The Developer Business Report *February 1995*

Apple Directions

Inside This Issue

Editor's Note: The Shadow Doesn't Know	2
IndustryWatch: Life in a Low-Margin World	3
OpenDoc CD Ships, "Kitchens" Open to Selected Developers	8
Apple Announces Support for First Kaleida Products	8
GeoPort Telecom Adapter Kit Offers New Opportunities	9
Apple Joins Versit Computer and Communications Interoperability Initiative	10
E.T.O. and MPW Pro #16 Now Available	11
CD Highlights	11
Human Interface: Fad Gadgets	12
QuickTime 2.0 in the Limelight	14
OpenDoc Human Interface Q&A, Part Two	19
New Technical Notes Available on Paper	21
Marketing Feature: HyperCard Alive and Well	22

Apple News

Developers Announce Support for Power Macintosh With PCI

Twenty major hardware developers have announced plans to develop PCI (Peripheral Component Interconnect) products for Apple's second-generation Power Macintosh systems, due by mid-1995. The new Power Macintosh computers will incorporate the industry standard PCI bus, whose adoption is also increasing in the DOS/Windows world.

The PCI bus gives you several advantages over the NuBus™ technology used by current Macintosh systems. First, the PCI bus is more than three times as fast as NuBus. PCI cards designed for the Macintosh computer can also be used, with slight modifications, on computers running the DOS, Windows, OS/2, and UNIX® operating systems, expanding the market for your expansion cards. Also, Apple's PCI implementation will be compatible with the PCI 2.0 specification, which means that

please turn to page 8

Strategy Mosaic

Blueprint for the Future: Publishing

By Gregg Williams, Apple Directions *staff*

It's More Than Just the Printed Word

Everybody knows it's easier—and far more efficient—to defend what you have than it is to capture what you don't have. The wisdom of maintaining strategic markets is not lost on Apple Computer, Inc., so Apple senior management has identified four markets in which Apple excels:

- publishing
- communications and collaboration
- learning
- technical

Management has established four Program Offices that report directly to Apple CEO Michael Spindler to ensure that Apple retains and advances its leadership in these markets. Each office is responsible for analyzing its market, formulating a worldwide strategy for Apple, and coordinating all efforts within Apple to implement that strategy.

Part of *Apple Directions*' charter is to give you the information you need for determining how your company should best make use of Apple technologies and products. In the

please turn to page 4

AppleDirections

Volume 3, Number 2

Apple Directions, the monthly developer newsletter of Apple Computer, Inc., communicates Apple's strategic, business, and technical directions to decision makers at development companies to help maximize their development dollar. It is published by the Apple Developer Periodicals group within Apple's Developer Press.

Editor
Paul Dreyfus (AppleLink: DREYFUS.P)

Technical Editor
Gregg Williams (GREGGW)

Business & Marketing Editor
Kris Newby (NEWBY.K)

Associate Editor
Anne Szabla

Production Editor
Lisa Ferdinandsen (LISAFERD)

Contributors
Peter Bickford, Dave Curbow, Alex Dosher, Elizabeth Dykstra-Erickson, David Gleason, Kris Newby

Manager, Developer Press
Dennis Matthews

Manager, Apple Developer Periodicals
Greg Joswiak

Production Manager
Diane Wilcox

Prepress/Film
Aptos Post

Printer
Wolfer Printing Co., Inc., Los Angeles, CA

Editor's Note

The Shadow Doesn't Know

Jungian analysts are fond of referring to such entities as our "Shadow Self." This is the little stinker inside us who, while also giving us our creative edge, makes us laugh at our fellow people when they are down and do other similarly awful things.

I'm glad to report that my Shadow Self is alive and well; the glee he experienced reading about Intel's Pentium floating-point fiasco probably provided me enough creative edge to get me through the next year's worth of *Apple Directions.*

Sadly, my Voice of Reason also proved itself a strong force. (The Voice of Reason is a part of ourselves dismissed too easily by the analysts, probably because it's just no fun.) The more joyful I became at the prospective, if improbable, demise of the Pentium chip, the more the Voice began to paraphrase the Bible with such sayings as "Laugh not lest ye be laughed at" and "Why do you point out the bug in someone else's product when you have a virus in your own."

I'd hope, now that Intel has agreed to replace the flawed chips and the amusement has died down, that we can all draw a rather sobering conclusion from the Pentium fuss: that we all need to focus our efforts *at least* as much on our customers' needs and wants as we do on our own if we want to avoid similar debacles.

I wrote that last sentence very carefully. Notice that I didn't say "Thou shalt always, 1000 percent of the time, put your customers' needs first," or some other impossible-to-follow commandment. We'd all like to think we put our customers' needs first all the time, and we'd probably advise others that they're dead if they don't do the same thing. But let's face it: It just doesn't happen that way.

We face compromises all the time in this business because we're trying to deliver a highly complex product and make it appear to be very simple. Here are the top two:

• *Compromise # 1: our need to deliver the next great technology regardless of how useful it is versus our customers' need to actually do something with it.* Even products with the most customer-centered design include features that exist just because the engineers liked them. (Along these lines, note that this column begins with a few sentences in which my desire to write something clever overcame your need to read something useful.)

• *Compromise #2: the necessity of delivering flawed products versus our customers' expectations (and our hopes) that computer hardware and software will perform flawlessly.* Given that there is virtually no such thing as bugless software or chips and that increased complexity and power tend to result in more bugs, this is always going to be the case.

It was in making Compromise #2 too much in its own favor, and not enough in its customers', that Intel ran aground. I think that this is by far the more dangerous of the compromises: Even though customers may be confused by technology that's not immediately useful to them, they can also find it intriguing and appealing. But few customers are intrigued to find that their 100-MHz system can't get math right.

When we make Compromise #2, we have to be especially careful not to underestimate the customer. Maybe it's time to stop spreading Pentium humor (fun as it is; for example, did you hear that Intel stock just split 1 for 1.999543? Oops, there goes my Shadow Self again) and, instead, reread Tom Peters's *In Search of Excellence* and *A Passion for Excellence* as well as other books that tell us how to put our customers' needs up front.

Paul Dreyfus
Editor

IndustryWatch: News & Perspective

Life in a Low-Margin World

Prepared by the Apple Directions *staff*

Editor's Note: IndustryWatch is our regular compilation of news about events happening outside the Apple Computer, Inc., R&D complex and the ways we think they affect your development efforts. Each month, we gather the most notable items, which we hope will point you toward new opportunities, help you avoid mistakes, and alert you to key developments in the industry. We're not trying to cover everything in the computer industry; other publications already do that. Instead, we're digging through the news to present you with the most interesting tidbits, ones that translate into ideas you can use. If you have comments about IndustryWatch, or it you want to alert us to issues you think the Apple development community needs to know about, send us an AppleLink message at A.DIRECTIONS.

PC Multimedia Woes: An Opportunity and a Challenge

Despite the reluctance of vendors in the DOS/Windows world to disclose problems their customers are having getting multimedia to work on PCs, some numbers on the extent of the problems are surfacing. According to the September 12, 1994, issue of *Electronic Engineering Times,* the return rate for the Windows version of Myst is 40 percent (compared with only 5 percent for the original Macintosh version). Also, the October 24 issue of *Computer Retail Week* quotes Richard Buchanan, director of product marketing for Creative Labs, as saying that return rates for PC multimedia upgrade kits were "as high as 25 percent" last Christmas. With multimedia upgrade kit sales growing, we expect that the return problem will receive increasing attention in the press.

Implications/Opinions: Clearly, the problems customers encounter when trying to do multimedia on DOS/Windows-based PCs poses an opportunity for the Macintosh computer, but it also presents a challenge. We all know that running multimedia software on the Macintosh is easy, while DOS/Windows-based multimedia can be a very difficult experience for customers. Lots of customers, though, don't know that, so to avoid being tarred by the same brush that's blackening the PC's multimedia reputation, we all have to get busy communicating the Macintosh multimedia advantage.

Apple has been running a series of television ads touting the ease of Macintosh multimedia; you can help by being sure that your advertising, product packaging, and other promotional materials let customers know that your product (a) runs on a Macintosh computer that (b) provides them with true plug-and-play multimedia capability. To arm you with information about the Macintosh multimedia advantage, we've posted on AppleLink a report compiled by Apple's Competitive Analysis group, "Using CD-ROM Titles on Macintosh vs. IBM-Compatible PCs." To find it, follow the path Developer Support:Developer Services:Periodicals:Apple Directions:Apple Directions February 1995.

Hayes in Chapter 11—Lessons You Can Learn

Modem manufacturer Hayes Microcomputer Products became the latest high tech firm to face difficulty because of the commoditization of PC peripheral products. Hayes found it difficult to compete in a high-volume, low-margin business where nearly anyone can build a state-of-the-art modem by simply putting a modem chip on a board. Ironically, the final straw pushing it to file for Chapter 11 was the company's inability to manufacture sufficient units to keep up with overwhelming customer demand.

Implications/Opinions: Hindsight is 20/20, and it's easy to point out that appropriate forecasting and sufficient inventory would have helped Hayes. But who among us can claim 100 percent forecasting accuracy?

Less obviously, Hayes apparently fell victim to the trend toward PC peripheral solutions being provided by chips rather than at the card level. Former PC graphics-board leaders such as Video Seven and Paradise earlier succumbed to this same trend. Card developers take note: To stay ahead of the curve, you'll want to invest in chip engineering. If you decide not to develop your own chips, you'll want to work with a chip maker (such as S3 or Trident) to make sure your products take advantage of the latest chip technology.

U.S. Federal Program Provides Grants for Technology Development

Uncle Sam may want to help you develop next-generation technology. Under its advanced technology program (or ATP), the U.S. Commerce Department has announced it will spend as much as $1.5 billion over the next six years on grants to develop a variety of new commercial technologies. Among those technologies are improved data storage systems and new methods of sending video images over computer and telecommunications networks. The Commerce Department has specific, ambitious goals for new technology developers; for example, one of its goals is a 40-percent annual increase in the amount of information that can be stored on hard disk or optical disk drives. The ATP will match U.S. firms' investments in the new technologies.

Apple Directions Online—March

Apple Directions Now on Internet and eWorld

Starting this month, you can read *Apple Directions* on the Internet's World Wide Web. It's available as a WWW page at the location www.apple.com; when you get there, select Developer Services and Products.

You can also find *Apple Directions* on eWorld in the Apple area of the Computer Center.

As usual, we post final *Apple Directions* files on AppleLink (path—Developer Support:Developer Services:Periodicals:Apple Directions).

The March issue of *Apple Directions* will be available at all three locations by February 15.

Implications/Opinions: Like every government spending program, the ATP will probably be strongly challenged by the new Republican Congress. However, the ATP was started in the Bush administration over the objection of budget officials, and with strong support from both the administration and Congress, it's become one of the fastest growing U.S. government programs. If you'd like to find out more about it and see whether it's worth your while to apply for an ATP grant, call the ATP in Washington, DC, at 301-975-2636 and ask to be put on its mailing list.

Groupware Alive and Well

Novell, Inc., recently announced it would work with Collabra Software, Inc., to sell the small network developer's groupware products and integrate them into the Novell product line. The alliance is taking aim at the market leader for collaborative software, Lotus Development, whose Notes product is used by nearly 1 million customers worldwide.

Implications/Opinions: You can expect to read a lot more about groupware in the coming months, as Novell begins to ship Collabra's products in early 1995 and as Microsoft releases its long-delayed Exchange Server sometime later in the year. As we've said before, if the "big guys" are spending their R&D and marketing resources in a product area, it's only because they're convinced that there's customer demand for solutions in that area. Collaborative computing has, for some time now, been one of the next hot areas for commercial development. If you have experience developing custom, in-house collaborative solutions, you have a leg up in this somewhat complex area of computing. And, if you're considering ways of capitalizing on that experience, don't forget that System 7.5, the latest release of the Mac OS, is the only currently available mainstream operating system with system-level collaborative capabilities. To find out more about AOCE (Apple Open Collaboration Environment), the technology that makes that possible—and that also enables your own collaboration solutions—see the November 1993 issue of *Apple Directions* (which can be found on the December 1994 Developer CD, path—Periodicals:Apple Directions:Apple Directions 1993:Apple Directions 11/93.)

Addendum: For Pentium-based PC Owners Only

Want to turn up the bug on your Pentium-based PC? Here's how: Using Microsoft Excel, enter 4,195,835 and divide it by 3,145,727; next, multiply the result by 3,145,727. The result should, of course, be 4,195,835, but on a machine using the buggy Pentium the result will be off by 256. In any case, you may want to take advantage of Intel's offer to replace your Pentium chip for free, an offer Intel made after weeks of wrangling and press-bashing. (Also, did you know that the calculator that runs on the Windows system makes math errors, as well? When the user subtracts 2.00 from 2.01 on a PC running Windows, the machine gives an answer of 0.00. Apparently, it has difficulty with other numbers ending in .01, too. After Walter Mossberg reported the Windows calculator's flaw in his Personal Technology column in the *Wall Street Journal,* Microsoft announced it would soon release an accurate version of the calculator.)

Implications/Opinions: Perhaps we're preaching to the converted, but this is just Why to Develop for Macintosh Reason #132.0000. The Pentium controversy should also remind us of how crucial it is to keep our customers' interests first and foremost in our minds and business plans. (See this month's Editor's Note for more on the last point.) ♣

Strategy Mosaic

Blueprint for the Future

continued from page 1

coming months, I'll be writing about each of these four markets, beginning with publishing.

What Is Publishing?

It's easy to think of publishing as *desktop publishing*—that is, the ability to use styled text, graphics, and layout elements to create printed output. That was my assumption, until I talked to Lisa Wellman, director of the Apple Publishing Program Office. She made the very good point that desktop publishing is not a differentiating technology—today, it's baseline technology that users expect to see on any computer, and any product that doesn't supply it isn't considered usable.

Apple's vision of publishing is simply stated: *packaging and communicating information—any media, anytime.* This is a long-term vision and, as such, it points to a state that has not yet been achieved. Still, we can see the beginnings of this broader definition of publishing. Multimedia software (assisted by QuickTime for both the Macintosh and Windows platforms) certainly constitutes publishing, as do the many forms of content (including multimedia) now published on CD-ROM. People are beginning to use Macintosh computers—especially the 630 and AV series computers—to create presentations on videotape. Every online service has its own forms of publishing; Apple's eWorld, for example, includes such diverse areas as Grolier's Encyclopedia, Fodor's Worldview (travel), Eaasy Sabre (airline reservations), Hoover's Company Profiles, INC. Online (business), Ad Age/Creativity Online, Nolo Press, and online content from *InfoWorld, MacUser, MacWEEK,* and *Macworld.*

As you will see, Apple's efforts toward strengthening its publishing platform concentrate on the business and corporate communications markets. However, many of Apple's efforts to advance publishing will benefit other Macintosh users—including the small business, home, and education markets—because those advances will help people work more easily with images, video, and sound, and will help people communicate more effectively.

