



TOUCH... *and* **Go!**

If you want to create interactive multimedia presentations, this guide to nine of the best Amiga authoring systems will put you “in touch” with the tools you need.

By Geoffrey Williams

So you want to put on a multimedia show. Not just any show, mind you, but one that the user can interact with and control. Well, there's a battery of Amiga multimedia software for doing just that. Each one allows you to create on-screen buttons and objects you can click on to control your presentation.

There are two types of interactive multimedia programs (more commonly known as authoring software). Script-based programs are considered the most powerful and flexible, but they are also more difficult to use because you must learn a scripting language in order to employ them effectively. On the other hand, object-oriented programming software (OOPS) allows you to construct programs by manipulating icons, or objects. Instead of a text-based script, object-oriented programs produce an iconic flowchart of your presentation.

(To locate the vendors of products mentioned in this article, see the “Manufacturers/Distributors’ Addresses” list on p. 89.)

Object-Oriented Programming Software

I WILL FIRST take a look at OOPS, because, while generally less powerful, they are much easier to use, and the presentations you create with them are easier to debug. Then, in the second part of this article, I will examine some script-based programs such as The Director, Version 2 (see “Directing a New Script,” p. 37, for more on that program).

AMIGAVISION

AmigaVision (\$149.95, *Commodore*) is at the top of the OOPS list. It comes free with the purchase of an A2000 or A3000, which means not only that it is in the hands of most Amiga owners, but also that there are a lot of AmigaVision presentations available that you can study and learn from.

Having employed AmigaVision on numerous projects—many with tight deadlines—I find it to be the fastest multimedia development product, and the easiest to use. Its interface is friendly, and the flowchart clearly lets you follow what is going on. I particularly like Amiga- ►



Vision's ability to telescope sections into a single icon: This feature enables you to put routines into separate icons, each of which you can drag to a different window and test by itself, and then save and load into other presentations.

AmigaVision also makes it possible—without touching the keyboard—to create scripts by double-clicking on an icon and using the gadgets in its corresponding requester to set the icon's parameters. With ARexx and full laser-disc support, AmigaVision has all of the basics you need to create powerful presentations. In short, if you are working with interactive multimedia, this program is a must.

DELUXEVIDEO III

While the interface of **DeluxeVideo III** (\$99.95, *Electronic Arts*) is icon-based and similar to that of AmigaVision, the major difference between the two

programs is that DeluxeVideo's presentations are time-driven rather than event-driven. This means that while DeluxeVideo can pause to wait for an event, such as a click on a screen button, and then branch to a different part of the script, its icons are laid out along a time line instead of as a series of events.

Only the script-based program **The Director**, Version 2 comes close to DeluxeVideo in the number of screen transitions, wipes, and effects available, and in the ability to scroll SuperBitMaps and overlay various graphic elements to create complex animations in real time from various still elements and animbrushes. This is an extremely memory-efficient way to work with animation, because you can create long sequences and reuse the various elements in memory to develop different effects.

With DeluxeVideo, you can make interactive buttons from regular brushes, animbrushes, or lines of text. In addition, the program has an ARexx send-only port. While it lacks such features as laser-disc control, database functions, and the use of logical operators like If/Then in script branching (because it is primarily designed for linear time-based presentations), DeluxeVideo III is a very strong graphics and animation presentation product with simple button and function-key branching.

PRESENTATION MASTER

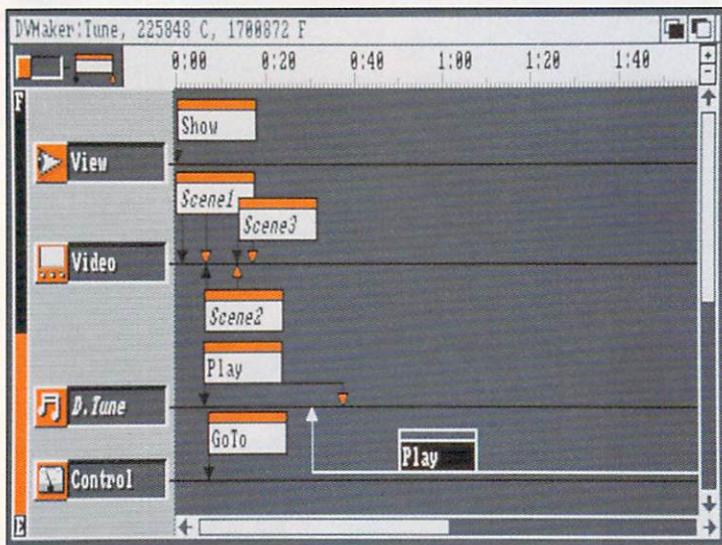
Presentation Master (\$399.95, *Oxxi*) is more reminiscent of the powerful slide-presentation programs available for the PC or Mac than of traditional multimedia programs for the Amiga. While the program's button-making is limited to visible and invisible rectangles, and there is no ARexx or laser-disc support, Presentation Master is still an absolutely critical tool for performing serious multimedia work.

With Presentation Master, you can create Amiga-type presentations that incorporate sound with animations and still images, with plenty of available transitions. In addition to these standard multimedia features, Presentation Master has the ability to let you design slides that are of desktop-publishing quality.

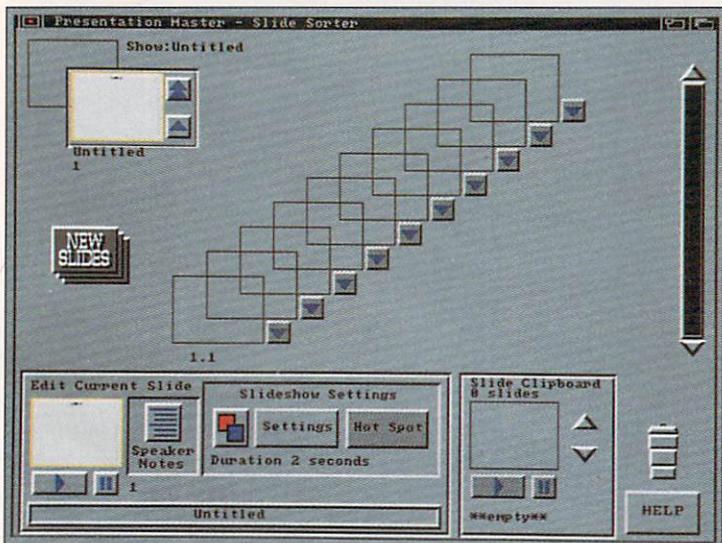
These slides can be shown as an Amiga-based presentation (using dithering to give more apparent colors), rendered to a PostScript printer, or sent to a PostScript slide service so that you can display them at full resolution and in full color (which you can specify using Pantone or DIC colors). You can save images in the major printer file formats for slide-service bureaus, create outlines and storyboards in a variety of formats, and even print miniature images with speaker notes.

