SMPTE 240M video (as well as NTSC and PAL) at full resolution and frame rate.

Algorithm Accelerator Processors — A computing resource consisting of four Intel i860 processors, running at 40MHz, each with 4 MB of local memory [Sun 91b].

Bulk Frame Memory — A large (256 MB per board), fast memory that can be accessed via the system's high speed bus. This memory is intended to provide, among other things, storage for just over one second (32 frames) of uncompressed HDTV-quality (2K by 1K pixels at 32-bits/pixel) video.

Video Output Processor — A double-buffered, multiple plane-group frame buffer, fully integrated into a standard window system, designed to facilitate the simultaneous display of multiple full-frame video streams. The Video Output Processor (VOP) contains two 24-bit true color image plane groups and two 8-bit index color plane groups. A window ID plane group combined with a window ID lookup table provides pixel-by-pixel selection of the image source and facilitates rapid switching between double-buffered image frames. The VOP is capable of producing non-interlaced high-resolution (2K by 1K pixels) output at up to 72 frames per second. The VOP also contains a 40MHz Intel i860 processor which can be used to control video transfers and accelerate window system and low-level graphics operations.

The high speed bus consists of a front-plane containing a 128-bit wide data path that is clocked at 20MHz (peak throughput is 320 MB/s). The high-speed bus contains no address lines; transfers are set up at the source and destination in a manner analogous to DMA transfers. The bus controls are routed through a set of FIFO's so that the setup for subsequent transfers can be overlapped with data transfers in progress. A separate control bus is provided to streamline this setup process. These facilities are necessary to provide actual throughput that approaches the theoretical peak.

Note that the high-speed bus is (barely) sufficient to update each pixel of a high definition display 30 times per second (at 24 bits/pixel true color), which requires 240MB/s. Although this interconnect is sufficient to perform meaningful experiments in the near-term, it is inadequate for long-term developments. For example, if real-time image processing were to be performed, two bus transactions per frame would be required, for a total bandwidth requirement of 480MB/s. Clearly, the ultimate answer is a switch-based interconnect which can allow many simultaneous high-bandwidth connections [Tennenhouse 89, Hayter 91].

HRV Workstation Software

The main thrust of the HRV software effort is to develop system software that supports multiple, continuous, high-bandwidth, time-critical data streams (e.g., digital video and audio Time Critical Media) in the workstation environment. In order to meet the needs of workstation-based media applications, the project's software efforts have concentrated on the area of system support for time-critical information. This includes the development of new programming abstractions for time-critical applications, and operating system, window system, and network software extensions to support them.

Programming Abstractions for Time-Critical Applications

To provide a natural means for applications to specify time-critical data manipulation requirements, the HRV workstation and DIME groups have developed new abstractions that elevate such data streams to the level of first-class system data types. Just as files

provide access to byte streams and sockets encapsulate networks links, the new programming abstractions of **conduits** and **transducers** [HRV 90b, HRV 91d] provide a means for manipulating time-critical data sources, sinks, and their connections.

A transducer represents a logical source or sink of continuous time-critical data. It can be a real device (e.g., a microphone or speaker), a virtualized device (e.g., a window within a workstation display), or an endpoint of some process (e.g., a digital filter).

A conduit represents a time-critical connection between transducers, and is responsible for maintaining the temporal coordination within the connection (and between connections). A conduit can be a simple connection of a source to a sink, or it can represent a composition of several simpler conduits. In that case, the (outer) conduit is responsible for temporal coordination between the underlying connections, even when the connections are distributed over many workstations. For example, it can ensure that a video stream remains coordinated with a separate audio stream. Conduits achieve this distributed synchronization via a software analog of a Phase-Locked Loop, which uses a stable local time base and error feedback to provide synchronization with an external signal.

Operating System Software

In order to arbitrate among the competing demands of individual applications using time-critical data streams, system resources are managed via an approach we call Time-Driven Resource Management (TDRM) [HRV 90a]. That is, all resources are allocated according to application-defined time constraints. Further, when resource demands cannot be met, the system chooses which activities to perform and which to defer based on application-defined policies [Northcutt 87]. When a time constraint cannot be met, the application is notified by means of an exception mechanism, and it can choose to accept a degradation in service, to modify its requests, or to abort its activity.

Information about the timeliness requirements of time-critical information streams can be extracted from media conduits and transducers, in most cases obviating the need for explicit control by an application developer. For example, an application may specify its requirements in a form that is natural within its domain (e.g. 30 frames/sec. NTSC video), and this can be translated at the conduit level into a form that is suitable for controlling system resource management decisions.

As a vehicle for experimentation in the HRV workstation, a mini-kernel (called the HRV Exec [HRV 91c]) has been developed. In order to facilitate the migration of successful results into mainstream workstation products, the Exec is a stripped-down version of a future Solaris (SunOS) operating system release that includes POSIX real-time extensions. In addition, a lightweight message-based IPC facility has been developed to support time-critical applications [HRV 91c]. The IPC facility introduces minimal overhead message passing, and is designed so it can be implemented in shared memory systems in a way that avoids all data copying operations (without requiring virtual memory system intervention). It also supports the notion of time-constraints as explicit attributes of messages, in order to support TDRM in distributed applications.