Publishing Program Office Mission

The Apple Publishing Program Office has objectives that are focused toward both users and developers. To users and the market in general, its objectives are

- to align Apple's publishing efforts to a single, easily understood vision
- to differentiate Apple publishing technologies, resulting in publishing solutions that are clearly perceived to be superior to competing platforms

Apple recognizes that its success in publishing depends on the quality of its partnerships with developers. Here, the office's objectives are

- to ensure a strong business proposition for third-party developers to supply key parts of superior Apple-based publishing solutions
- to develop and present creative, profitable opportunities for the specialized resellers, VARs, consultants, and systems

integrators who serve the publishing market

Not only will realizing these objectives allow developers to make a living from supporting Apple technologies, but it will also bring new customers to the Macintosh platform (which will be good for Apple, developers, and customers).

Apple's Global Efforts

The above goals cannot be accomplished solely by internal efforts. Apple has been working throughout the world to promote Apple technologies and to demonstrate leadership in the field of publishing. Apple has sponsored a number of events and ongoing efforts:

- the Worldwide Publishing Consortium
- technical briefings in Japan and Australia
- the Seybold Conference in Paris last June
- the Japanese Publishing Consortium
- the Apple Publishing Center in Beijing, China
- the upcoming DRUPA commercial publishing convention, a mammoth 15-day event to be held next May in Dusseldorf, Germany

The Worldwide Publishing Consortium is a vendor-neutral consortium for advancing the state of computer-based publishing. See the text box on this page for details.

Important Directions in Publishing

The Publishing Program Office has identified four key directions that it believes Apple must pursue to continue its leadership of the publishing sector. Some are obvious, some not so—but to maintain a leadership role, you have to choose the right technologies and promote them *before* they become well established. Following are descriptions of the four directions that Apple will be pursuing vigorously.

Color. You don't have to be an art director to realize that color is here to stay. For users, the Macintosh platform—both hardware and software—is the best platform for displaying and printing color. Furthermore, the Macintosh is the platform of choice for high-quality color publishing. According to Wellman, "Virtually all the high-end four-color work by the professional market originates on Macintosh equipment." For developers, the Macintosh platform is the easiest and the most powerful one for color-intensive applications. Apple intends to continue enhancing the Macintosh platform so that both you and Macintosh users will continue to enjoy the easiest and most sophisticated use of color on any personal-computer platform.

Cross-media. This term refers to the ability to use different media—printed reports, collateral materials (brochures, data sheets), videos, interactive CDs, and more—to communicate your message efficiently and effectively. Today and even more so in the future, advantages will accrue to companies that can take one finished piece and quickly adapt it to other published forms meant to reach different customers. This is especially true for corporations. According to Wellman, "Apple will help corporations communicate who they are and what they offer. In other words, what is the experience your customers have with your product? In the 1990s, I see corporations differentiating themselves more that way than through their actual product. Corporations need to reach customers wherever and whenever they can with information in the most appropriate form. This includes things you may not have thought of—interactive point-of-purchase displays, for example, or videos mailed to the home. Corporations can do such things on Macintosh computers easier, quicker, and with higher quality than they can on competing platforms."

Collaboration and communication. Publishing, in almost all its forms, is a shared task. Any number of people—writers, artists, content experts, graphic designers, copy editors, managers, even sound and film professionals—must be involved with an item to be published.

WWPC—Participating in the Future of Publishing

By establishing the Worldwide Publishing Consortium (WWPC), founding sponsors Apple, Radius, and Kodak (among others) are in effect saying, "We know publishing is a complex process. We need to stay in touch with the people who will be using publishing products, and the WWPC is the way we're going to do it."

The Worldwide Publishing Consortium offers a vendor-neutral forum on the future of publishing, a source of information on major vendors' views of the industry, and chances to communicate (both electronically and in person) with others in the publishing field.

According to Alice Locke-Chezar, managing director of the WWPC, the organization's objectives are

- to facilitate the implementation of technology into the process of publishing (which the WWPC defines as "the turning of information into communication," a broad definition of publishing that is similar to Apple's)
- to provide a forum in which individuals representing the publishing, graphics arts, communications, and multimedia industries can exchange and share knowledge and experience
- to influence the direction of development activities, products, policies, and services of all vendors in the interest of all members
- to educate members and the public in the advancement, improvements, and uses of technology in communicating information through meetings, forums, conferences, special interest groups, and newsletters
- to acquire, preserve, and disseminate data and valuable information relative to the functions and accomplishments of the membership and associated industries
- to cooperate with local, regional, national, and international counterpart groups

Individual memberships in the Worldwide Publishing Consortium are $200 per year, and institutional memberships are $500 (which includes membership for three people). In addition, for a membership fee that is proportional to your company's sales, you can become a vendor sponsor of the WWPC, which entitles you to take a more active role in setting the direction of the WWPC.

For more information, send e-mail to AppleLink address WWPC or Internet address wwpc@applelink.apple.com, or call 800-865-9972 or 408-354-3966.

Therefore, tools that allow these people to work together—regardless of their geographical location—will become increasingly important. As companies continue to be concerned with their efficiency and competitiveness, they will greatly favor products and technologies that facilitate the publishing process by enabling the people involved to work together better.

CDDI. Short for *corporate direct digital imaging,* CDDI refers to the ability of corporate users to publish (in the broad sense of the term) from their desktops. This is the trend that is furthest from fruition, but Apple believes that it is a very important trend for the corporate publishing market.

CDDI—All-Digital, All the Time

The ultimate expression of CDDI will be the corporation that does everything—up to the actual moment of printing—electronically, never generating a physical object that is photographed and then reproduced. (By eliminating the manual labor associated with pasting up and photographing a "mechanical," you automatically eliminate one of the most time-consuming parts of the publishing process.) Using CDDI to its full extent (which I'll call "complete CDDI") delivers several important benefits:

- much faster turn-around time internally and quicker time-to-market
- potentially greater control of the project
- the ability to target submarkets (or even individual clients) by customizing your deliverables to the intended recipient
- savings to the environment and the company's bottom line (companies can reduce both the cost and environmental impact of publishing by reducing or eliminating the role of conventional photography, which uses chemicals).

Granted, complete CDDI doesn't and won't make sense for every company, but its adoption is still important because it will lead to two benefits for everybody:

- In the path from concept to final deliverable, more parts will be digital (and the conventional wisdom in publishing is "Get your content digital as early as possible and keep it digital as long as possible"). The more of your content that is digital, the more you can do with it and the more you can reuse it.
- As more digital solutions become available, you are more likely to have multiple ways of getting your content published. This gives you a flexibility (usually involving a tradeoff of money and time) that you don't have when there is only one way of accomplishing a task.

The industry has already come a great distance toward complete CDDI. Desktop computers have become a commodity in many businesses, and portable computers and Newton devices make workers more likely to capture data in the field and bring it back to the desktop in digital form. Scanners and digital cameras (such as Apple's QuickTake 100 digital camera) are gaining acceptance, especially for things like real estate and automobile ads in newspapers.

QuickTime makes it possible to add video to applications for both the Macintosh and Windows platforms. Low-cost CD mastering and recordable CD-ROM drives make the CD a new medium for publishing, as well as a medium for distributing color images, video, and animation. In addition, networks, online services, and computers themselves have increased the amount of raw material that is available in digital form.

Apple believes that the personal-computer industry can serve business—and, in particular, corporations—in ways that it is currently unable to do. This implies significant business opportunities for companies willing to pursue them. Apple will continue to supply the hardware, the system software, and the vision that will make the Macintosh platform the best one for delivering cutting-edge publishing solutions; the rest is up to you.

Things to Think About

Considering that Apple's definition of publishing includes media such as print, video, CDs, and online services, it is difficult to make a meaningful list of specific recommendations. However, I have come up with a laundry list of things to think about.

WWPC. First, send off for information on joining the Worldwide Publishing Consortium (see the text box on page 5). Even if belonging to the WWPC gives you only one insight or one contact in the industry, it still will be worth the cost of membership.

Key technologies. If you haven't already, support the following key technologies (as appropriate for your products):

- Power Macintosh (publishing applications benefit greatly from the extra speed that Power Macintosh computers provide)
- AppleScript and scriptable/recordable applications (to make it possible for others to use your application as part of a scripted custom solution)
- PowerTalk (so that your application can communicate more easily with others using the PowerTalk mailer)
- QuickTime 2.0 (for multimedia and the playback of video within documents)
- Apple Guide (to give customers a better experience using your application)
- QuickDraw GX (to enable your application to do more with graphics and type than is possible using QuickDraw; also, every application that does printing should support QuickDraw GX printing)
- Macintosh Drag and Drop (to give users a way of moving data between documents that is more effortless and intuitive than the Clipboard)

Publishing applications are more demanding of a computer platform than almost any other class of software. By adopting the above technologies, your application—which exists in a very competitive environment—will be able to offer more to potential customers.

Peeking over the horizon is another key technology, OpenDoc. (In fact, Apple Partners and Associates and readers of *MacTech Journal* already have an OpenDoc CD in hand and can start doing serious development.) OpenDoc is a key technology for mixing different kinds of content in the same document, and it will also make it possible for you to update your application quickly—both of which are increasingly important in the publishing industry. It's significant to note that Adobe is one of the major sponsors of the Component Integration Laboratories (the owner of the OpenDoc technology) and that it has announced plans to create parts that will allow content from Adobe Illustrator, Photoshop, and Premiere to appear in OpenDoc-based documents.

Looking at Corporations

Apple expects corporate publishing to become increasingly important in the years ahead, and looking at the needs of corporations may give you some ideas for new business opportunities.

Work flow is a key issue for corporate publishing. Teams of people working on the same project may be physically separate, working in different departments or scattered across the world. These teams need mechanisms to help them track content,

coordinate and share their work, and manage the project's overall flow.

Today, numerous products handle single tasks within a project, but they were designed for individual use and have no "knowledge" of (and offer no help in accomplishing) the overall project. The compelling publishing solutions of the future will act within the context of the entire project.

Crafting solutions that deal with the work flow of large projects will be difficult, but people will pay well for a solution that works. Your solutions must confront the difficult issues surrounding group dynamics and corporate policies. For example, users are more likely to embrace solutions that allow them to continue working the way they're used to. They're not going to change their ways unless the new solution does something for them (and they definitely won't use anything that is more work for them but benefits only upper management).

Also, your application may have to address the conflict between the individual's desire for access (to, say, artwork for the company logo) and the corporation's need for control and the enforcement of procedures. How do you decide what to do? You talk to your potential customers and find out how they work and what they need.

New Product Ideas

Here are some things you should consider doing:

- Look at your current applications and figure out where their "boundaries" are—that is, where they start and stop being helpful within the overall publishing process. Then ask yourself some questions. What else, currently outside the scope of your applications, might they do in the context of the entire publishing process? What do they have to know about the overall process to be of use?
- Bundles may not be made in heaven, but they're increasingly made in the boardrooms of software companies all over the world. You should consider bundling your applications with other software that handles other parts of the publishing process. Bundles are popular because customers don't want to have to figure out what combination of products to buy.
- Your bundled solution will be more compelling if the products involved work in concert with each other. One way to make that happen (for a modest investment of your resources) is to write AppleScript scripts to integrate the products; you may want to use FaceSpan™ (included in the AppleScript Software Development Toolkit) or HyperCard 2.2 to create an easy-to-use human interface for your scripts.
- With a bit more work, you can turn a suite of loosely integrated applications into a major new product. On the PC-compatible side, Lotus Corporation has done this quite successfully, using macros and utilities to tie together a spreadsheet (1-2-3), a word processor (Ami Pro), a graphics program (Freelance Graphics), and several other programs to make its Smart Suite product. Smart Suite is noteworthy because it offers the only significant competition to Microsoft Office—and it's holding its own against Microsoft. It's possible that you could turn the right combination of products (including yours) into a very successful publishing solution.
- One thing that corporations (and many other users) want is training. So you may find a profitable market in selling training materials. To date, computer training materials have focused on single-product tutorials. However, since most publishing efforts require the use of multiple products, why not have training materials that show how to get the job done with those products? For example, you can use FileMaker Pro and QuarkXPress (along with AppleScript) to automate the layout of office-supply and other kinds of catalogs. A product that shows people how to do that sounds like a good idea to me.

It's the Content, Stupid

You can't publish without having content, so this area should offer at least a few business opportunities, including the following ones.

It's here somewhere. . . . Sound, images, and video are relatively new data types, and customers everywhere will want products that help them store, catalog, search for, and manipulate them. In addition, high-end solutions should address the problems of archiving and of media stored in more than one location (perhaps even on unconnected networks).

Talent in a box. Many customers have more money than time (or creativity), so there'll always be a market for affordable collections of "clip" art, sounds, video, photographs, animations. (The Software Publishers Association recently reported that the sales of clip art in the first half of 1994 increased 68 percent over the same period in 1993.) Much of the hassle of creating multimedia projects is making sure you have the right to use a piece of content; because of this, you should be sure that whatever "clip media" you produce includes the appropriate level of legal right-to-use for the intended audience.

Partners in commerce. Many companies have tons of content or data but aren't computer-savvy. Partner with them and modify their content (which is usually in printed or database form) to do something useful in another form. The target media available include CD-ROM discs, online services, and handheld devices such as the Newton MessagePad. Also consider opportunities for bringing relevant data to specific markets (for example, teachers, students, health-care professionals, or salespeople).

Think Big

Publishing is big business, and it will force you to think big. Some products will be for use with large groups of people; they'll generate big files and will require fast Macintosh computers with large amounts of memory and big hard disks. The most compelling solutions will probably involve a large development effort, but you'll make big profits from them—corporations, when they buy, purchase large numbers of units at the same time.