The program allows you to create images in an object-oriented paint module, or to import Adobe Illustrator EPS files and render them to the screen. In addition, you can use Amiga bitmap fonts, scalable fonts, and Polyfonts (the Oxxi format), and you can import text into templates and have it automatically formatted with bullets. A spreadsheet that lets you turn numbers (whether from input or import) into a variety of charts—from the traditional bar and exploded pie to Gantt—is also part of Presentation Master.

Included with this product are well over eight megabytes of EPS clipart, bitmap clipart, Amiga fonts, scalable fonts, and templates. Suffice it to say, Presentation



With Deluxe Video III, time is on your side.



Presentation Master's Slide Sorter interface.

Master is a program that all serious multimedia producers need to have.

SCRIPT-BASED PROGRAMS

ALTHOUGH NOT AS intuitive as their icon-based cousins, script-based multimedia programs offer plenty of power.

INTERACTOR

I classify **Interactor** (\$79, *Very Vivid*) as script-based because it uses words rather than symbols. It has an odd, folder-based interface that takes some getting used to, but which is fairly easy to employ once you get the idea. The interface displays only the current operation, so you cannot get lost in a large script, and you can add new options by pointing and clicking with little typing.

Interactor is far from being a full-featured authoring tool, as it has limited sound capability, no ANIM support (it uses its own format from individual frames), and no transitions except for fade. It does have some important and unique features, however. The most significant of these is the flexibility it offers in creating interactive buttons. These can be animated to move across the screen, can move independently with their own velocities, can be affected by gravity, and can be activated by colliding with other objects on the screen.

Interactor seems ideal for creating interactive animation and some rather sophisticated presentations (via its laser-disc support); it can even be used to create arcade-type games.

MANDALA

Mandala (\$495)—also from *Very Vivid*, and with the same interface as Interactor—is an update of the virtual-reality software that the company originally sold years ago. Mandala's reincarnation has strong MIDI capabilities as well as laser-disc support. It also includes an ARexx port and the ability to send commands through the serial port to another device or to another copy of Mandala running on a different Amiga.

The program requires A-Squared's Live Digitizer, which digitizes a video camera's live shot of a person and adds a single color image of the person to an interactive environment. When the boundary of the person's image crosses the boundary of an object on the screen, it triggers an event. You can use a genlock—with or without a chromakey device—if you want to see a live video image of the person on screen.

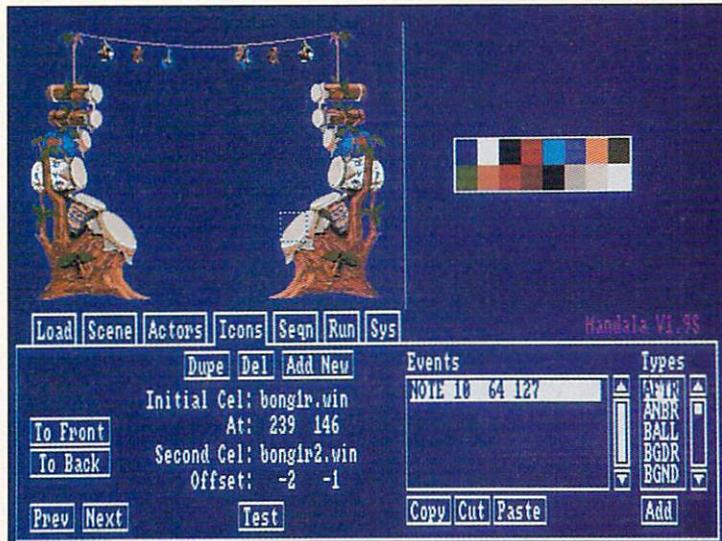
Mandala's power stems from the unique ways it lets you interact with on-screen objects. You can make bells ring, play musical instruments, and even grab objects that stick to your hand. With Mandala, any object can become animated when touched. Moreover, objects do not have to be stationary: As with Interactor, objects can move along random paths, interact with one another, and be affected by such forces as gravity and velocity.

As a virtual-reality product, Mandala is great fun to play with—I feel as if I can step into a cartoon world and interact with it. It is difficult to get away from the cartoon look, however, for the program supports only lo-res graphics. (The company is currently looking into developing a version for NewTek's Video Toaster; that combination would have the potential to offer significantly enhanced features.)

Yet, as it seems to be with all great things, there is a downside. Both Mandala and Interactor sport one of the most insidious and impractical copy-protection schemes I know: the dreaded dongle. If you lose the dongle, which plugs into the joystick port, the product will not run, so you might just as well throw away the software, as a dongle cannot be replaced. This is a frightening prospect.

CANDO

CanDo (\$149.95, *INOVAtronics*) requires that you do some script writing, although you can accomplish the basics for multimedia presentations by simply pointing and clicking on the different options. Most of the actual typing is likely to occur when you employ CanDo's advanced features to create fully functional programs—anything from a calculator to a front-end control panel for ARexx-compatible multimedia programs. ▶



Mandala lets users interact in unique ways.



Interactive multimedia? CanDo!

It is not specifically a multimedia program, as it lacks laser-disc control, built-in transitions for pictures, and music capabilities (except for digitized samples). It can handle the basics of playing animation, sounds, and pictures, and it also has a number of advanced features, such as the ability to grab a section of a picture, display it as a brush, and draw with it.

It also has a complete paint program that can record and play back every stroke in a presentation, and you can draw paths that can be edited for animbrushes. Of all the programs available, CanDo has the most options for interactive buttons, with different events activated by a click, double-click, click-drag, or click-release. In addition, you can make anything into a button, or generate a wide variety of buttons from within the program.

CanDo's real strength is that it lets you create programs that fully support the Workbench environment, with pull-down menus, file requesters, Workbench 2.0 border styles, radio knobs, rollo gadgets and APPEvents, plus multiple windows with scroll, size, close, drag, and depth gadgets. I particularly like CanDo for adding interfaces to CLI-only pro-

gramming and for creating customized interfaces for programs such as NewTek's DigiPaint and ASDG's Art Department Professional. CanDo handles the basics with a few limitations, but gives you a considerable number of additional capabilities that can create very sophisticated interactive environments.

Script writing is made easier with Foundation's ability to load in all the commands from a file requester and then fill in the options. It offers a full script editor, and, as in CanDo, the scripts are attached to interactive objects such as buttons and pull-down menus. Another option Foundation includes is Hypertext, which makes words and phrases in a text file become interactive objects.

This product also has a full range of commands for CDTV, making it a strong CDTV development environment. Foundation appears to have all the necessary ingredients for an authoring program, and while I have not explored it fully, it looks very promising.