The HRV workstation project has an ongoing research effort into the practical applications of TDRM. Among the areas of current research are processor and system bus scheduling. Some of the pieces of information necessary to properly schedule activities with respect to time are: *timeliness* — when the activity must be completed; *importance* — how important completion of the activity is to the application or user; and *resource requirements* — how long the activity will take [Jensen 75]. This is different than traditional real-time approaches that attempt to combine into a single value (a priority) the orthogonal attributes of importance and timeliness.

Extensive simulations have been performed on processor scheduling algorithms. The preliminary results show that there are efficient TDRM-based heuristics which, when resources are available, can complete all activities within their allotted times, and when an overload occurs, can ensure that the important activities are completed (by deferring or aborting activities of lesser importance). However, the results also show that the effectiveness of these methods in inherently non-deterministic environments (e.g., a workstation) depends on having distinct importance and timeliness attributes directly available to the scheduling algorithm.

Window System Software

Another focus of the project's software effort is to fully integrate video into the standard window manager. Unlike text and graphics windows, windows which display video information may not be paused arbitrarily while a window server performs screen management functions. The HRV group, in conjunction with other groups at Sun, is defining a uniform way to identify video window attributes which can be exploited by window systems to handle video windows more appropriately.

Because the Video Output Processor contains a general-purpose CPU, the HRV workstation team has also taken the opportunity to experiment with the optimal functional split between window servers and accelerated frame buffers. An interesting result of this effort is that the general-purpose i860 processor, running straightforward C-language code, has achieved better 24-bit text and 2D graphics performance than 24-bit frame buffers using specialized hardware and hand-crafted microcode.

Network Software

Because current computer communications networks have not been designed to support media transport, resource management decisions, such as bandwidth or buffer allocation, are made without regard to the needs of time-critical applications. The HRV workstation project is investigating the application of Time-Driven Resource Management techniques to network resources. This should enable networks to meet the applications time constraints when possible, and deal with overload conditions in a manner consistent with the requirements of the applications. However, the lack of a global time base complicates the direct application of techniques developed for local environments. To address this issue, the HRV team has begun a collaboration with research teams at BBN [Escobar 91] that have been looking into similar problems. Another collaborative effort is underway with a group at USC-ISI to extend their work in audio and video network protocols [Topocic 91] with concepts from Time-Driven Resource Management.

Conclusion

Sun Microsystems has a wide-ranging program of research and development in multimedia issues. Many compromises were made in current products' hardware and software architectures due to cost and time-to-market pressures, and these ultimately affected their performance. Within the advanced development and research projects, other trade-offs have been made due to development cost or component availability, or in order to provide sufficient flexibility so that various approaches may be fully explored. Nonetheless, these projects have produced valuable insights into the proper approaches to multimedia system architectures.

One interesting result is that specialized hardware to accelerate certain operations can rapidly become irrelevant due to algorithmic advances or improved general-purpose processor speeds. Another is that, in order to achieve the number of concurrent operations necessary to support advanced multimedia applications, workstations must evolve from bus-based architectures to switch-based architectures.

References

- [Arons 89] B. Arons, C. Binding, K. Lantz, and C. Schmandt
The VOX Audio Server
 Multimedia '89: 2nd IEEE COMSOC International Multimedia
 Communications Workshop, April 1989.
- [Escobar 91] J. Escobar, D. Deutsch, and C. Partridge
A Multi-Service Flow Synchronization Protocol
 Technical Report, Bolt Beranek and Newman, March 1991.
- [Hartney 91] K. Hartney, *et al*
The i750 Video Processor: A Total Multimedia Solution
 Communications of the ACM, April 1991.
- [Hayter 91] M. Hayter and D. McAuley
The Desk Area Network
 Technical Report No. 228, University of Cambridge Computer
 Laboratory, May 1991.
- [HRV 90a] High Resolution Video Workstation Project
*High Resolution Video Workstation: Requirements and Architectural
 Summary*
 HRV Project Technical Report #90051, May 1990.
- [HRV 90b] High Resolution Video (HRV) Workstation Project
*System Support for Time-Critical Media Applications: Functional
 Requirements*
 HRV Project Technical Report #90101, November 1990.
- [HRV 91c] High Resolution Video (HRV) Workstation Project
High Resolution Video Workstation: Executive Software Specification
 HRV Project Technical Report — in preparation, August 1991.

- [HRV 91d] High Resolution Video (HRV) Workstation Project
High Resolution Video Workstation: Video Programming Abstractions
HRV Project Technical Report — in preparation, September 1991.
- [Jensen 75] E. D. Jensen
Time-Value Functions for BMD Radar Scheduling
Technical Report, Honeywell System and Research Center, June 1975.
- [Northcutt 87] J. D. Northcutt
Mechanisms for Reliable Distributed Real-Time Operating Systems: The Alpha Kernel
Academic Press, Boston, 1987.
- [Sun 91a] Sun Microsystems, Inc.
VideoPix Technical Information Document
Part No: FE297-0/15K, January 1991.
- [Sun 91b] Sun Microsystems, Inc.
VX/MVX Visualization Accelerator
Sun Technical Manual #800-6432-10, August 1991.
- [Tennenhouse 89] D. Tennenhouse and I. Leslie
A Testbed for Wide Area ATM Research
In Proceedings ACM SIGCOMM, September 1989.

Tables and Figures

Bytes Per Frame	153600	86400	21600
SS2 to SS2: local net	2.4 fps	5 fps	10 fps
SS2 to SS2: via 1 Gateway	1.5 fps	3 fps	9.5 fps

Table 1 Video Server Frame Rates

Figure 1 DIME Architecture

Figure 2 HRV Hardware Structure