Even if your company doesn't see any opportunities for itself in publishing, it's important to know that Apple has devoted considerable thought to improving its position in the personal-computer market and providing you with healthy markets for your Macintosh and Newton products. Publishing is one of Apple's most visible markets, and Apple intends to continue its leadership and preeminence in it. ♣

Apple News

Developers Announce Support

continued from page 1

any existing 2.0-compliant PCI card with the appropriate software driver will work in PCI-based Power Macintosh systems. Further, PCI cards can be cheaper to manufacture if they incorporate industry-standard PCI-compatible chips. For more details about PCI, see "PCI Cards for the Second Generation of Power Macintosh" in the August 1994 issue of *Apple Directions.*

The developers who announced their support of PCI design and manufacture cards that support virtually all major solutions provided by hardware add-ons, including digital video production, communications, client-server computing, enterprise computing, data acquisition, and networking. PCI cards from these vendors are intended to support a full range of peripherals, including SCSI devices, mass storage and tape backup products, printers, scanners, processor accelerators, and NuBus expansion chassis.

Apple is also encouraging the development of PCI cards to provide new peripheral-based solutions for the next-generation Power Macintosh computers. These solutions include MPEG (Moving Pictures Experts Group) and JPEG (Joint Photographic Experts Group) formats to facilitate full-motion video on the screen by compressing video images and high-speed networking, such as ATM (Asynchronous Transmission Mode) and FDDI (Fiber Distributed Data Interface).

The following is a list of the developers who have announced plans to provide products that support the PCI architecture:

- 4-Sight (International) Ltd.
- Adaptec, Inc.
- Asanté Technologies, Inc.
- ATI Technologies, Inc.
- ATTO Technology, Inc.
- Avid Technology, Inc.
- DCA
- Diamond Multimedia Systems, Inc.
- Digital Equipment Corp.
- Farallon Computing
- FAST Multimedia AG
- FWB, Inc.
- Harlequin Ltd.
- Hermstedt GmbH
- Intelligent Resources Integrated Systems
- Linotype-Hell AG
- Neutral Ltd.
- QLogic Corp.
- Radius, Inc.
- Second Wave, Inc.

For details on each company's plans, check the version of this news item on AppleLink, using the following path—Developer Support:Developer Services:Periodicals:Apple Directions:Apple Directions January 1995.

OpenDoc CD Ships, "Kitchens" Open to Selected Developers

As promised, Apple Computer delivered an OpenDoc CD to developers by the end of 1994! (Well . . . just barely, and it isn't the long-awaited beta, but c'mon, it's *software.*) Apple shipped the OpenDoc with SOM Developer Release CD to all Apple Associates and Partners in the January 1995 monthly mailing (which was mailed in mid-December 1994). Just to make sure that *everybody* who wants OpenDoc can get access to it, Apple arranged with Xplain Corporation to bind this same CD into the January 1995 *MacTech Magazine,* which was handed out at Macworld Expo San Francisco earlier this month and which should still be on the newsstands.

Besides being the first wide distribution of OpenDoc to the developer community, the OpenDoc with SOM Developer Release is important because it's the first release to use the System Object Model. SOM, a piece of technology that IBM contributed to OpenDoc, provides a robust, cross-platform way of building, packaging, and manipulating object-oriented class libraries in binary form.

Though this release is not of beta quality, it's definitely good enough to use for getting up to speed with OpenDoc—we know, because several dozen developers have already used it. On November 9–11, 1994, in Kista, Sweden, and again on November 28–30, in Watsonville, California (epicenter of the big earthquake in 1989), Apple hosted two different OpenDoc "kitchens." Several dozen developers attended and used essentially the same version of OpenDoc as is on the OpenDoc with SOM Developer Release CD to create OpenDoc parts.

The results? In the Watsonville OpenDoc kitchen, "Kamp Kodabunch," 14 developers showed partially or fully working OpenDoc parts after fewer than three days of work. (Many developers began with the source code of existing applications they had written, and adapted that code to the OpenDoc world. This proves that it's quite feasible for you to turn existing applications into OpenDoc parts.)

Apple is hosting OpenDoc kitchens each month, so that as many developers as possible will become productive with OpenDoc (and so that finished commercial OpenDoc parts will be available by the time OpenDoc ships later this year). Since members of the OpenDoc team attend these kitchens, attending one is the best way to learn OpenDoc. If you're interested in attending an upcoming kitchen, send your request to AppleLink address OPENDOC or Internet address opendoc@applelink.apple.com, along with information on your company and what you'd like to do with OpenDoc.

The next release of OpenDoc, soon to be available, will be a true beta release and will include an early release of the OpenDoc Parts Framework, which speeds up the process of developing OpenDoc parts for both the Macintosh and Windows platforms. (For more details, see "OpenDoc *Is* Cross-Platform" in the November 1994 issue of *Apple Directions.*)

Apple Announces Support for First Kaleida Products

You might remember that two years ago, as part of their strategic partnership, Apple Computer, Inc., and IBM Corporation cofounded Kaleida Labs to develop a platform-independent multimedia standard. Recently, Kaleida shipped the first fruits of its labors, the Kaleida Media Player and the ScriptX Language Kit. Apple, along with IBM, immediately announced its support for the products.

The Kaleida announcement opens opportunities for you to write ScriptX-based applications that will run on the Kaleida Media Player. It also provides opportunities to write ScriptX-based authoring tools so that customers can create their own multimedia materials.

The Kaleida Media Player (KMP) is software that works with

a variety of operating systems to run CD-ROM–based multimedia applications written using ScriptX. An application written using ScriptX will execute with the KMP regardless of the platform it was created on or the computer it's running on. Currently, ScriptX and the KMP are available only for the Mac OS and the Windows operating system; OS/2 Warp versions will ship in the second quarter of this year, and Kaleida plans a UNIX version of the KMP.

This translates into an unprecedented programming advantage for multimedia developers, since you'll be able to write a single version of your program using ScriptX on the computer of your choice; your customers will then be able to run it with the KMP on the computer of their choice. You'll no longer have to spend the resources to develop and support multiple versions of your CD-ROM–based applications.

Because of Apple's and IBM's immediate support of Kaleida's technology, it's likely that there will soon be a significant market of KMP-equipped computers to run your ScriptX applications. In the words of David Nagel, Apple senior vice president and general manager of the company's AppleSoft division, "Not only do we see the Kaleida Media Player becoming a popular feature on future Apple systems, we also intend to support the Kaleida software platform with ScriptX tools." Similarly, IBM has announced that it, too, intends to include the KMP with future IBM systems.

Needless to say, the KMP is at the heart of Kaleida's product offering and vision; it's the element that customers will see when they run ScriptX-based applications. Just as crucial is the ScriptX Language Kit, which consists of the ScriptX language and class library—more than 250 preprogrammed core classes as well as browsers and a debugger. The ScriptX language is object oriented, providing encapsulation, multiple inheritance, and polymorphism.

Since it is a programming language—and not an authoring tool, like some other current software products that are used to create multimedia titles—ScriptX is best used by experienced programmers. Kaleida is encouraging other companies to provide a variety of authoring tools that will give customers the ability to create their own ScriptX-based titles. It's expected that Apple will be among the first companies to release ScriptX authoring tools.

The ScriptX Language Kit is available directly from Kaleida (phone: 800-6-KALEIDA or 415-335-2098; fax: 415-335-2097). You can license the rights to distribute the Kaleida Media Player with your products. Kaleida also offers developer membership programs to provide technical support; you can obtain more information about the programs on the Internet World Wide Web at http://www.kaleida.com/ or by contacting Kaleida at the numbers just mentioned.

GeoPort Telecom Adapter Kit Offers New Opportunities

At Macworld San Francisco earlier this month, Apple Computer, Inc., announced the GeoPort Telecom Adapter Kit. To users, this kit provides an all-in-one personal communications solution that makes a user's Macintosh computer into a telephone (with full-duplex speaker phone capabilities), a single source of voice messages, fax, and e-mail, and a high-speed (14,400-bps) send and receive fax modem. To developers, it signals the integration of telephony services into the Macintosh platform.

The GeoPort Telecom Adapter Kit, which will be available next month, will consist of

- the GeoPort Telecom Adapter hardware
- new Apple Telecom 2.0 software (a collection of system software extensions and applications and bundled third-party software
- Cypress Research's MegaPhone screen-based telephone software
- Aladdin's SITcomm Special Edition communications software

The Apple Telecom 2.0 software includes

- *Telephone Manager 2.0.* This version of the Telephone Manager adds support for sound recording and playback through the Macintosh Sound Manager.
- *GeoPort real-time services.* This new version of the GeoPort real-time services engine for Power Macintosh includes a native PowerPC implementation of the 14,400-bps Express Modem in software (which now supports v.17 high-speed fax). The GeoPort real-time services software also implements other analog telephony services, such as DTMF detection, dial tone detect, and fax tone detect.
- *Express Modem fax software.* This software now includes a PowerTalk personal gateway (message service access module) and QuickDraw GX support. In addition, the fax application can open a fax and convert it to text using built-in OCR (optical character recognition) software, or it can print it using a fast fax-printing algorithm.
- *GeoPort telecom adapter device handler and Express Modem control panel.* A key feature of the Apple Telecom 2.0 software is its "intelligent answering"—that is, its ability to determine whether an incoming call is voice, fax, or data. This new feature enables communications, fax, and telephony applications to share a single telephone line. This allows GeoPort Telecom Adapter Kit users to effortlessly receive fax documents, voice-mail messages, and (with the appropriate gateways) e-mail through their PowerTalk In tray.

Apple will be bundling a version of Cypress Research's MegaPhone screen-based telephone application with the GeoPort Telecom Adapter Kit. This serves the dual purpose of giving users a complete out-of-the-box solution and whetting their appetites for other telephony applications from other vendors. MegaPhone implements the Telephony suite of Apple events, allowing telephony-aware applications (including Adobe/Aldus's TouchBase Pro and Symantec's Act! personal information managers) to be used as part of integrated telephony solutions and allowing custom telephony solutions to be developed using AppleScript. MegaPhone also supports integration with PowerTalk business cards and synchronization with databases and PIMs using database Apple events.

The GeoPort Telecom Adapter Kit will be available for sale in February. Current owners of GeoPort hardware devices and Apple Express Modems will be able to download the Apple Telecom 2.0 software and the MegaPhone application from various online services, including the Internet, AppleLink, and eWorld; they can also purchase the software and manuals for a nominal fee.

Developer Opportunities

This announcement is the latest step in delivering support for the Macintosh Telephony Architecture, or MTA. (For a more detailed discussion of the MTA and Apple's telephony strategy, see "A Phone on Your Desktop," the Strategy Mosaic column in the March 1994 issue of *Apple Directions*.)

Here are three important implications of this new GeoPort Telecom Adapter Kit:

• If your application tracks names or uses telephone numbers, you should make it telephony-aware. This is as simple as supporting an appropriate subset of the Telephony Apple event suite. If your application is already AppleScript-attachable, a quick way to make your application telephony-aware is to include scripts that make use of the Telephony suite.

• There will be a healthy market for screen-based telephone and programmed telephony (voice-mail) applications. Individual preferences regarding these types of programs are quite varied, and no single human interface or feature set will satisfy the majority of users. As more Macintosh users learn the benefits of computer-telephony integration, demand for new and varied applications will continue to grow. Telephone Manager 2.0, included in the Apple Telecom 2.0 software, provides an enhanced feature set for building these telephony applications independent of the underlying telephony device or services.

• The GeoPort architecture makes it far more practical to create Macintosh-based communications and collaboration solutions because these fax, voice, and data applications can all simultaneously use the same phone line. Equally important, with the intelligent answering capability delivered in the Apple Telecom 2.0 software, the Macintosh computer can distinguish between different kinds of incoming calls and direct a call to the appropriate software.

Apple believes that communications and collaboration is a primary usage for Macintosh personal computers and that computer telephony integration (CTI) will soon be an important element of every user's system. The Macintosh Telephony Architecture is vastly superior to the environments for telephony development found elsewhere. This market is just beginning to grow, so it presents some very attractive business opportunities for innovative developers.

Apple Joins Versit Computer and Communications Interoperability Initiative

What's in a word? Sometimes, the name of a new corporate entity. Apple Computer, Inc., recently announced that it has joined AT&T, IBM, and Siemens Rolm Communications, Inc., as a founding participant in Versit, whose name is taken from the middle letters of the word *diversity.* This global initiative has been established to provide common specifications to improve the interoperability between existing and future computer and communications platforms, including network services.

At a press conference in New York City, company president and CEO Michael Spindler pledged Apple's commitment to make Macintosh a universal client for a wide variety of communications products and services, such as telephones, PBX systems, personal digital assistants, and computer networks. To that end, the company announced Versit-compliant product plans that adhere to Versit specifications.

"Customers are frustrated with proprietary standards that build walls between telephone networks and computing devices," said Michael Spindler. "At Apple, we don't think the customer should have to worry about device interoperability. We will assume that burden for them by making the commitment to produce interoperable products."

Within the next 12 to 18 months, Apple intends to introduce Versit-compliant products in the areas of connectivity, personal data interchange, computer-telephone integration, and conferencing and messaging.

In the area of connectivity, Apple is currently shipping GeoPort-compliant Macintosh computers. Versit selected Apple's GeoPort architecture as the basis for its specifications for a universal multimedia interface. GeoPort is the only cross-platform serial interface that is designed to link a computer with virtually any other computer, mobile device, and telephone and to transmit any kind of information, anywhere in the world.

In the areas of personal data interchange and infrared (IR) connectivity, Apple plans to have "Versit IR"–compliant products for its PowerBook and Newton product lines. This means, for example, that PowerBook users in the future will be able to beam their Versit-standard electronic business cards, by means of an infrared protocol that complies with the IRDA (Infrared Data Association) specification, to any other "Versit IR"–compliant device without any need for data re-entry. In addition, the PowerTalk Personal Address Book will support Versit's Personal Data Interchange Specification (PDIS) for electronic business cards. (PDIS is based on the Bento structured storage mechanism, which is part of the cross-platform, vendor-neutral OpenDoc technology.)

In the area of computer-telephone integration (CTI), Apple intends to continue to extend its Macintosh Telephony Architecture (MTA) by incorporating support for the "Versit Telephony Control Services" specifications. This extension means that the Macintosh computer will provide even greater levels of interoperability and functionality for computer-telephony integration, while existing elements of MTA (including Telephony Apple events, the Macintosh Telephone Manager, and other Mac OS facilities) ensure that Macintosh remains the best development platform for telephony-aware, screen-based telephony, and programmed telephony applications.

Finally, in the area of conferencing and messaging, Apple plans on supporting the Versit specifications that describe interoperability between various videoconferencing architectures over diverse networks.

In addition, the other three Versit founding partners announced details of their support for Versit standards, including the following:

• AT&T announced plans to incorporate infrared support into its Definity System phones to allow them to interact with mobile computers and other wireless devices. AT&T will also incorporate the Versit GeoPort technology into its PassageWay Solutions product line.