THE DIRECTOR, VERSION 2

Considered by many to be the most powerful multimedia product on the market, **The Director, Version 2** (\$129.95, *The Right Answers Group*) also has the steepest learning curve and requires the most typing. To make text entry easier, however, the program provides a text editor that can run your script and show you where errors occur. To help reduce some of the typing, The Director provides several interactive utilities that can be called up from its text editor: The Button utility helps you create interactive rectangular and free-form buttons; the Blit utility lets you cut out areas of an image in memory for later manipulation; and the Polygon utility allows you to create and edit polygons that can be morphed from one into another.

The Director's power comes from the way it manipulates graphics, as it can hold images in memory and grab pieces of them to display in various ways. You can even perform partial page-flipping by using the images from a single screen.

If you need a powerful authoring language that lets you do almost anything with sound, music, graphics, and animation—all in memory-efficient ways—then The Director is for you. If you need advanced and sophisticated interactivity, The Director is again well up to the task. It will take a considerable investment of time on your part to learn this program, but many of the best multimedia developers I know use it exclusively because of its power and execution speed.

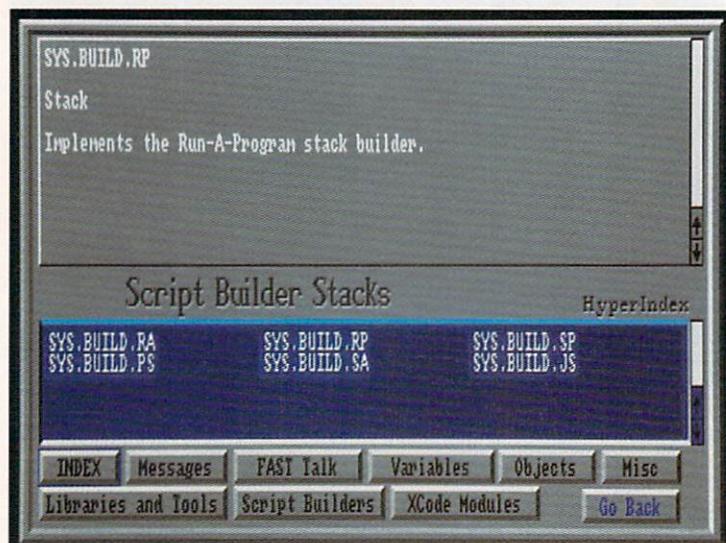
AEGIS VISIONARY

While not intended to be an interactive multimedia product, **Aegis Visionary** (\$99.95, *Oxxi*) does have some unique features. Designed to allow you to create graphics and text-based adventure games, Visionary supports multimedia capabilities such as scrolling SuperBitMaps, animation, sound, speech, and MIDI. And to branch to different areas of the program, you can click on objects on the screen.

Allowing typed responses and then parsing them (reducing them to basic components and interpreting them) is something unique to Visionary. With a little imagination, you can use that feature to create very different types of interactive presentations.

While Visionary is a true authoring language, it does not offer any hand-holding: You use any word processor to write your script, and then you compile it into a working program. ■

Geoffrey Williams is Executive Producer for Creative Business Communications and head of the Amiga Video-Graphics Guild. Contact him c/o AmigaWorld Editorial, 80 Elm St., Peterborough, NH 03458.



You can build on the power of Foundation.

grams and for creating customized interfaces for programs such as NewTek's DigiPaint and ASDG's Art Department Professional. CanDo handles the basics with a few limitations, but gives you a considerable number of additional capabilities that can create very sophisticated interactive environments.

FOUNDATION

Impulse, the creator of **Foundation**, has a reputation for excellent products and exceptionally poor manuals. Foundation (\$250) follows in that tradition. My first glance through the manual left me with a very negative impression, but once I began to play with the program, I gained a much greater appreciation of its power and flexibility.

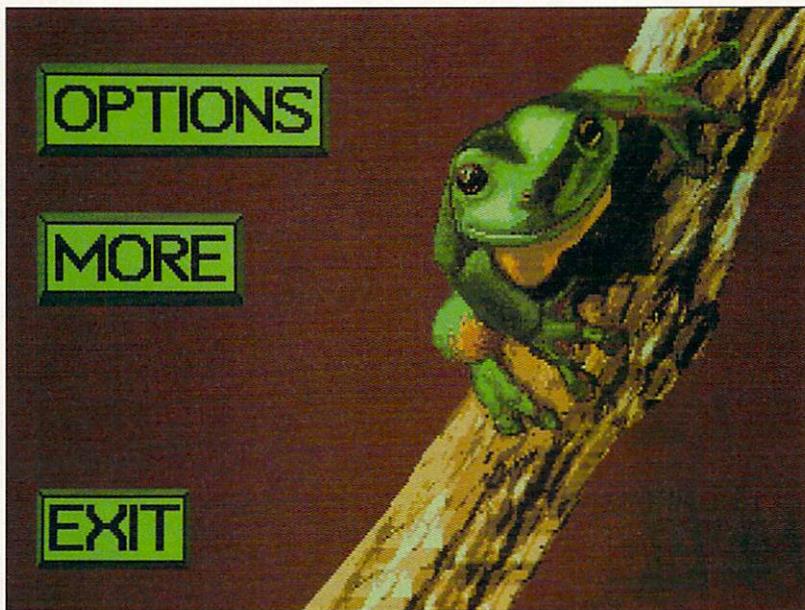
Foundation's authoring language, Fast Talk, has several hundred commands, but the manual makes it a real chore to reference them. Thankfully, the program offers a nice on-line help feature that lets you avoid agonizing over the documentation. The Help notes for each command include examples you can automatically paste into your script and edit.



The PRODUCTION L I N E

The shortest route between conception and delivery of a multimedia presentation depends upon several things. Here is a guide to planning and carrying out your production—whether you are doing it for a client or yourself.

By Robert Edgar



Start at the beginning, go until the end, and then stop. Sounds easy enough. When you are starting out on a new multimedia project, however, it is sometimes hard to know where the beginning and end are, never mind the steps in between. Let's map out the territory of interactive presentation development, taking a look at all the points along the production line.

Because knowing where you are going becomes all the more important when working for a client, I will present the information in those terms, showing what it takes to produce an interactive presentation for a client, and what is involved in such a contract. Keep in mind, however, that when you are producing a project for yourself, you are the client. Thus, it is important to go through the same basic process—asking many of the same questions and considering the same issues—when working for yourself. ▶

Here, then, is a seven-part structure that you might use as a starting place for winning and planning a project. (To locate vendors of products mentioned, see the "Manufacturers'/Distributors' Addresses" list on p. 89.)

1 Determine the goals of the project.

Before you get a job, you may need to bid on it. Before you bid on it, you need to know what the task involves. When you meet with your prospect, don't get so lost in your sales pitch that you forget to lis-

and/or integration? Now's the time to define the scope of the job. (For some specific design concerns affecting your choice of equipment for the bid, see the sidebar "System Integration.")

In many cases, the initial meeting is a time when you must balance the nurturing of the client relationship with giving some free education on the intricacies of interactive video production. You must judge how much time you are willing to spend on the possibility of getting a contract.

This stage is always longer than it seems it should be, and the more expensive the project, the longer it will probably take to close the deal. Six months from your first meeting is likely for interactive video contracts. My experiences show that this time is both well spent and appreciated, and that very few prospects will take extended advantage of your time without giving you the contract. But if the talks stop moving forward, or if the prospect keeps changing his or her mind about basic elements, you may be better off investing your time elsewhere.

2 Put together the project bid.

In bidding a project, make sure you consider every deliverable item for each party throughout the scope of the contract. Specify in writing where and when you will receive content information, who decides on music, where source images will come from, and in what medium and in what size they will be. Also designate responsibility for obtaining written releases for images and music. Specify the dates when you will review the project and what will be covered at each review point; if everything is fair game to be changed at every step, you are in trouble before you begin.

The difficulty of delivering a bid up front is that no one may know enough at that point to make an accurate estimate of the work to be done: not the client, and not you. One approach is to define what you know of the project, and insert a "not to exceed" clause. This means that you will design and produce to a ceiling amount (and it is up to you to protect your profit while keeping the client happy). This has the strong benefit of locking you into the entire project from the beginning, and is probably the route to take with all but the most confused of clients.

A second approach is to bid for a design document that describes the project in detail and contains a bid for the production. The danger here is that the client can go elsewhere for production once he or she has the design document. Try to include an agreement that you will do the production. Use this approach if you feel that the client doesn't know his or her expectations clearly enough to give you a basis for estimating the cost. You might also use this if a client has not decided whether or not to include interactivity and needs to see more of what it would entail.

Try not to underestimate how much time it will take you to produce each stage. After all, it may rain during one of your outside shoots. You may have trouble debugging the program or handling some problem between image palettes. Assume that your most optimistic schedule will not happen. It is better to do so up front than to have to back-pedal near the expected delivery ▶

Title:	Shot #:	Branch From:	Date:
VIDEO	COMPUTER GRAPHIC	TEXT FILE	
Description		File Name File Name Font Name	
Frame #	Programming Notes:	Branch to:	HOT SPOTS
AUDIO Narrator:		1. 2. 3. 4. 5. 6. 7. 8. 9. 10. 11.	
SFX:			

Figure 1. A sample storyboard template, to be used in conjunction with a written script and flowchart. Across the top are three frames for drawing the video, computer graphic, and text elements of a frame (this is assuming genlock-overlay capability). "Video frame #" shows the video-disc start and stop frames for the shot. The effect of a click on each hot spot is indicated in the "Branch to" section.

ten and ask questions. What quality and technology is he or she expecting? How much does the client expect to pay? How near is the client to beginning the project, and how soon must it be completed? The sooner you know the answers to these questions, the better.

To begin the dialog on product cost and quality, I suggest you create a demo reel that displays a graduated range of production values, from the least expensive you would deliver to the most expensive. Prepare to discuss how long each image took to create, what sources you used for imagery, and so on. For each example, have costs ready in terms of video per minute, graphics, animations, coding complexity, and the like. This is not your bid, but a quick lesson for your client on interactive production. The client will want to purchase your best work, of course, but should know up front what it costs, as well as the quality he or she can afford.

What you should know up front is how much of the project the client wants you to do. Will you be responsible for overall design? Research? Scripting? Storyboarding? Graphic and video production? Editing? Programming? Maintenance? Hardware purchasing

date, which puts your client in trouble with his or her own deadlines and managers.

3 Write a content outline.

I suggest that before you create a storyboard and flowchart in detail, you outline the content of your presentation according to what the client provides. This gives the client a chance to make adjustments before you are so far into production that it becomes costly.

In your bid, you might want to freeze the content upon acceptance of this document. This would mean that if the client wants to change the content after approving it, he or she knows that it will mean additional time and cost. Changes to content once you are in production can kill you if you have not raised the issue previously. On the other hand, if you get this freeze in, you do not need to charge the client unless such changes actually do cost you money. Helping a client out can earn good will that might help you win the next job.

4 Create the design document.

The design document is generally the next level after the bid, and the first one to be informed by the actual content you have received. It should reflect your design decisions and bring them to your client's attention so that he or she can approve them before you begin production.

The design document should show the concept of the presentation and your ideas for the way it should be realized. Will you have voice-overs? Is a narrator seen, or always off camera? Where will you use animations and computer graphics? Where will you use video? What kinds of actors will you use, and how should they be dressed? Where will you use humor? Do you have a music library picked out? Where will music be used?

At this early stage, you face a danger in trying to

communicate heavily visual concepts without the help of the actual results. Beware of clever and original ideas that may get tossed if you cannot describe them well. Ride the pulse of this communication carefully; it will make the difference between a genuinely inventive production and a cliché-filled work that does not catch the intended audience's attention.

5 Detail with flowcharts and storyboards.

The flowchart and storyboards, based on the design document, must supply enough detail to allow someone else to produce from them. Storyboards for interactive projects can be quite detailed (see Figure 1). You must not only show what will be on the screen, but also differentiate among various sources. What part of the screen image is coming from a video disc, and what part from the computer? What text is spoken by a narrator, and what is generated as characters on the screen? What areas of the screen are hot spots, and to what point does each hot spot branch?

If you are doing the programming, you may want to use a flowchart in addition to storyboards. There are several flowchart types, as well as tools for designing them.

When I'm ready to consider the flow, I get a pile of 3x5 index cards and write the name of one piece of content on each. Just brainstorm, mix categories unabashedly, and throw them all into the center of the floor. When you have exhausted everything you can think of, find a big table, or just use the floor, and arrange the cards in a hierarchical tree structure representing the flow of the piece. The beauty of the cards is that you can quickly and easily sort out the categories, fill holes and try new structures.

Once you have completed the flowchart, sketch it and begin your programming. This is a great way to proceed from no structure to one that's ready to be realized in an AmigaVision flow—that hidden step not mentioned in any documentation.

Some database structures and hypertextual links ▶

(Text continued on p. 36.)