• IBM reaffirmed its commitment to the IRDA standard, noting that its recently announced ThinkPad 755CE portable computer already has an IRDA-compliant link. IBM will also support the PDIS standard for electronic business cards and Versit's API (application programming interface) for telephony services.

• Siemens Rolm Communications announced that it too will build GeoPort and IRDA technologies into its future products. In addition, it will evolve its CSTA (Computer Supported Telecommunications Application) standard for CTI into the Versit CTI standard. ♣

Technology

Inside This Section

Human Interface: Fad Gadgets 12

QuickTime 2.0 in the Limelight 14

OpenDoc Human Interface Q&A, Part Two 19

New Technical Notes Available on Paper 21

E.T.O. and MPW Pro #16 Now Available

The #16 releases of both E.T.O.: Essentials • Tools • Objects and MPW Pro are now available. Both products contain significant updates and additions to the core development tools for Macintosh software development released by Apple Computer, Inc.

Most significantly, E.T.O. #16 and MPW Pro #16 include beta versions of improved C and C++ compilers for the PowerPC processor, which will result in dramatic productivity gains for "native" PowerPC application development. The new compilers, MrC and MrCpp, compile code four times faster than the native PowerPC C compiler (which they will ultimately replace) because they use speedy front-end technology licensed from Symantec Corporation. The quality of the code generated by MrC and MrCpp, in terms of run-time performance, remains the best there is on the Macintosh platform because they use the same code generator as the native PowerPC C compiler.

Here are other key features of release #16:

• pre-release Code Fragment Manager 680x0 run-time software, which supports the development and use of shared libraries on 680x0 Macintosh systems in the same way as on Power Macintosh systems

• pre-release System Object Model (SOM) run-time software for building shared class libraries

please turn to page 21

CD Highlights

Tool Chest Edition, February 1995

I'd like to start off this month's column with yet another plea for feedback (and I don't mean the kind my little brother makes with his electric guitar. . . . On second thought, that would be pretty cool; send your sound files to alyx@apple.com). Conveniently enough, we've got a new survey for you in the What's New folder. In this survey you can opine at length about the Developer CD, *Apple Directions, develop,* and other Developer Periodicals products, in a format that should be easier to use than the old SimpleText survey.

Tool Chest Edition, February 1995

Much of the feedback we've received concerns sample code, and DTS has been busy. On this month's disc you'll find a big pile of QuickTime code from the QuickTime 2.0 SDK, and the Macintosh Sample Code and System 7 Samples have been given a preliminary once-over to check for compilability under some of the newer development environments. This will be an ongoing process, so stay tuned.

In addition to the new worldwide system software promised last month, updated Universal Interface files, and snippets, here are this month's new and revised items.

ASLM SDK version 1.1.2

ASLM 1.1.2 is a bug-fix release for ASLM 1.1.1. It is binary-compatible with ASLM 1.1 clients and shared libraries. They do not have to be rebuilt to use ASLM 1.1.2. See the Change History document for a list of changes and bug fixes.

Chinese Language Kit version 1.1.1 Updater

If you have installed System 7.5 and the Chinese Language Kit on your computer, you need to use the Updater program on this disc. Do not use the Updater if you are not running System 7.5.

Important: Use the Updater program after you have installed the Chinese Language Kit software. If you have installed only one version of Chinese (Simplified or Traditional) and later decide to install the other version, you must use the Updater program again after installing the other version.

Developer CD WWW Page

This is a World Wide Web page. It can be viewed by any World Wide Web browser

please turn to page 21

Human Interface

Fad Gadgets

By Peter Bickford

A couple of columns ago, I wrote about the incredible faith our industry has in the power of technology to solve any problem. Along with that faith comes the belief that products inevitably get better from revision to revision—the "New and Improved Whatzit 4.0" is bound to be more powerful, smoother, and altogether more useful than the now-archaic Whatzit 3.1.

But, as they say, it ain't necessarily so. Product designs take all sorts of detours and wrong turns down the evolutionary road. Some, like quadraphonic records and beta VCRs, seemed to be genuine technical breakthroughs at the time but just didn't pan out in the marketplace. Others, like OSI (Open Systems Interconnection) or the Ada programming language, proved to be the technological equivalents of the crock pot—something that everyone had to have one year, and that was condemned to the hallway closet ever after.

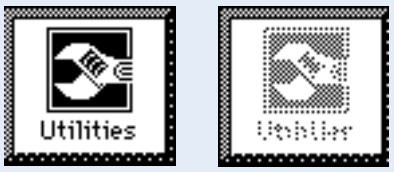
Anyone who's been watching the design game for a few years knows that a lot of what gets designed is less about usefulness than it is about style and fashion. It's like the way stereo equipment colors change every few years, or how one year your car was no good unless it had tail fins. As the new year begins, I thought I'd take the time to look back at some of the human interface fads that have arisen in recent years. With luck, we can all share a good laugh without too many people objecting, "What do you mean, that's a fad!?" and looking distressed because they just implemented their programs in that style. At the same time, I'll try to point out the more long-lasting values that are reflected in the trendy interfaces of yesteryear.

1989: The NeXT Computer, and All Things Dark and Beveled

Five years is an awfully long time in this business. In 1984, Steve Jobs was proudly announcing the Macintosh with its huge 128K ROMs ("a toolbox of built-in routines that allows you to write much smaller programs") and 400K floppy disks ("the disk of the future"). Five years later, he set the industry abuzz with the NeXT workstation from his new company. This impressive black cube practically oozed technological sexiness from every vent. It had a lightning-fast processor (for its time), DMA, and a high-capacity hard drive. This time out, the "disk of the future" was a magneto-optical drive that, although a bit slow, held hundreds of *megabytes* on each disk.

The thing that was most impressive, however, was the look of its interface. Using only four shades of gray, the designers created a sleek 3-D look that perfectly complemented the look of the computer itself. At the time, the Macintosh interface was almost exclusively black and white, with color used only for the desktop pattern and the Apple icon itself. Suddenly, black Chicago font against a white background no longer felt like the cutting edge of interface design.

Gray with jealousy, Macintosh developers immediately began adding 3-D effects to every button, dialog box, and window they could find. Unfortunately, since too many Macintosh computers only supported black and white, the required grays had to be generated using patterns of black and white. The result was that buttons that usually looked something like the one pictured on the left looked,



when disabled, like the one pictured on the right.

Of course, the "NeXT look" didn't stop with buttons. Macintosh developers eagerly applied pixilated 3-D looks to text boxes, field groupings, window borders, and so on. It took well over a year for folks to calm down and realize that it didn't matter how cool your buttons looked if nobody could read them.

Lessons learned:

- Stay true to your own interface—throwing in "cool" widgets from some other platform's interface just serves to weaken your application's sense of consistency and aesthetic integrity. Even if you manage to apply the alternate interface to your entire application, it'll still be something new to learn for users who are accustomed to the regular Macintosh interface.
- Make sure your interface scales to the available technology. The interface that looked great on the NeXT didn't hold up when used on a then-typical black-and-white Macintosh. Your interface will fail if it only looks good when used with thousands of colors, lightning-fast processors, or large-screen monitors.

1990: Tear-Offs

1990 was the year cutting-edge interface designers began tearing off everything: menus, palettes, window panes, and so on. It began with a few paint programs, and the HyperCard Tools and Patterns menus. Later, I'd see program demos where the presenter would proudly demonstrate how every menu in the program could be torn off to become a palette, leading to some very strange results, such as the odd tear-off menu below.

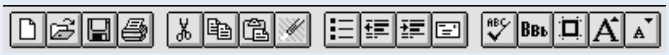
Lessons learned:

This was a classic example of a good idea (tear-off menus) being taken to its logical and clumsy extreme. The point of tear-offs is to put commonly used tools next to the user's work. A palette of tools is clicked more or less constantly to switch between painting and selection, different sorts of brushes, and so forth. An average menu doesn't get nearly the same workout. Instead of making all menus tear-offs, it's better to figure out what the common tasks are and give them Command-key equivalents or other accelerators. Later years would see another solution: the toolbar.

1992: The Rise of the Toolbar

The toolbar can trace its origins to the word processor's ruler or the various window ornaments that had been creeping into Macintosh applications over the previous few years. By 1992, however, the toolbar went from being a couple of handy icons embedded in a window, like the following:

to a free-standing strip crammed with obscure icons, like this:



While the idea of giving the user easy access to commonly used tasks has its merits, the current toolbar mania has really taken the idea over the top. Instead of making options easy to find, they are being obscured by the dozens of similar icons. Moreover, It seems that many of today's ribbons and toolbars are implemented mostly to give otherwise dull programs a bit of graphical "flash," rather than to actually make a user's life easier.

The toolbar mania lives on today, topping out (I hope) with the multiple toolbars, button bars, ribbons, rulers, and control bars present in today's word processors. It is telling that most of these are turned off on users' computers.

Lessons learned:

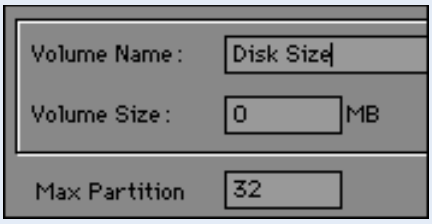
- If you cram too many objects into an interface, it becomes just as hard to access a particular task as if the user had to hunt for it in a menu.
- Limit use of toolbars and window gadgets to the tasks that are most useful to a user. Don't just reimplement the entire menu structure as a set of graphic objects. You should consider letting the user choose which items appear in the toolbar.
- There's little need to put items such as Cut, Copy, Paste, and Undo in a toolbar. The vast majority of users will use the methods they already know for such tasks (either menus or Command-key equivalents). The same goes for Print, Save, Open, New, and other standard menu commands.

1993: Fade to Gray

By 1993, the standard Macintosh computer had advanced to the point where grayscale interfaces were practical, particularly if they could revert to black and white when necessary. This let developers begin experimenting again with interfaces that weren't just black text on a white background.

Used well, grays are a great way to group different items as well as to add a bit of color to the interface. Grays are particularly useful in helping the user see which areas of the interface are active. A good example is using white for text fields and pop-up menus against a light-gray background. You can also use 3-D effects to make buttons seem to rise above the surface of a dialog box, making them look more "pressable."

The current fad, however, is to use grays and 3-D as if the supply of dark pixels will run out tomorrow. Virtually every bounding box is set into its own 3-D plane, and interfaces are becoming darker and darker, until the overall look is a bit like a graveyard filled with beveled slabs and hard-to-read inscriptions. At left is an example of how gray can be used to obfuscate.



Lessons learned:

- As yin needs yang and good needs evil, gray needs white for it to be meaningful. Contrast is everything. Gray looks best when the color next to it is white—not another shade of gray.
- Keep it meaningful. Use white for active objects. Use 3-D effects for things that can be clicked. If you use it for random interface elements, you'll confuse users who are trying to figure out what parts of your interface are active.
- Never, ever, put black text on top of a dark-gray background. It looks sort of neat on sound equipment, but nobody expects to read their stereo controls day in and day out. Eyestrain gets old really fast in an interface.

1994: Collaboration

At the 1994 Worldwide Developers Conference, collaboration was hailed as one of the four key technologies for the future. If we repeat the mistakes of the past, however, it's going to be remembered as just another fad.

Traditionally, collaborative software has been a way for people to work together, adding to or commenting on the work in progress regardless of rank, station, or place in the corporate hierarchy. It's also usually been an utter failure in practice. The trick is that real collaboration is a social event—and you can't expect to have the whole thing work if you ignore the social rules. For instance, the members of a collaborative decision-making project may all put in their two cents, thinking it'll all be weighed equally and that the majority decision reached will stick. On the other hand, 98 cents of that decision might come from the rank-and-file, with pennies left over belonging to the corporate CEO. If the CEO is convinced his contrary opinion is right, guess which way the decision gets made?

Similarly, groupware systems won't be successful if they only benefit one or two people, while the work of using them is carried by everyone else. Several project accounting systems come to mind: The project managers were required to enter in all sorts of data every week so that a financial controller could see how the numbers balanced out on the grand scale. Granted, this was an important thing for a company to do, but for the project managers, it was simply one more dull task to add to their overburdened work weeks. Moreover, it gave the folks upstairs a way to check up on them that was not entirely welcome. Inevitably, the project managers would "forget" to update their project accounts in the system, and otherwise find ways to sabotage things. After a few months of

trying to force the project managers to play along, the systems were inevitably scrapped.

A while back, Jonathan Grudin wrote a great article on this issue called "Groupware and Cooperative Work: Problems and Prospects." It covers these and other problems of building collaborative software. On the bright side, it also explains why electronic mail stands out as one of the few collaborative technologies that really works. If you're working on collaborative software, take a few minutes to track down and read his article—you'll be glad you did.

Human interfaces are part science, part art, and—especially lately—part fashion. Looking over the past few years you can see certain trends: toward building richer-looking interfaces, devising ways for users to access the important features in products that have become ever larger, and so on. In their own way, all of these fads contain an element of true progress in them.

When you design your product's interface, however, try to avoid using a certain look or a certain widget just because everyone else is doing it. Instead, you should be asking yourself if it's really the clearest way to get your message across, or if it's just the style of the moment. Just as in fashion, you generally do best to stick to the classics and avoid whatever extremes are in vogue this season. Later, this year's trendy gray-chiseled look is pretty likely to become tomorrow's Nehru jacket or bell-bottom jeans.

Till next time,
Doc

Got an interface gripe? Send it to Doc at AppleLink address THE.DOKTOR or Internet address the.doktor@applelink.apple.com.

QuickTime 2.0 in the Limelight

New Media Integration Features Help You Quickly Deploy High-Quality Content to Multiple Platforms

By Kris Newby

To watch a movie in 1905, you needed a projector backlit with an acetylene gas "limelight," a person to crank film through the projector, and an organ player. Ninety years later, QuickTime 2.0, Apple's multimedia integration software, is in the limelight, enabling millions of people to watch movies or take virtual walks through 3-D spaces using their home or office computers.

Today QuickTime is in the fortunate position of being the most powerful technology for integrating video, animation, music, text, and other time-based media on personal computers. And it's still the only cross-platform technology available. These attributes have helped it become the most widely adopted multimedia integration technology, with nearly seven million copies installed on computers worldwide.

In this article we explain what makes QuickTime 2.0 technically better than the competition (even if you only develop for Windows-based computers) and why Apple's efforts will provide you with the fastest on-ramp onto the "information superhighway" that will take you into the living rooms (and wallets) of the entertainment-hungry public.