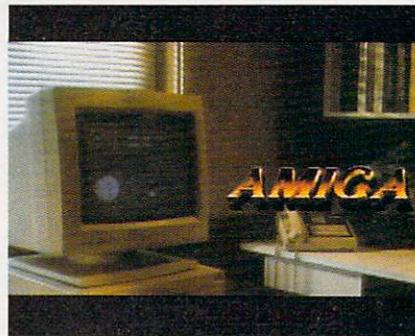


Figure 2. In a point-of-sale system for Commodore, three short video segments introduced each Amiga application area. The first was a testimonial by a professional using the Amiga for that application (left). Then we presented examples of the output for that application, along with features and benefits (center). Finally, we ran an animation that suggested a configuration the customer could immediately purchase to accomplish what he or she had just seen (right). A separate hands-on introduction to multimedia provided a fun way to get acquainted with the system. We provided retailers with access to the code so that they could customize the presentation to meet changing product lines and specific market needs—and many did so.

System Integration

WHAT ARE THE platform characteristics the job requires? The components you choose for delivery of your system will depend, among other things, on your budget, speed requirements, and the scope and setting of the presentation. Here are some of the things you should consider:

SPEED OF RESPONSE

The faster the required response time, the more expensive the platform. You can purchase speed through:

- 1) Memory. With enough RAM, you can load both current and next-in-queue files. This requires tricks for interactive sequences.
- 2) Accelerators. Faster processors can speed animation playback, database searching, and load and save times.

STORAGE DEVICES

Don't even consider using floppy disks. Access time is too slow.

CDTV is inexpensive, but loading files from any CD can be time-consuming and difficult to cover up. On the other hand, delivery of CDTV XL digital video files is a great way to deliver sync motion and audio on an inexpensive delivery platform.

Searches on laser disc can range from shorter than vertical interval (the time between video fields) to over a second, and a major factor can be the speed of the player. In general, you pay for the speed: faster players cost more than slower players. A second approach is to use two video-disc players, having your software continually cue one up while the other is playing.

VIEWING DISTANCE

Will one viewer at a time see your production, or will groups watch concurrently? If you have a large audience, you may need a video projector.

Think about screen interlacing in relation to audience size. A single viewer may be very close to the screen, necessitating a noninterlaced image and noninterlaced monitor. With distance, the jittering effect of screen interlacing tends to diminish, although large monitors can extend the problem for some distance.

Providing a noninterlaced signal while using video for an input can be a problem. Some genlocks that can provide a noninterlaced output will not do so if receiving an interlaced in-

put. One that will is GVP's Impact Vision 24—with the optional RGB decoder—but this is not a low-end genlock, and it may not be applicable for low-budget jobs.

If you need a noninterlaced display but cannot budget an answer, be sure that all horizontal lines are at least two scanlines thick, and that you test all graphics on an interlaced monitor during development.

INPUT DEVICES

The simplest interface for the passerby is a touch screen, but you should add about \$600 to your cost per delivery station if you use one. A trackball is less expensive, but can be offputting for people frightened by technology.

Stay away from a keyboard unless you need the viewer to enter text. Some situations lend themselves to touch-screen emulation of a keyboard (which can be clumsy for a typist, but otherwise fine).

If you have multiple groups watching, will anyone be able to provide input? Do you need multiple input devices?

VIDEO

Don't consider video for a production budget of less than \$2000 per finished minute unless the video is already done, you need less than a minute of video, you own your own production and post-production equipment (and are sure the quality it provides can meet the task), or you are a guerrilla videographer and know from experience that you can produce satisfactory video for such a budget.

If you are going to put the video on disc, you will need a Beta-SP, C, or D2 master tape to send to the disc production house. This does not mean you must shoot in one of these formats, but it does mean a trip to another post-production studio if you don't.

If you need a touch screen and video-disc control, you face the problem of having only one serial port in the standard Amiga. Either you can purchase a multiple serial-port card, or you can look into a touch screen that uses another port, such as that offered by Amigo Business Systems.

AUDIO

What quality of audio is necessary? You can do a lot with the Amiga's eight-bit internal sounds, but if you need CD-quality, you should think about a MIDI

sound box. The power of high-definition audio to improve the impact of a low-definition image is something that you might not expect, but it is in fact considerable.

If you are mixing both video and audio sources, take a look at Progressive Peripherals & Software's Video Blender, which provides for audio mixing, cross fades, and dissolves, as well as video-to-Amiga graphic transitions. It also serves as a genlock, and may be quite cost-effective for complex systems that require smooth transitions.

SCOPE

Will the finished product be installed at more than one site? For multiple interactive kiosks, consider CDTV—especially when multiple-site costs are a prime consideration. I recommend adding any of the following to the basic CDTV unit if conditions require: DCTV (for full-color animations); genlock and videodisc (for full-screen, high-quality video); a MIDI sound box such as the Roland CM-32P (for 16-bit, syncable, high-quality audio delivery driven by Bars&Pipes' MIDI player, the AmigaVision CDTV driver, and an ARexx script); a deinterlacer (for stable image).

OUTPUT

If the viewer needs some type of output before leaving your presentation, you may need to add a printer to your setup. If so, consider the quality of output that is necessary. Will you need to print graphics and video in addition to text? Who will replace the paper, and how will they know it needs replacing?

How about videotape output? I helped develop an interactive marketing system that Georgia Power used with prospective customers. Sitting in a dark room and using two touch screens for input, company representatives would "travel" through Georgia viewing videodisc segments on various aspects of cities and towns, along with data on the economy, interactive maps, available buildings, and so on. Their "tours" were recorded as they watched, and at the end of a session, the visiting representatives could take a video tape with them. This let them present their findings to other decision makers back home, and it extended the impact of the presentation. □

—RE □

From p. 32.

may not lend themselves to a hierarchical format. Your final design may exist partly with index cards, partly on paper, partly in tinker toys, and partly only in your program. Do whatever makes sense.

6 Carry out design and production.

Amiga graphic and video production has been covered well in other articles, but there are some points specific to multimedia—especially interactive presentations.

Before you begin, create a production chart for your work. Your chart should allow you to see all steps in the production of all deliverables, as well as what steps are prerequisites to others, and what processes can bottleneck your project.

How much branching and answer-checking you need may determine the program you use to create your presentation.

Programming tools that are essentially linear, such as ShowMaker (Gold Disk) or DeluxeVideo III (Electronic Arts), may do perfectly for looping presentations, but not for anything interactive (see "Touch and Go" on p. 24 for details on multimedia software). Other programs,

such as AmigaVision (Commodore), CanDo (INOVAtronics), or The Director (The Right Answers Group) are more suited to interactivity. (For more on using the new version of The Director, see "Directing a New Script," p. 37.)

If you need to incorporate product images in the work, try to scan photographs instead of printed images, which may give moire patterns when print dots interact with the dot-and-line pattern of video. Also, be mindful of your screen edges. It is easy to create images that you believe to be full-screen size, only to find that the outside edges do not match up as you advance from screen to screen. It is also important that the edges of the video image match with the edges of the computer graphic screens. Test the images out in an underscan mode of a monitor like the Commodore A1950.

Should you use icons or text labels for hot buttons? Of course, the nature of the job will help you decide, but as a rule, text is preferable, because very few icons can immediately communicate the title of a category. One interesting strategy I have recently seen is to use icons or images for hot spots, but to have the cursor change into a text label (or text otherwise appear) when the cursor passes over the hot spot. This will not work for touch-screen systems, but for mouse or trackball interfaces, it provides for un-

derstandable and graphically interesting screens.

In a related vein, try not to separate labels from their corresponding buttons. Putting labels on the buttons themselves both saves screen space and prevents confusion as to whether to touch the label or the button.

Because RGB computer images are sharper than video images, you should use computer graphics for any still images—and in overlay mode for still text. If your screen is not RGB, but NTSC, I suggest your text be in white lettering, with a black shadow down and to the right, and that you employ very little text (use audio and video segments for explanations).

When you do incorporate video segments, make them short—5 to 30 seconds. The only time you should make such segments longer is when you're hiding computation and file-load times. This is a good practice that demands powerful and relevant video. (A boring piece or an apology for a wait can destroy the experience for the viewer.)

Note that it is possible to economically and intelligently preload files for linear presentations using software such as ShowMaker. Because branching systems cannot predetermine what the viewer wants to see next, you may be stuck with pauses while the system loads a graphic or searches for a video segment. If you have enough RAM, load all of the next possible choices in advance, and follow it with a video segment covering your loading of the next few files.

7 Installation, upgrades, and maintenance.

Consider the setting of your presentation. In placing a kiosk, for instance, beware of reflecting lights that can make screens hard to read. Before the installation is done, you should think about updating and maintaining your presentation systems. Think about how often you will need to update the sites and how you might go about it. You may be able to update from a remote location by installing a modem. A version of GVP's Scala presentation software has been developed just for such distant updating.

Who will fix an image if a green line suddenly appears across the middle of the screen? How are sites to be maintained in general? Maintenance can be an income source for you, but you will need a plan—and possibly another contract—for carrying out the task.

The line from point A to point B in multimedia production is anything but straight. It is filled with options and is often highly complicated. And undeniably, it is also fun to travel. When interviewing for the position of Multimedia Producer for Commodore, I was asked whether I understood what the job entailed. "Sure," I said, "I get to play with the toys." I got the job. ■

Robert Edgar is a multimedia developer for Simon and Schuster Technology Group. Formerly, as Multimedia Producer for Business Markets at Commodore, he worked on a mixed-mode CDTV and video-disc demonstration system, an interactive point-of-sale video-disc system for demonstrating the Amiga in retail stores, and the A3000 launch "Multimedia Live" video. Robert is an exhibiting artist, and has performed with his "Living Cinema" performance system throughout the US.



DIRECTING A NEW SCRIPT

Powerful new additions to a multimedia-authoring

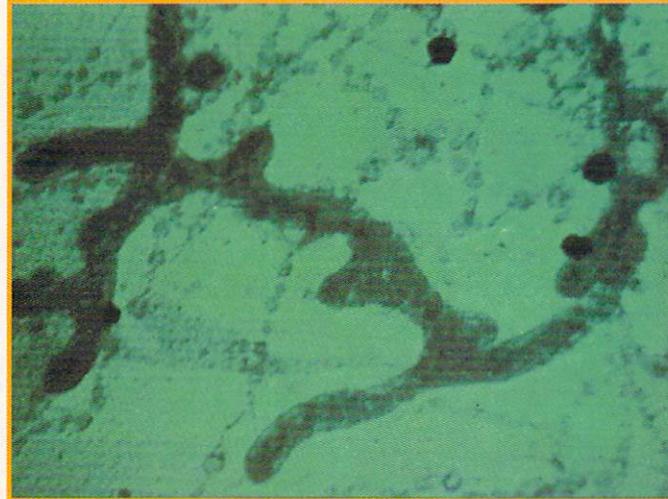
standby can lend dazzling new twists to your interactive presentations—

if you know the right techniques.

A program well known to Amiga artists is **The Director** (version 2, \$129.95, *Right Answers Group*). A number of prize-winning animations have been made with The Director, including Brad Schenck's "Sentinel" and "Charon," my own "RGB Hazard," and Brian Williams' "Walker II." You may be surprised to discover, though, that The Director is by no means designed just for animations—it is also a sophisticated tool for creating interactive multimedia applications. In fact, Commodore used this program to make the promotional demo for its CDTV project.

Like the original program, The Director 2 is script-based. Within its editor you create a set of written instructions that precisely control the sequence of events in an animation. The Director 2 turns this script into a "film" file that anyone can play by using the freely distributable "projector" player program. Although this may sound a bit complicated, it is not—especially if you know a few tips that make manipulating the program easy.

The ten techniques described below focus on interactive and audiovisual applications, and they should give you the inside



information that you need to make your projects come alive. Learning these tips may even give you the freedom to experiment with creative options on other multimedia programs. (To locate product vendors, see the "Manufacturers/Distributors' Addresses" list, p. 89.)

1. Take Control of Animations

Getting to know The Director 2's many powerful capabilities can help you create unique and interesting inter-

active projects or complex real-time animations. This program even draws praise from professionals, because it plays back animations quickly and chains together multiple animations seamlessly. You can play animations continuously or for a specified number of times, or you can ping-pong them forward and backward. The Director 2 lets you play your animations directly from a hard or floppy disk, and you can load them while other animations or sequences are running—without disturbing the flow of a presentation.

If you are creating an interactive project, use the BLOAD-ANIM command and plan the flow to take advantage of background loading. After all, when the user hits a screen button, ►

By Joel Hagen

something should happen immediately to give instant feedback. You could include, for example, a confirmation sequence in RAM that plays in response to button selection while the desired animation or image loads simultaneously in the background. While the animation loads, the user sees continuous action, rather than "dead air."

You can also make use of The Director's full frame-by-frame control of animations. At each frame during animation playback, you can vary the pause before the next frame, trigger sound effects, initiate other display events, or wait for a keyboard or mouse response from the user. Having this level of control makes an animation more useful—you can make parts of it play at different rates, or you can hold indefinitely on a title screen.

2. Take Control of Color

By specifying RGB values, you use COLOR, PALETTE, and CHROMA commands to build or modify colors "on



Figure 1. With The Director 2, you can create an interactive screen for a particular project (such as the Apollo mission animation above) from which images, text, and long disk-based sound samples can be called up.

the fly." Use these commands to draw attention to a selection or to direct a user through a series of items. By simply specifying new color values, you can give a single high-resolution background-texture image over 65,000 variations—all without using a lot of memory.