What Is QuickTime?

QuickTime 2.0 is a system software extension that makes it possible for users to view and edit video, animation, music, text, and other dynamic information on Macintosh, Windows-based, and Fujitsu FM Towns computers. This system software architecture not only includes system software, but also compression facilities, human interface standards, and industry-standard file formats. In total, these components provide you with a cross-platform technology that enables you to create multimedia content once, then deploy it across multiple platforms with virtually no additional work.

From a technical perspective, QuickTime supports two basic types of files: pictures and movies. Picture files contain still images, and movie files contain time-based data. Within a movie, QuickTime supports five different kinds of data:

- video, including Motion Picture Experts Group (MPEG) compressed video, a standard used for interactive television
- animation
- music, including Musical Instrument Digital Interface (MIDI)–compatible data
- sound
- text

A QuickTime movie can contain any or all of these data types, and, most important, all of these data "tracks" are automatically synchronized, so a movie's images, sounds, and text are played and displayed at the proper time.

QuickTime is a strategic technology for Apple, and the resources that the company is dedicating to it will help QuickTime developers in several ways. First, by providing you with the best media integration and playback technologies, it helps establish the Macintosh as the multimedia platform of choice. The result? Apple sells more CPUs, and you benefit from a larger installed base of potential customers. And second, by making it easy for you to leverage your development costs across multiple platforms, it makes it easier for you to run a more profitable business, enabling you to create more great titles for Apple platforms.

What's New With Version 2.0

On November 9, 1994, QuickTime 2.0 for Windows began shipping, offering developers of applications for Windows systems a means of delivering the same QuickTime content to both Macintosh and Windows users. One of the first developers to ship a product using both versions of QuickTime 2.0 was Imergy, creator of the hot new title *Star Trek: The Next Generation Interactive Technical Manual.* This entertainment product provides users with a 3-D "virtual reality" tour of television's Starship Enterprise.

"What we like best about QuickTime 2.0 is the significant

improvement in video playback rates for both Macintosh and Windows movies," says Peter Mackey, the project director and lead programmer on Imergy's team. (*Star Trek* is also the first commercial title to use QuickTime VR technology. For a sneak preview of how Imergy used it in their product, see the text box on page 16.)

QuickTime also includes a host of other new features that will help you create better-quality movies and more easily deliver them to multiple platforms.

Details of these features follow:

• *Faster video playback.* On the Macintosh platform, QuickTime now supports professional-level video editing and playback with the inclusion of Society of Motion Picture and Television Engineers (SMPTE) time codes and 60-fields-per-second video. To optimize performance on the Windows platform, this version displays images by bypassing the Windows system's Graphics Device Interface (GDI) and communicating directly with a system's video card.

• *Larger video images.* Version 2.0 makes it possible for users to play full-screen video on entry-level computers. For example, it's now possible to play video at 30 frames per second (fps) on a 25-MHz 486 computer with just 4 MB of RAM. And using QuickTime alone, you can now fill a screen with 320-by-240 pixel full-motion movies. Though this version provides you with twice the maximum screen size allowed in version 1.0, it doesn't provide you with twice the image resolution: To achieve this size increase, the software essentially doubles the size of each pixel on the screen. This resolution is good enough for most applications, but if you wish, you can increase resolution by using a different compression format for your application.

• *Support for interactive television.* QuickTime 2.0 now supports MPEG compression, a standard that enables smoother, higher-quality video playback and creates an infrastructure for development of interactive television applications. (To take advantage of this feature, you need to add an MPEG card to your Macintosh or Windows-based computer.) This format is expected to become the industry standard for the delivery of interactive television applications, such as video on demand (VOD) and home shopping. An advantage to using MPEG with QuickTime is that you can use it to add text annotations or additional music tracks to MPEG movies.

• *Synthetic music support.* QuickTime 2.0 makes it easier to create, edit, play back, and synchronize music with its custom format for describing musical notes stored in a music track. This format is derived from the MIDI standard, so you can import data from standard MIDI files and play the QuickTime 2.0 format through virtually any MIDI playback device.

QuickTime 2.0 also includes a set of 43 musical instrument synthesizer sounds, licensed from Roland Corporation. (The full set of sounds is only available on the CD version of System 7.5: The floppy version contains only the basic synthesizer functionality and piano sounds.) The biggest advantage of this feature is that you don't have to connect a synthesizer to your computer to play back stored music. (Press any key to hear multimedia developers, educators, and musicians cheering.) And by accessing these built-in synthesizer sounds, you can save program memory. (Computers that run software for Windows systems can't take advantage of this feature because of hardware limitations.)

For those of you not familiar with the advantages of MIDI, MIDI-generated sound is to digitally recorded sound what old-fashioned piano rolls are to tape-recorded music: a memory-efficient system for triggering musical notes at the appropriate times. While digitally recorded sound provides excellent quality audio (it's a necessity for reproducing movie soundtracks and human voices), it also consumes a considerable amount of memory—typically around 22K per second. For certain aspects of a project, such as providing background music with "synthetic" instruments, MIDI data can provide similar sound quality at a

What's New With QuickTime 2.0

Feature	Benefit
Faster video playback	• Improves customer satisfaction • Saves time and hardware costs by enabling professional-level video editing and playback on a Macintosh
Larger video images	• Allows you to create full-screen movies that run well on a wider range of computers
Interactive television (MPEG) support	• Enables you take advantage of new market opportunities such as set-top box content delivery, video on demand, and home shopping
Synthetic music support (MIDI compatibility)	• Empowers you to create richer movie music and sound tracks that use less memory • Conveniently allows you to play back MIDI-based music without having to attach a synthesizer to your computer • Opens the door to new types of products that creatively take advantage of built-in synthesizer instrument sounds
Text track support	• Eliminates unwanted font substitutions within movie text tracks • Enables new product features—for example, foreign-language movie subtitles
Custom palette support	• Ensures that users with low-resolution systems view high-quality color in movies

considerably lower data rate. Nearly two hours of music can be stored on a floppy disk using MIDI, compared to one minute of digitally stored sound.

QuickTime 2.0 also supports the Interactive Multimedia Association (IMA) 4:1 audio compression format, a standard that enables you to store higher-quality 16-bit sound on a CD-ROM, along with high frame rate video. And for users who don't have 16-bit sound hardware, QuickTime converts the data to 8-bit sound in real time.

• *Text support.* One common problem that QuickTime 2.0 solves is the unexpected font substitution that occurs when a movie text track font isn't available on a playback system. QuickTime 2.0 eliminates this problem by drawing text while a movie is being created, and compressing this "pre-image" using one of the video compression formats. This way you can incorporate any fonts into a text track, rather than being limited to the fonts on a standard system. And since fonts are drawn in advance, they can be anti-aliased to remove "jaggies."

Text files are simultaneously stored as ASCII data, so this pre-rendered text can even be searched and extracted. And this functionality means you could create foreign language soundtracks for movies, highlight words as they are spoken, create scripts for the hearing impaired, search for specific lines in a movie, or even turn a computer into a sing-along *Karaoke* player.

• *Indeo video release 3.2.* This emerging PC video compression/decompression (codec) standard from Intel has been licensed for inclusion in both QuickTime 2.0 for Macintosh and QuickTime 2.0 for Windows. It allows for real-time compression using Intel's Smart Video Recorder on Windows systems, and it can be played back with Cinepak-like quality (Cinepak is SuperMac's popular 30-fps codec).

• *Custom palette support.* While QuickTime 2.0 provides support for displaying video at 8,16, and 24 bits per pixel, Apple recognizes that many users' systems only support 8-bit color. Left to its own devices, a low-resolution system running a movie with sophisticated color will automatically substitute colors from its limited palette, most often resulting in unattractive color combinations. To avoid this problem, QuickTime 2.0 enables developers to attach an adjusted color palette to movies.

Cross-Platform Development Tips

Apple's long-term goal is to make QuickTime for Macintosh and QuickTime for Windows as similar as possible. Right now it's possible to use QuickTime to deliver the same content to Macintosh and Windows-based platforms, but if you want to author content on Windows-based computers, you have to rely on third-party authoring tools. Until these versions are equalized, there are a couple of differences that you need to be aware of during cross-platform development.

First, files for Windows and DOS systems don't recognize the Macintosh concept of having two forks of information—data and resources—to describe an operation; they recognize only a single data stream. So, to run your QuickTime for Macintosh movies on this platform, files need to be "flattened" by appending the resource fork to the end of the data fork and removing the old resource fork. (This can be accomplished fairly easily using a tool such as MoviePlayer 2.0.)

Second, there are a few QuickTime features in the Macintosh version that aren't available in QuickTime for Windows. One feature missing from QuickTime for Windows is the Roland synthesizer sound library. There are also some QuickTime authoring tools that aren't available with QuickTime for Windows. (For example, when authoring on the Macintosh, you can use a tool that enables you to analyze and adjust audio compression over different segments of a QuickTime movie.)

Mackey talked about the challenges of developing their Star Trek product for both Macintosh and Windows platforms: "We decided it would be easier to author our title on the Mac, but we made sure to try it out on Windows at several points during the development process. This way we were able to adjust for surprises earlier rather than later. One area to look out for in cross-platform development is soundtrack limitations on the Windows side. With QuickTime for Macintosh, we were able to

Imergy "Beams Us Up" With QuickTime VR

The first commercial title to use a pre-released version of QuickTime VR hit the shelves this holiday season, and early sales indicate that it will do extremely well. This product, *Star Trek: The Next Generation Interactive Technical Manual,* was created by Imergy and published by Simon & Schuster Interactive. Star Trek is essentially an on-screen photographic visit to the Starship Enterprise that enables users to randomly zoom in or out of rooms on the ship, navigate from one room to another, and even pick up and inspect objects. For example, users can take a virtual walk down a ship corridor, enter Captain Piccard's quarters, zoom in on a Klingon knife, then examine it closely by rotating it in any direction.

Peter Mackey, the project director and lead programmer on Imergy's team talked about using this new technology. "Our goal with this product was to create an environment where people could acquire knowledge the way they do in real life—by walking around and picking up objects."

For creating virtual environments, Imergy used QuickTime in a very unconventional manner. Instead of shooting video footage to create motion sequences, they took still photographs and then animated them using a stop-motion animation technique. First they traveled to the actual Star Trek television set and took thousands of photos of the set's interiors and corridors. Then they blended the stills together using Adobe™ Photoshop, Macromedia Director, and Equilibrium's DeBabelizer. The advantage of this photographic technique, versus a video approach, was that it resulted in richer colors and superior image quality.

"Our biggest challenge in designing this title was coming up with a spatial navigation metaphor that would feel realistic and be intuitive enough for new users," said Mackey. "We settled on our final design after a lot of experimentation and feedback from Apple's Advanced Technology Group."

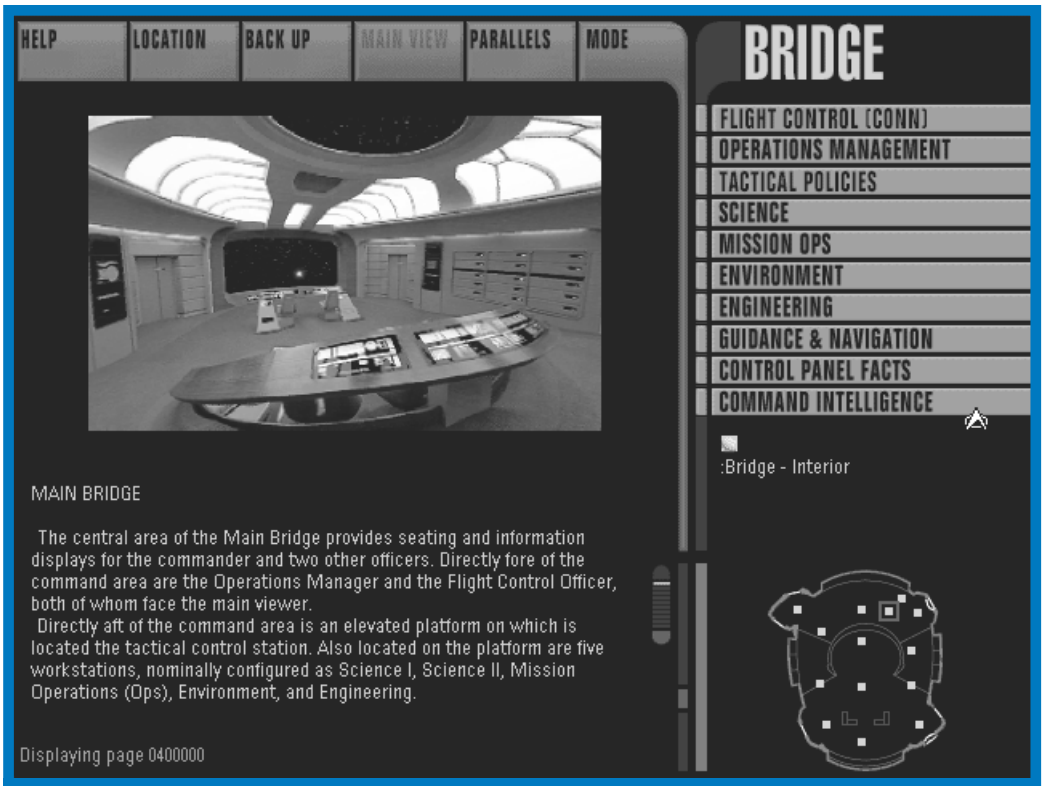
"We were pleasantly surprised at how easy VR was to learn, even though we were using alpha and beta versions of the software. And in spite of the smooth, seamless animations that you see in our final product, we aren't even using all the capabilities of QuickTime 2.0. We haven't maximized frame rates or screen size, and we're not using the text or MIDI tracks. And we can't wait to implement QuickTime VR's cool feature that lets you automatically create links between an object in a scene and a 360-degree navigable view of that object."

mix background sound, music, and other sound effects on multiple tracks inside our authoring program. On Windows, we had to shut down background sound when a QuickTime movie was running, making it more difficult to create seamless sound."

For developers who prefer to use a Windows-based system to create QuickTime movies, there are several ways to do it using third-party products that directly support QuickTime:

- Use Intel's SmartCap program to capture Indeo-compressed video within a QuickTime movie.
- Use Autodesk Animator to create animations.
- Use Adobe Premiere for Windows to edit your video. Premiere can also convert an animation file to a QuickTime movie.
- Use Macromedia Director 4.0 for Windows to combine your QuickTime movies into a complete multimedia application.