The Director's Chroma feature lets you modify the color bias of an entire palette or portion of a palette in a single stroke, making it possible to display a full-color image, then gradually bleed out the color to gray in real time, or bleed it back in. This technique gives you a subtle effect that is reminiscent of the opening to *Butch Cassidy and the Sundance Kid*.

The CYCLE and RANGE commands control color cycling, even during animation playback. You can create different ranges of colors at any time and cycle them in either direction. If you cycle a range a specified number of times, you experience the economy of the technique without the aesthetic drawback of continuous cycling. A brief moment of color cycling in a glint on a title, for example, can be a subtle touch.

With the FADETO command, you can fade an image to black or any color. By using FADEMODE, you can even elect to fade or alter the chroma of only part of the palette. This command opens up interesting options for unusual effects and transitions.

3. Use Formatted Text

FTEXT is a special feature that lets you read an entire page of text from disk and display it to the screen formatted with word wrap, drop shadows, right justification, margin changes, centering, and color, font, and style changes. Embedded symbols placed in the text file provide the format changes. This is an incredibly economical way to create impressive screen displays "on the fly" during an animation. You can even use text files as "cards" in HyperCard-like applications. By setting up references within the text files to other text, image, or sound files, you can intelligently manipulate on-disk data based on navigation through the data files themselves.

4. Explore Sound and SMUS Synchronization

Although The Director 2 can play sound effects and SMUS (Simple Music Score) files during your presentations, you can add flair to your projects with other unusual audio capabilities, too. You can play large sound samples directly from disk with the LOADDISK command, thus conserving precious chip RAM. This playback is independent of other display and sound events, so it preserves the pace and flow of the sequence.

Using this method is an excellent way to add a long segment of digitized speech or any other complex sound to a presentation. An interactive project on the Apollo program (see Figure 1), for example, could play long recorded quotes while associated images or animations appeared on the screen. For special effects, you could control volume and rate of play for these sounds during playback.

A powerful audio feature that is especially important to animators is The Director's ability to synchronize the display to notes in a SMUS file. With the SETSIGS command, for example, you can display alternating cartoon frames of a drummer each time a drum note ▶

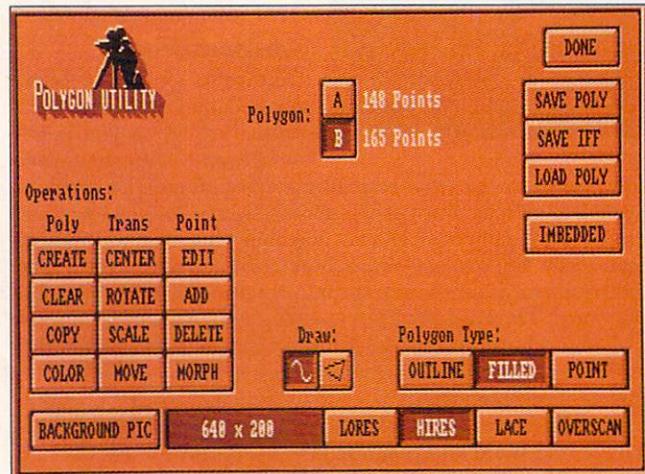


Figure 2. Use The Director 2's Polygon utility for creating polygons and movement paths.

sounds in a musical score. If you use this technique carefully, it can give the illusion of complex, nonperiodic animation. In an informational presentation, words or images could blast onto the screen in perfect time to a musical score, quarter note by quarter note. You can synchronize your animations to notes, measures, or even to tracks used just for timing and triggering display events.

5. Polygon Tricks

A special Polygon utility (see Figure 2) in The Director 2's menu lets you draw filled or unfilled polygons, then rotate, scale, and move them as part of a presentation. I like the Morph feature, which tweens one polygon into another. You can load a picture, trace a complex polygon, and then morph it into another shape you have drawn or traced. I have transformed one high-contrast face into another with this RAM-efficient technique. Stored as an IFF image, each two-color face

takes up about 6500 bytes on disk. Traced and stored as a polygon, each takes up about 550 bytes, a saving of over 90 percent.

Perhaps even more useful is the ability to draw a polygon and use it as a movement path for brushes, animbrushes, or sprites. You can draw the polygon, using a loaded image as reference if needed, and make the resultant movement path as complex as you desire. This is a great way to animate multiple paths over the same image. You could animate many routes of exploration in the ancient world over the same map, for example, using less disk and RAM space than if you were to create multiple animations.

6. Create Complex Buttons

Another useful feature in The Director 2's menu is the Button utility (see Figure 3), which makes it easy to create different kinds of buttons to use as active areas, or "hot spots," in interactive presentations. When users select these on-screen areas with the cursor, they can trigger any kind of audiovisual event. Buttons are not limited to simple rectangles or circles, either; they can be irregular and unconnected areas of any image. In a biology tutorial, for example, you could use an electron micrograph of cytoplasm as a button screen in which disconnected microsomes are treated as one button, mitochondria as another, and fatty granules as yet another (see Figure 4).

The trick to this kind of presentation involves making a second screen as a hidden mask (see Figure 5). The mitochondria on that screen are painted in a single solid color, microsomes in another color, and so on. When a user selects an area of the micrograph, the program checks the position of the cursor and correlates it with the color at that same position on the hidden-mask screen to determine what has been selected. Creating a hidden-mask screen lets you design sophisticated interactive applications.

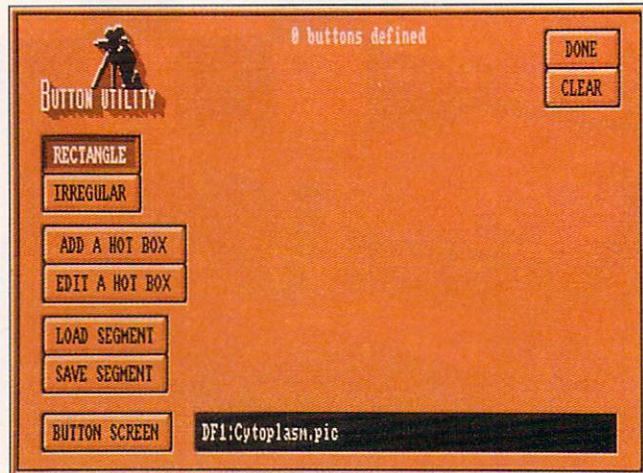


Figure 3. The Director 2's Button utility allows you to create interactive screen buttons, or "hot spots," for your presentation screens.