If you're creating movies on computers that run software for both the Mac OS and Windows systems, there are translators in the QuickTime Starter Kit that you can use. And you can use Equilibrium's DeBabelizer software to convert nearly any graphics data into a QuickTime movie.



Imergy's new title, *Star Trek: The Next Generation Interactive Technical Manual,* uses QuickTime VR technology to enable users to take a virtual walk through a 3-D representation of the Starship Enterprise.

QuickTime vs. Video for Windows

The other media integration product competing for developer mind share is Microsoft's Video for Windows (VFW). For those of you using this software, QuickTime 2.0 offers you more reasons than ever to move to the Apple standard, the most important of which is that it can save you development time and money.

"We estimate that it costs us about twice as much to create content using VFW and Windows-based authoring tools than it does to use QuickTime and the Apple Media Tool," says Matt MacLaurin of Mediatrix, a multimedia development firm based in Santa Cruz, California. (This firm recently completed a Windows-based project for Dow Jones, and it will ship its first cross-platform game in March.)

Here are the advantages that QuickTime offers you over and above VFW:

- *Easier cross-platform development.* The biggest advantage of QuickTime over VFW is that it's a cross-platform solution, enabling you to reach more customers with less work. You'll avoid the time-consuming manual conversion of content from the Windows system to the Mac OS and be able to market your product to the additional 3.5 million Macintosh CD-ROM users. And since QuickTime enables your program to "share" media files between platforms, some developers will be able to create hybrid discs (one CD that can run on both Macintosh and Windows systems), saving duplication and distribution costs.
- *Built-in synchronization.* QuickTime has synchronization built into the system, so it's easy to register sound, video, and other media tracks to a common time base. VFW, on the other hand, doesn't have built-in synchronization. Using Windows, it's assumed that video will be played at its frame rate and sound at its sample rate—an assumption that requires an element of faith that the tracks will match up across a wide range of various PC configurations. For long VFW movies, manually synchronizing tracks is usually a painful, time-consuming process. And if your movie has discontinuous sound—i.e., a part of the movie where no sound is playing—you have to add "filler" to the music track.
- *Richer media integration.* QuickTime supports more data types than VFW. VFW only supports a set of audio and video application programming interfaces (APIs), and other data formats—for example, text—cannot be easily synchronized to a movie track. Because of this, QuickTime enables you to provide your customers with richer media integration, such as multiple-language text tracks.
- *Better performance.* QuickTime 2.0 for Windows files play back more smoothly and synchronously than VFW files. The reason behind this performance difference is that QuickTime 2.0 can write images directly to a Windows system's video card (QuickTime recognizes most of the major video chip sets), bypassing Windows' performance-degrading GDI graphical layer. Additionally, QuickTime 2.0 is capable of previewing an entire movie and adjusting for synchronization problems before they occur. Video for Windows, in contrast, looks ahead in a movie only a block at a time.

To give credit where credit is due, VFW does offer some features

for professional motion picture producers that QuickTime hasn't implemented yet. For example, QuickTime doesn't support Kodak's KeyKode, a common industry method for numbering movie frames. But overall, QuickTime is, hands down, the feature winner in a comparison against VFW. And there are three QuickTime extensions under development that will provide developers with many lucrative market opportunities—QuickTime VR, CD Plus, and interactive television interfaces.

A Sneak Preview of QuickTime VR

One of the most eagerly anticipated QuickTime extensions is QuickTime VR "virtual reality" software. This software provides users with 360-degree viewing of photographic or rendered representations of a scene. As users change their view of a scene, correct perspective is maintained, providing the effect of being at the location and looking around. The technology is completely interactive, providing real-time zooming, navigating, built-in "hot spots," and the ability to view objects in 360 degrees. And best of all, no special 3-D hardware or goggles are required.

All QuickTime 2.0–compliant applications will have the ability to play QuickTime VR files, so you won't have to re-engineer your QuickTime 2.0 application to tap into QuickTime VR. The compression capabilities built into QuickTime result in small VR files: Each photographic representation of a 360-degree space takes up less than 1 MB. And computer-rendered representations, such as architectural space, have the potential to take up even less memory.

The importance of QuickTime VR is that it will enable developers to economically offer realistic, 3-D information and effects to a large number of users. It's anticipated that QuickTime VR will be quickly adopted by title developers, architectural planners, engineering application developers, and game manufacturers. Apple is currently "product-izing" QuickTime VR and finalizing a licensing fee structure, and it expects to release it to the general public later in 1995.

CD Plus—Music to Developers' Ears

On average, only a small percentage of a typical CD's storage is used when an audio CD is pressed, and utilizing this capacity for computer-readable matter represents a huge opportunity for musicians, record labels, and developers. Musicians realize that this untapped storge can provide them with another vehicle for creative expression, record labels know that they can charge a premium for CDs that include interactive media, and developers see this as an interesting new business opportunity.

Recent efforts by Apple will make it significantly easier for music professionals to use Apple technology in creating interactive music products. First, Apple is actively involved in the development of the emerging "enhanced CD" format, an interface that will enable compact discs to be played on audio CD players and computers' CD-ROM drives. This dual-purpose CD, sometimes referred to as "CD Plus," eliminates the "track-one problem"—the static one hears when a CD with computer-readable information is placed in an audio CD player. CD Plus is a standard interface that enables a CD to be partitioned into two parts—one for audio and one for computer-readable matter.

"We think the QuickTime approach to interactive music is a powerful and elegant solution, because QuickTime is an open cross-platform standard," says Ty Roberts, co-founder of Ion, a Bertelsmann Music Group company. "It's a natural solution for the music industry because the creative services departments of the record labels already use Macintosh computers."

Apple currently has a team of people working on developing key relationships with music industry representatives, and they're working with artists such as Duran Duran, Peter Gabriel, and Thomas Dolby to showcase QuickTime and QuickTime VR technologies. In addition, Apple's new Interactive Music Developer Program, already more than 1,500 members strong, aims to provide music publishers and professional musicians focused resources to help them create interactive CDs. Together, these efforts should help Apple position the Macintosh platform and QuickTime as technologies of choice for the music industry.

Maximizing Your Market Through Players and Interactive TV

Apple is investing in R&D and relationships that, it hopes, will establish QuickTime as an important standard "under the hood" of two types of media delivery systems—multimedia players and interactive TV. During the last six months, Apple made several related announcements that promise to greatly expand the market for QuickTime-based content.

The most recent development was the announcement of Apple's new low-cost multimedia platform, Pippin, a lean-and-mean PowerPC 603 processor–based CD-ROM player that hosts a run-time version of the Macintosh operating system. Apple is licensing Pippin technology to other manufacturers, the first of which is Bandai, Japan's entertainment industry giant. Bandai will introduce their "Power Player" version of Pippin in late 1995 for around $500.

The significance of this technology is that it opens a whole new market for your QuickTime-based

Resources

• *QuickTime 2.0 for Macintosh or Windows.* This software is available on the System 7.5 CD; on CompuServe for $10; or through BMUG for $10 at 510-549-2684, 800-776-2684, or bmuginfo@bmug.org.

• *Run-time licensing.* Developers can license QuickTime 2.0 for redistribution with applications, titles, and media clip libraries for $300 annually per title or $400 annually per title for both Windows and Macintosh versions. If you use the Apple Media Kit to author your product, this product and licensing is available for $500. For detailed licensing information, contact Apple Software Licensing at 512-919-2645 or by AppleLink at SW.LICENSE.

• *The QuickTime SDK.* The QuickTime 2.0 Software Developer's Kit for both platforms is now available for $195. Upgrades will cost $99 through APDA. And all Apple Multimedia Program members will receive this kit in their January mailing.

• *Multimedia Tuner.* This extension includes a set of software enhancements that improve the performance and reliability of some software applications that utilize Apple's QuickTime and Sound Manager software. You can download this tuner from AppleLink (path—Worldwide Multimedia:MM Prods & Services:QuickTime).

• *The QuickTime 2.0 for Windows Technical Backgrounder.* To get a download of this document along with other QuickTime information, look on AppleLink (path—Worldwide Multimedia:MM Prods & Services:QuickTime).

content. You only have to make slight modifications to your current Macintosh software to make it compatible with Pippin-based systems. And titles developed specifically for Pippin can be played "as is" on Macintosh computers.

"Prior to Pippin, moving software from the Macintosh to any of the set-top systems required a huge development effort," says Michael Kripalani, CEO of Presto Studios, Inc. "We're already engaged in many Pippin projects, including a new version of the Journeyman Project."

"Pippin should challenge other game players because of the speed of the PowerPC chip, and the system builds on mature technology—we don't have to 'join a new religion' to write games for it," says MacLaurin of Mediatrix. His firm is also laying the groundwork to release games on Pippin and emerging interactive TV platforms by testing all their screens on TV monitors. "We make sure that our screen colors are optimized and important image areas aren't cut off."

In the realm of interactive TV, Apple has also announced that it will market its own set-top box, and British Telecom has already ordered an undisclosed number of these units for internal trials. What's more, Bell Atlantic is building QuickTime into their set-top boxes. And though details of these set-top boxes haven't been disclosed yet, it's safe to say that there will be an easy migration path for developers wanting to move QuickTime-based content over to Apple's set-top boxes.

Lights, Cameras, Action!

In 1905, who could've predicted the dramatic effect that a simple new technology—the movie projector—would have on our society? Nearly a century later we stand on the brink of a revolution in "content" delivery that is just as exciting: Computer technology will enable us to deliver information and entertainment in ways that will surprise and delight us all.

You're in the fortunate position of being able to create those new titles, and QuickTime technology can help you in your quest. Apple's efforts to establish QuickTime as a cross-platform multimedia integration technology will enable you to develop content once, then deliver it efficiently to users' platforms of choice. So pull up your director's chair and QuickTime 2.0, and put those ideas into action. ♣

Kris Newby is a technical communications consultant and writer based in Palo Alto, California.

OpenDoc Human Interface Q&A, Part Two

By Dave Curbow and Elizabeth Dykstra-Erickson, OpenDoc Human Interface Team

This month, we continue to answer questions we've received about OpenDoc from engineers in the human interface community through conversation and e-mail.

Q: When a frame is resized, should the crop area of the contents change, or should the contents be scaled?

A: When the user resizes a frame, two part editors are involved—the editors for the container and the embedded part. When the user has selected a part frame and resized it, the default behavior of the selected part's editor should be to change the cropping of the content. For example, if the selected part is a spreadsheet, resizing will show more or fewer cells. Optionally, your part editor may have a setting to allow the user to specify "change cropping" vs. "scale to fit frame." For "scale to fit frame," the currently visible content should expand or contract to fill the resized frame.

We recommend that your application change the cropping of the resized frame as the standard behavior for resizing. If your editor supports scaling of embedded content, you should employ a separate mechanism for scaling. Consult the *OpenDoc Human Interface Guidelines for the Macintosh* for a description of different approaches.

Q: Some of today's applications support different modes. For example, HyperCard allows several modes, including use mode, several different edit modes, and a paint mode. How should multiple modes be implemented in OpenDoc?

A: By all means, use multiple modes for your part if that makes sense. Do follow the standard development guidelines for making the kinds of visual distinctions that will help users identify which mode they're in. For example, use distinct tools other than the standard arrow tool for different modes.

Q: My part editor supports different tools. What should happen when my part becomes inactive? For example, if a paint part is set to the eraser tool, should it remain so when the frame becomes inactive?

A: Since the guidelines suggest that cursor feedback over inactive parts should be the content-specific cursor, the part should retain its tool state when it becomes inactive. However, there are exceptions. Because OpenDoc uses inside-out activation, some tool states can cause the user to make inadvertent changes to content. For example, a single click in an inactive frame whose tool is set to the eraser could erase content; a single click in an inactive frame whose tool is set to the draw tool would create a point. Therefore, when your part becomes inactive, change to a safe, neutral tool such as a selection tool. This is consistent with the way many applications work today: Tools such as the eraser tool work for only one time, then revert to the selection tool.

Q: The linking functionality is hidden within copy and paste. Why doesn't linking have its own command, such as Paste Link, instead of hiding under Paste As?

A: Copying is copying. In fact, both copying and dragging are simply transferring data. Copying is a simple-to-use, fundamental operation. Linking is a feature of copying that allows the resultant pasted copies to stay synchronized. The mental model we want to support is that linking is just an automatic method of copying. This adds some interesting complexity to the copy operation. OpenDoc hides this complexity from the user except when the user needs explicit control of pasted content. In that case, the user may indicate whether copied content should simply be pasted, or pasted and linked to the original. Additionally, copied content can be pasted as an embedded part or merged with the destination. Since linking has the same set of options as a paste operation, linking is implemented in OpenDoc as a part of the paste operation.

Q: In OpenDoc, how would I make annotations in the margins of a document? Can I drop annotations in as a part?

A: The part developer can choose to support this feature. In OpenDoc, every part (except the root part) can be thought of as being contained within another part. Each container controls the layout of its embedded parts. A container may allow the user to place a part inside the margins of a page, but the container editor must be written to support that capability. The container editor would need to detect the case where the user drags a part or piece of content to the margins, and then place the dragged item there.

Q: Every document should be viewable. Why doesn't OpenDoc allow part viewers to be included in the document?

A: We agree that every document should be visible, but encapsulating multiple part viewers into a document needlessly swells the size of the document, just as always storing a PICT file for every part does.

We discussed viewers and translation as OpenDoc's answer to the "missing editor" problem in last month's article. OpenDoc allows the part to store multiple data formats for a part. And while a PICT may be appropriate in some circumstances, other formats are often more compact.

We considered a "Make Self-Contained" feature that would produce a document with all the necessary viewers encapsulated but decided against this, because it looked like a route for viruses. Also, we see this as a third-party opportunity for creating tools that produce self-contained documents.

Q: When the user opens a part into a window using View In Window, should my editor save those window positions?

A: We don't see any reason for OpenDoc parts to deviate from the standard Macintosh human interface guidelines for windows. To reiterate those guidelines: Before closing a window, check to see whether the user has changed its size or position. Save window positions, and reopen windows in the size and position in which the user left them. If a user opens, moves, and closes a document window without making any other changes, save the new window position but don't modify the date stamp of the document. If the user does not change the size or position of the window, don't save the position when the user closes the window. Before reopening a window, check to make sure that the size and state are reasonable for the user's current monitor or monitors, which may not be the same as the monitor on which the document was last open.

Q: Can I use a Window menu for subordinate windows and window-management commands such as Tile and Stack?

A: OpenDoc uses the Application menu (also known as the Process menu) as a way to find the documents that are currently open. So, in contrast to today's model, in which a Window menu shows only the windows of the current application, in OpenDoc the Application menu is system-wide.

Therefore, we recommend limiting the use of any Window menu to situations in which there are multiple views of a document, such as subordinate windows. For example, in a CAD system, it may make sense to list these views or related commands in a Window menu. In this case, you should consider whether a View menu adequately addresses their needs before resorting to a Window menu. See the *OpenDoc Human Interface Guidelines* for detailed discussion of these menus.

Q: Does OpenDoc allow copied content from a non-OpenDoc application to be pasted into an OpenDoc document?

A: Yes. The first thing to keep in mind is that the transferred data is not a part, it's just data. There are two cases to support: (1) the destination part can receive the data and merge it into itself; and (2) the destination part can't merge the data but allows it to be embedded.

When a destination receives a paste or drop, it looks at the contents and tries to incorporate or merge the data into its contents. For example, suppose some text from a word processor is pasted into an OpenDoc text part. The OpenDoc text editor sees that one of the kinds of data available on the Clipboard is text and pastes it in. In fact, there are often different kinds of content available. In this case, there may be simple text as well as richer formats such as styled text. The OpenDoc text editor should use the richest format possible in order to preserve information.

If the content on the Clipboard is of similar kind, then it can be incorporated into the destination. The destination part determines whether the content is similar by using the *kind* of the content to look up its *category.* If the category is the same as the destination, then translation is used to incorporate the content.

If the category is different from that of the destination, then a part of the appropriate kind is created to store it, and it is embedded in the destination part. If the contents being added were created with SurfWriter, then a SurfWriter kind of part is created; if the SurfWriter editor is not available, then the part is defined by the user's preference. So, if the text is then pasted into SurfDraw, the system examines the user's preferences (set in the Editor Preferences control panel on the Macintosh computer), creates a part of the preferred kind within the drawing, and finally puts the pasted contents into that part.

Q: In embedding OLE parts, when should OLE standards be used and when should OpenDoc standards be used?

A: We believe that OpenDoc provides a better overall user experience than OLE 2.0. Thus, whenever OLE objects are embedded within OpenDoc documents, we attempt to present an OpenDoc experience. For example, the menus are made to match the OpenDoc standard.

However, it is not always possible to make these OLE objects behave exactly like OpenDoc parts. Therefore, we use the OLE visual feedback for OLE parts so that the user will at least know that these are not standard OpenDoc parts.

Q: What kind of standards exist for common command elements, such as menus and toolbar buttons?

A: We are working with CI Labs to create standards for toolbar icons, placement of items and frequently used menus, toolbar reuse, and standard icons. If this is an area you're interested in contributing to, please contact us. ♣

Got a burning question? Need some clarification? Subscribe to OpenDoc-HI@CILabs.ORG and send us your questions and comments. To subscribe, send a message to ListProc@CILabs.ORG and include "subscribe OpenDoc-HI <your name, not e-mail address>" in the body of the message. We also check for questions on the Applelink Human Interface Bulletin Board.

E.T.O. and MPW Pro

continued from page 11

- a version of PPCAsm (the PowerPC assembler) that supports symbolic debugging
- a pre-release version of a native PPCLink linker that generates PEF executables directly
- Ad Lib, a new user interface editing tool for building MacApp 3.x views
- MPW p2c, an Object Pascal to C/C++ source-code translation system

To obtain E.T.O. or MPW Pro, contact APDA or your local Apple office. (See page 28 for information on how to contact APDA.)

New Technical Notes Available on Paper

You're probably used to looking for new Technical Notes from Apple Computer, Inc., in electronic form on the Developer CD, the *develop* Bookmark CD, or the Developer Services AppleLink Bulletin Board. Now, the entire set of Technical Notes, including recent additions, are available on paper from Field Copy and Printing, who will also be updating them on a quarterly basis.
You can order a complete printed collection along with a subscription to the next three quarterly updates; or, if you already have the latest Technical Notes on paper, you can choose to receive only the quarterly updates.

The complete electronic set of Technical Notes will remain available on the Developer and Bookmark CDs, as well as on AppleLink. If you'd like to order the paper versions, contact Field Copy and Printing (phone: 415-323-3155; AppleLink: FIELDCOPY). ♣

CD Highlights

continued from page 11

(such as MacWeb, Mosaic, or NetScape). Clicking the hypertext link on this page will take you to the Developer Services main page on Apple's World Wide Web server. On this server, you can find most of the content of this CD (in ftp form). You can also view online versions of *Apple Directions, develop,* and Technical Notes.

Developer Notes Update 01/95

Included here is a developer note for four new products: the Power Macintosh 6100/66, 7100/80, 8100/90, and 8100/110 computers.

Developer Press Survey 1.0

The Developer Press organization is interested in your opinion, in order to improve our information and documentation products. We are focusing on what content you like and need, and how you use our current tools to access that content. Filling out this survey (which should take about 15 minutes) and sending it in by electronic mail will help us deliver our information according to your needs. Additionally, if you supply your name and address, we will automatically enter you in our monthly drawing for a free Developer Press product!

Dylan Related

This folder contains information and sample code pertaining to the Dylan programming language. New items this month include demos built with Dylan 1.0a1, a new FAQ document, and Dylan information of interest to Japanese developers.

Macintosh Technical Notes Update

Technical Notes are a collection of short (and not-so-short) articles dealing with specific development topics. This month features three new notes related to QuickTime: "IM—QT Components Addendum," "IM—QuickTime Addendum," and "QuickTime for Windows Addendum."

Palette&GWorld (Fat)

This code sample demonstrates how to use a palette when drawing into a graphics world (GWorld). The trick is understanding that while you may set a palette to a GWorld, doing so does not change the GWorld's color table.

The solution is to make a palette from the color table (or the color table from a palette) and to use that color table to create or update the GWorld. After calling SetGWorld, you can either draw with the Index2Color and RGBForeColor routines, or you can set the palette to the GWorld and draw with the PmForeColor routine. These techniques are shown in the procedures createRGBForeColorImage and createPmForeColorImage in Palette&GWorld.c.

QuickTime Sample Code

This folder contains a selection of QuickTime sample code from the QuickTime 2.0 SDK.

Sprocket

Sprocket is a small, System 7.5-savvy framework. Think of it as a "Powerplant for dummies."

Coming Next Month

DocViewer's Last Stand?

Alex Dosher
Developer CD Leader

Business & Marketing

Marketing Feature

Market Research Monthly will return next month.

HyperCard Alive and Well

The Tool of Many Users

By David Gleason

The message of this article can be expressed concisely: HyperCard is alive and well, and for you as a developer, this is very good news.

Let me explain. As a software developer, you are constantly dealing with the demands and constraints of development tools. It seems there are too many choices, and often it's difficult to find the right tools for what you need. And the proliferation of programming platforms and options is growing, not getting smaller. It's confusing. Yet sometimes the answer can be found in a tool that you know and that is close at hand, but that you simply haven't thought to use in awhile.

I recently spoke with five developers who are using HyperCard for a wide range of Macintosh software development needs. The ways in which they employ HyperCard may surprise you and, at the same time, give you some ideas about how you can use HyperCard to enhance, simplify, and speed up your development efforts. Perhaps these stories will inspire you to create new products or solutions that you can implement quickly with a familiar tool.

The developers featured in this article represent the following diverse categories:

- CD-ROM-based entertainment product developers
- developers of database front ends and client-server–based applications
- systems integrators and value-added resellers
- developers who are not really programmers
- educational product developers

Despite their diversity, in each case you will see that HyperCard is a powerful and versatile development tool. HyperCard may be able to help you create new products and meet customer needs as it has for these developers.

HyperCard and System Software

Although HyperCard version 2.2 has been available for some time, subsequent releases of Macintosh system software have significantly enhanced HyperCard's value to developers. Of special significance is the way AppleScript is integrated with HyperCard, providing tremendous flexibility to developers in creating custom solutions. In addition to using HyperCard's HyperTalk scripting language, built-in features, and XCMD architecture, you can now extend that functionality by using AppleScript to launch, control, and exchange data with more than one hundred scriptable applications. For more information, see the text box "Key Features of HyperCard Version 2.2"on page 26.

Also consider that with Macintosh System software version 7.5, the Finder is scriptable. This permits HyperCard to control and customize the desktop through AppleScript. For example, HyperCard can be used to automate a weekly report in a word processor such as WordPerfect. The script could retrieve and open a report template in PageMaker, automatically insert the report text, and then open the most recent budget spreadsheet in Microsoft Excel, selecting current figures and copying them directly into the report template, which the script then saves as a new file. All of this scripting can all be controlled within HyperCard—for example, by a stack that notifies you when the report is completed and displays it automatically.

Apple Supports HyperCard

You may recall that when HyperCard was released in 1987, it caused quite a stir, and introduced an element of fun and excitement to the world of Macintosh development. At the same time, it opened up a realm of programming to anyone from school children to academics, musicians to theologians. And because HyperCard was so different, no one could quite define what it was. Everyone could comprehend the basic HyperCard metaphors of the stack and cards, but beyond that, it wasn't at all clear what to call this thing. That hasn't changed, and, in fact, there are probably more ways to define HyperCard today than ever—authoring tool, information organizer, computer erector set, front-end development tool . . . you name it.

What gives HyperCard its indefinable quality is the way in which it permits the user to access and display information in an unstructured yet predefined manner. Stacks and cards can be organized in an endless variety of ways, and information can be assessed and displayed in a myriad of fashions. There are many ways to combine and present text, graphics, and data, and how you put it all together is entirely up to you.

But it is more than the interface that makes HyperCard such a powerful development tool. The power "under the hood" is in HyperTalk, the scripting language that is built into HyperCard, and now other scripting languages can be used with HyperCard as well. HyperCard commands and functions, called *XCMDs* and

XFNCs, have been written in programming languages such as C, Pascal, and assembly, and they can be added to HyperTalk scripts and stacks to extend the power of HyperCard in many ways. Over the years, thousands of XCMDs and XFNCs have been written and distributed; many are available for free on electronic bulletin boards. Many application developers include them with their products.

According to Peter Christy, senior director of the Apple Developer Products Group, "HyperCard broke new ground when it was introduced by providing a tool with a unique and broadly usable model of 'programming.' Since its introduction, HyperCard has been used to create countless valuable applications ranging from personal tools through corporate applications all the way to some of today's most innovative and successful multimedia applications."

Christy makes it clear that Apple sees the value in supporting HyperCard as a development platform. He states, "Apple is more committed to this kind of solution tool than ever before. We in the Developer Products Group recognize that our 'developers' include a lot more HyperCard users than C++ programmers, and in sheer numbers there are more applications created with HyperCard and similar tools than with C++ and other traditional programming languages."

Further evidence of HyperCard's continuing popularity as a development platform can be found in the Redgate listings on AppleLink (path—Developer Support:Redgate Directory: HyperCard), where some 465 commercial HyperCard products are currently listed. And consider that hundreds more products are developed as customer solutions, and are not available to the public. In addition, you can use America Online to access the HyperCard Forum, which provides a number of software libraries, including several hundred stacks available online. You can download the extensive library of XCMDs and XFNCs written by HyperCard expert Frederic Renaldi. Peter Baum is the Apple Forum Leader for HyperCard, and he can be contacted at AOL address: AFL HYPCRD or key word: MHC.

What's in the Box

If you haven't looked recently, I suggest you review the current version of HyperCard and what it includes. HyperCard version 2.2 (released a year ago) introduced a number of significant features, including the following:

- AppleScript/OSA support
- ADDMotion II from Motion Works
- the Color Tools stack (includes the AddColor XCMD)
- the StackToApp extension for building stand-alone applications
- enhanced QuickTime support and Tools stack
- WorldScript support

See the text box "Key Features of HyperCard Version 2.2" (page 26) for details on these features and how they can help you to develop interesting applications.

If these features don't provide all you need, the HyperCard package includes a brochure of more than 40 HyperCard companion products and services that can improve your use of HyperCard in such areas as printing, scripting, compiling, generating reports, building tables, adding color, and more. Like any powerful and popular development environment, HyperCard has generated a minor industry of add-ons and supplemental products that can extend the power of HyperCard for you and your customers.

Now let's take a look at some outstanding examples of HyperCard end-user products that have been created by developers and are in use in the market today.

First, There's Entertainment

Consider first of all that HyperCard is a tool you can use to create products that are interesting, exciting, and even fun. And there are many things you can do with HyperCard that provide all of these benefits, as well as potential profit for you. An outstanding example of HyperCard's usefulness in multimedia development is the CD-ROM *Myst* by Cyan. Like the best of CD-ROM games, *Myst* is complex, entertaining, engaging and graphically impressive.

Cyan, founded by Rand and Robyn Miller, is a company in eastern Washington state that created a number of children's games in HyperCard, including *The Manhole.* Their newest product is *Myst,* a wildly successful CD-ROM–based game, which had sold at least 250,000 copies by the middle of 1994 (and the rate of sales is still climbing). *Myst* takes you as the user on an experience that begins mid-journey, placing you in a land and time that are unknown and unfamiliar. The application provides no menus, no initial text, just a few clues; you have only a hand cursor and your imagination and curiosity to motivate and guide you. Progress in *Myst* is controlled entirely by navigation, and even that relies on user intuition—there is no hand-holding in *Myst,* which is part of its pleasure. It is also a graphically stunning production, with some 2500 rendered color images.