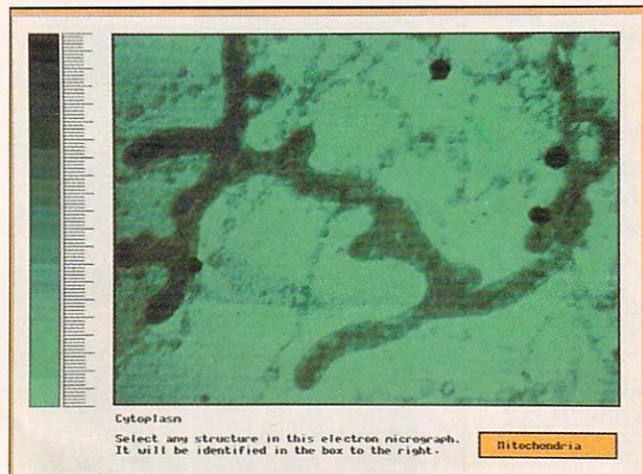


Figure 4. An electron micrograph button screen used in an interactive biology tutorial. Even disconnected areas such as the beaded microsomes can be treated as a single button.

7. Sprite Animation

The Amiga's hardware supports a special kind of graphics called "sprites." You can move sprite images around the screen at great speed without affecting the underlying image. The Amiga supports eight sprites, one of which is the mouse pointer. You can use The Director 2 to grab graphics information from a loaded image and convert it into sprite format to use in your application. You can even control sprites with a joystick. They do, however, have resolution, size, and color limitations. Sprite animation is not as simple as other approaches, but it is a powerful trick to have in your bag. If you are interested in game development, sprites are especially worth exploring.

8. Experiment with Interruption

The Director 2 lets you use the mouse, keyboard, or timer to interrupt your program for input and to transfer control to another section. You can do this in two fundamentally different ways. In the first method, the application reaches a certain point, then waits for the user to hit a key or press a mouse button. Then, the program switches to the appropriate sequence and continues. Instruction screens in our biology applica-

tion, for example, could include the sentence, "Hit any key to continue." The program could display each screen, then wait for the user to finish reading and press a key before moving on.

In the second method, the program does not need to wait for a particular point in its sequence of events to check for the mouse or keyboard event. The ONKEY and ONMOUSE commands recognize at any time that the keyboard or mouse has been used. With this method, for example, the user could return to the main menu of the biology application at any time.

ONTIMER uses the computer's timer to cause an interrupt based on the passage of any interval of time. In our first "hit any key to continue" scenario, you could add a timer interrupt to make the program move on to another sequence if no key were hit in 75 seconds. You can apply this technique to creating autorunning interactive presentations. The application invites the user to sit down and interact, but continues on a predetermined sequence if no action is taken.

9. Create Custom Transitions

Such transitions as The Director 2's wipes, fades, and dissolves are the mainstay of many Amiga applications. The Director 2 offers many additional capabilities, so that you can create custom transitions to give your presentation a fresh look. Two of my favorites are "animwipes" and custom-dissolve modes.

You create an animwipe by making a two-color animation (in Electronic Arts' DPaint, for example) and then using it in conjunction with The Director 2's Stencil feature to make a transition between two pictures. You could use DPaint's Move requester to create a simple animation of a star that starts as a small dot and spins in until it fills the screen. This could then become a mask between successive images in a slide-show-style presentation. Each new image could be revealed through the current image as a growing, spinning star.

This is a great way to create transitions that look like animated brushstrokes, drips, and so on. Because this technique may not be an obvious use of the Stencil fea-

ture, The Director 2 tutorial disk includes an example animwipe script.

You can control The Director 2's Dissolve style with the DISSMODE command. Instead of using single-pixel-sized cells to dissolve one image into another, you can define the width, height, and skew of the cell. This means that one command can dissolve either single pixels or large chunks of one image over another as a transition. I like to specify cells two pixels wide and ten high, skewed to the left, to create a dissolve effect that looks like a rainstorm.

10. Utilize Randomness

The use of random numbers in animations and applications is a particular obsession of mine. Controlled randomness can introduce an organic feel to any presentation. In the Dissolve-mode transition effect mentioned above, for example, I like to define width, height, and skew as random numbers. Each transition is unique, but I use only one simple command.

The Director 2 creates random numbers with the "?" symbol. The expression ?16, for example, generates random numbers between 0 and 15. This particular value is handy for changing the RGB levels of colors randomly, as I suggested in the color-control section. Where any value or location is specified in a presentation, you can make that number random.

One of the most interesting uses I have found for random numbers is to add surprise endings to animations. As mentioned earlier, one of The Director 2's strengths is its ability to seamlessly chain animations together with no distracting pause. This feature makes it possible to create a narrative animation as a group of animation files to be chained together by The Director 2 at the time of playback.

The trick is to create multiple endings as separate animations. Each ending animation proceeds from the last frame of the single beginning animation. Just before The Director 2 begins preloading or background loading the sections it chained together, the random-number generator "flips a coin" to see which ending it will use. You never know which ending will appear.

Variations on this idea are obvious. In interactive applications, you can introduce a great deal of variety into what otherwise would be predictable user feedback. Your program, for example, could select randomly from half a dozen different messages acknowledging a correct button choice. This enhances the feeling of interaction with a sophisticated application.

Perhaps these tips will give you ideas for enhancing your next Amiga project. Some techniques are incredibly easy to implement, while others are fairly sophisticated. Although many of these techniques rely on The Director 2's speed and memory efficiency, you may be able to modify them so that they work with other software products. Whatever software you use, if you are creative with its features, you have the key to producing unique animations and applications. ■

Joel Hagen, one of the founding members of the Right Answers Group, is the author of AW's "Accent on Graphics" column. His credits include work in art, astronomy, science fiction, and software development. Write to him c/o AmigaWorld Editorial, 80 Elm St., Peterborough, NH 03458.

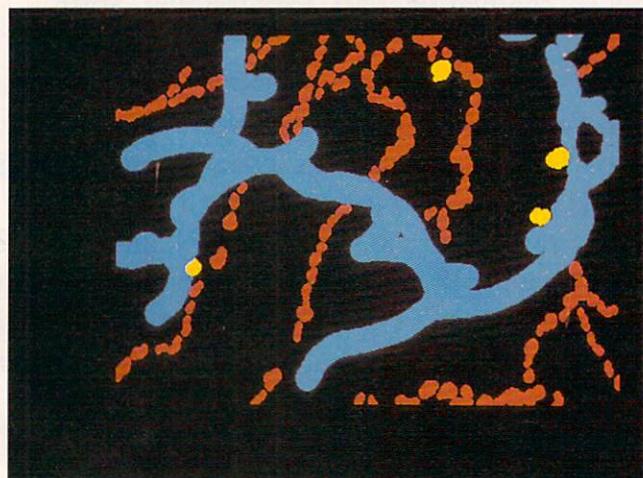


Figure 5. Here is the hidden screen that accompanies the micrograph image in Figure 4. Structures are delineated in different colors to act as complex buttons.