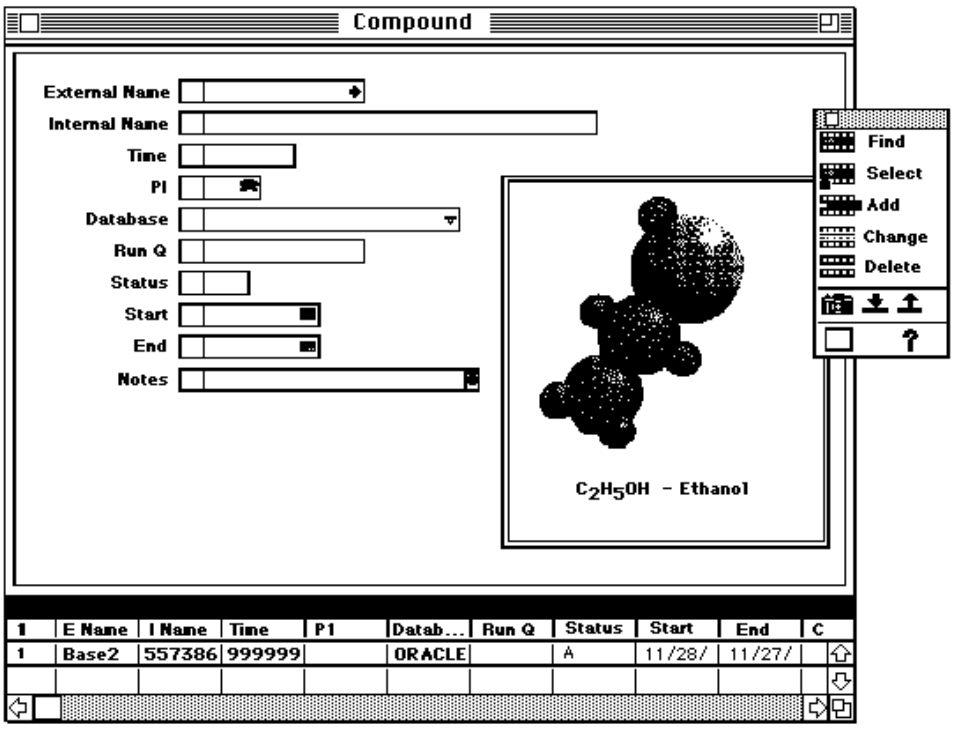
Rand Miller, the lead programmer and one of the masterminds behind *Myst,* says, "Basically, our business is here because of HyperCard; we love it. *Myst* is mostly HyperCard, with some XCMDs for color and QuickTime and sound. It's used for all the products we sell; I've been programming with HyperCard since 1987. We use HyperCard in our business; we have our time cards in HyperCard."

Miller, somewhat of a cult figure in the cyberspace press, is direct in his appreciation of HyperCard's ease of use as a development platform. He says, "HyperCard makes it easy to put things together. It's great at not getting in the way of the creative process. HyperCard allows us to focus on content—it's almost like it's not there."

Myst is a clear descendant of *The Manhole,* a less complex game that was first released in 1988; *Myst* shares much of the graphic imaging and sense of motion and exploration that *The Manhole* introduced. Miller says, "*The Manhole*—that's how we got started. I was interested in doing some children's products, that allowed you to use pictures, as an interactive book. So I sent HyperCard to Robyn, and he was hooked. He wasn't a programmer, but he started staying up late working with HyperCard, because he loved it. It allowed him to link pictures and build these worlds."

Myst was an improvement on *The Manhole* in several ways; not only is it more adult and more complex, but it uses color. Rand Miller points out that the limited implementation of color on HyperCard turned out to be a benefit. He explains, "We were anxiously waiting for color, but as it turned out, it worked out fine, because it allowed us to make a smaller, faster application. Because HyperCard is all black and white, that makes for an incredibly fast shell. Rather than be burdened with all the things that come along with color, we are able to go to XCMDs, and keep the speed."

HyperCard is not the only tool they use at Cyan, but it is a preferred one. Miller says, "We love Macromind Director, and we use it, but we use it more as a time-based animation tool. We are not



Base2 screen for HyperCard application that displays data when user clicks on chemical compound "hot spots."

time-based—that's not the right paradigm for our worlds and how they work."

As for future products, the people at Cyan are looking at a number of options, but they still maintain their faith in HyperCard. "We are doing due diligence and investigating other products," he admits. "But it's going to be tough for any product to compete with HyperCard. Most of them are trying to be all things to all people." Miller says he would like to see HyperCard add more integration of color, "but not at the high end. There are already some good products out there for the high end. I'd rather see HyperCard take more of a middle level, that allows you to control your own destiny."

There are other interesting CDs based on HyperCard; the *Explora 1* CD by Peter Gabriel offers a musical adventure as interesting as *Myst* but in a different way. While *Myst* is a game with a progression and an end result, *Explora 1* is more of an exploration of a multimedia musical experience from famed British musician Peter Gabriel. Subtitled *Peter Gabriel's Secret World,* it too takes advantage of the user's curiosity and imagination, and allows different pathways and discoveries to motivate and delight you. Along the way, you get to hear some excellent music, see top-rate, well-produced videos, and go backstage to meet musicians and try out exotic instruments yourself. The varieties of experience that are offered in this CD are numerous—it probably would require a full page just to flow-chart all the things you can do. Comparing such diverse products as *Myst* and *Explora 1* suggests the range of possibilities open to a developer who has imagination and a vision of a product that requires a development environment such as HyperCard can provide.

Systems Integrators and Value-Added Resellers

While not as entertaining as CDs, database access tools created from HyperCard are an area of tremendous interest to users, and an area of significant development opportunities. HyperTalk's scripting capabilities, much expanded with version 2.2, provide a rich graphical information display tool set, which can be used to access and display information that is both graphically pleasing and relatively easy to implement.

From the beginning, HyperCard has been used as a prototyping tool for databases on mainframes and minicomputers. Over the years, developers both large and small have created thousands of front ends using HyperCard.

Base2. An example of a small company that relies entirely on HyperCard for its development environment is Base2, located in Los Gatos, California. Cecil Koupal uses HyperCard and a suite of extensions (both XMCDs and XFNCs) to provide a tool that can "self generate" forms directly from the database schema.

Koupal says, "We are professional developers, and our approach is to bring commercial-grade products to the database market faster than has been possible with older, third-generation languages such as C. I see HyperCard as what I call a 'RAD C-S tool'—that is, a 'rapid application development client-server' tool. It is what I call crossroads developing. In fact, given the reality of high-speed, low-cost Macintosh computers, scripting can replace C the way that C replaced assembly in the 1970s.

"People are using databases transparently these days; the database interfaces are growing in sophistication, and this means incorporating the features of multimedia. Users want to see multimedia ideas used in their work. In other words, older ASCII interfaces are being replaced by sophisticated graphical, multimedia and heuristic elements that enable a new level of user-data interaction."

Koupal sees scripting as a solution to the increased complexity that all these new expectations create on the development cycle. "Scripting environments are where you get the greatest increases in productivity," says Koupal. "With HyperCard, you get more of a tactile feedback; you get an instant result, so you know if you are on the right track. You can often do with 10 lines of HyperTalk what would take you 1500 lines of C; you can look at the code and see—really see—what you are doing. With C, it takes a lot longer to write it, and much longer to review it and see where you are going."

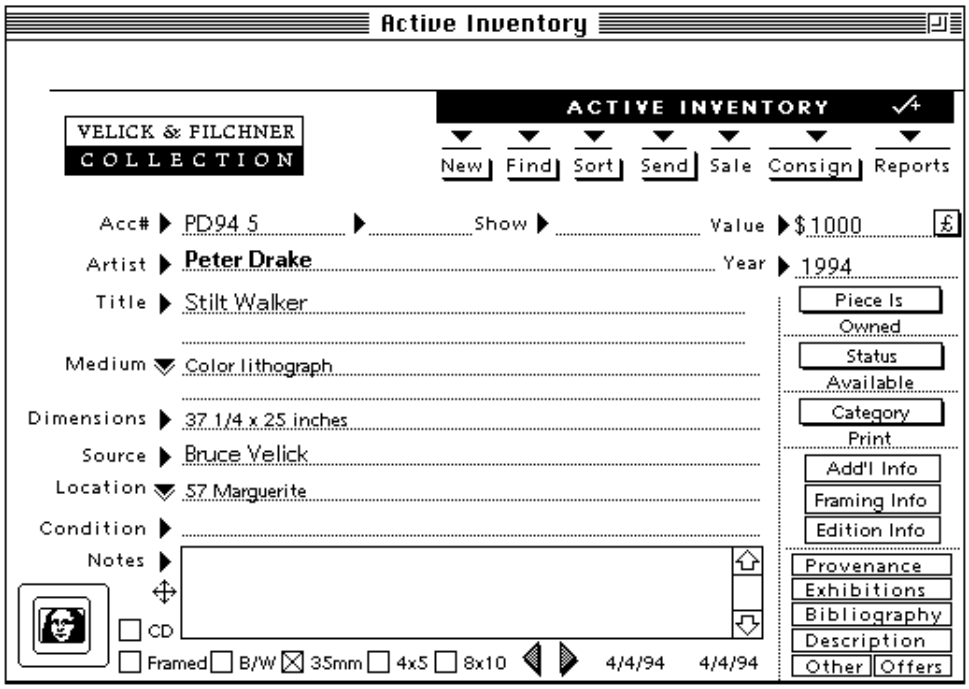
Koupal believes the power of scripting has not yet been fully realized. He says, "We could revitalize the professional development world with a full-blown implementation of scripting."

Recently, Koupal experienced an example of the power of scripting in his own skilled hands. As

he tells it, "We were doing a demo for a biomedical firm. There are two incompatible sectors in the biomed industry—the chemists versus the biologists—and because they don't always see eye to eye, they don't cooperate as fully as they might. So there isn't a lot of useful software." This potential client came to Koupal and expressed interest in a front end that would permit the user to display a graphical image of a chemical compound. "He wanted to be able to click on certain hot spots," explains Koupal, "and have that pull up a window with information on that specific part of the chemical structure. The idea was to simply bring up the chemical picture and click the hot spots." So Koupal gave a demonstration of what he could do, using HyperCard, to create such a front end. He says, "Creating that could takes weeks with C, but with HyperCard, it takes minutes."

So what did the customer think? Koupal says, "He was sold."

PCI software. James Fox is an independent contractor who does custom programming for a variety of customers using HyperCard. For example, a recent project was for a mortgage company that put out advertisements, listing their toll-free 800 telephone number; back at the office, they have five operators who take calls. Fox developed a HyperCard system that takes the caller through a detailed set of questions. Upon completion of the interview, a financial profile of the caller is created and the data is recorded into a FileMaker Pro file by means of a combination of HyperTalk, AppleScript, and XCMDs. It's an example of a product that can be quickly and easily created in HyperCard, with a lot of power and flexibility. PCI also does a lot of multimedia programming, using a combination of Macromedia Director and HyperCard.



The active inventory screen of Art Stacks.

When I reached Fox, he was working on a project for the medical staff at a hospital in Sacramento, California. Fox explains, "A couple of the doctors had written a HyperCard program to record their patient progress notes. They are smart guys, but they are doctors, not programmers, so they brought me in to improve the level of the product, and also to network it. An added benefit of HyperCard on this project is that a doctor who is familiar with HyperCard can customize his own copy of the application to his specific needs."

While he could have used any of a number of programming languages for such a task, Fox chose HyperCard as the best solution. Fox says, "The real power of HyperCard is hidden. I think that not enough people seem to know what it is, and many people have a misunderstanding that it isn't powerful enough. I'm using HyperCard because it's the best product for the job, and I'm just more comfortable with HyperCard. It's more like the way I think. I'm an artist, a creative type, and I prefer developing in HyperCard. It allows me to immediately test my code as I write it, to work dynamically, as opposed to the 'code-compile-test' routine that a programmer must endure in C or assembly."

In the areas where HyperTalk doesn't have enough power or speed, Fox relies on XCMDs. He says, "For me, my collection of XCMDs and XFNCs are like a library of routines. I have hundreds of them, some of which I wrote myself, some I got off of CompuServe for free, some I paid a small sum for. With XCMDs and XFCNs, I can compete with other providers who are writing in Pascal and C."

Fox adds, "When deadlines and tight budgets are a consideration, HyperCard offers the best, and sometimes the only, solution. You can develop powerful custom applications in a fraction of the time that it would take in a 3GL. And HyperCard is very stable; that's something that several other environments can't boast."

Products From Non-programmers: Art Stacks

Not all HyperCard providers are programmers. Because of its ease of use and friendly interface, HyperCard has opened up the world of programming to thousands of nontechnical people who simply want to create a product that works for them. And some have turned those home-grown products into commercial successes.

Bruce Velick is a model of a professional who became a programmer out of necessity; before working with HyperCard, he had absolutely no programming experience, and no desire to learn. A photographer and collector, he simply wanted a tool to help him keep track of his art collection and his gallery, and to help him run his business.

Art galleries require a lot of specialized information, such as

registration numbers for works of art and assessed value, and they also need to track the art that they collect and sell. But because they make up a relatively small market, there are few specialized tools available on any computer platform. And in the art world, the tool must be easy to use. As Velick says, "Art people don't want to be computer people. They just want to be able to get the information they need out of that box."

For Velick, the solution did not come easily or directly; in fact, becoming a HyperCard user was a result of a series of unlikely events. Velick explains, "I had an art gallery in the 1980s in San Francisco; then a guy from Apple came into the gallery. We traded some artwork for a Macintosh. So that's how the product got started, just for my own use. And then I began showing it to other people. Some dealers in New York said, 'Gee, I could really use this.'

"But with the recession of the 1980s, people stopped buying art. So I expanded the capabilities of the program, for collectors and artists as well as galleries. Galleries require more information—invoicing, client information, and so forth. Most collectors don't need all that. But instead of separate programs for each, I did it for both." And it was a good decision, because it increased Velick's potential customer base.

In 1987, when HyperCard was still in beta form, Velick began working with a HyperCard programmer, and together they created the product that is now Art Stacks. The choices then were limited in terms of third-party applications such as databases, so they decided to do everything in HyperCard, from financial management to inventory, graphics display, and mailing lists. They used some XCMDs and external code, but the bulk of the product is in HyperTalk scripts.

Art Stacks includes a Home stack with nine stack options to choose from, including artist accounts, active inventory, sold inventory, mailing list, invoices, and more. Velick keeps separate stacks for handlers, for XCMDs, and for graphical images of the art work, thus reducing overhead and stack size for the primary stack. And Art Stacks, because it is well designed, is surprisingly fast at accessing and displaying information. For example, Velick did a demo for me, searching his mailing list of 1100 names, finding 67 entries who had expressed interest in a particular artist. Running on a Macintosh IIfx, the search and display process took less than 4 seconds.

Although today he could replace some HyperTalk scripts

Key Features of HyperCard Version 2.2

If you haven't worked with the current version of HyperCard, you may be interested to know what is included. In addition to AppleScript and WorldScript support, HyperCard 2.2 also includes the Color Tools stack that provides color, and QuickTime and AddMotion for animation.

AppleScript

Probably the most exciting feature of HyperCard version 2.2 is support for AppleScript and any other scripting language that is compatible with Apple's Open Scripting Architecture (OSA). Examples of OSA-compliant scripting languages are UserTalk from UserLand Software, Inc., and QuicKeys Script from CE Software. When you open the HyperTalk script editor, you automatically are offered two scripting language choices, HyperTalk and AppleScript. If you install other scripting languages, they are displayed as well.

Recall that AppleScript provides access to Apple events, and so is an interface to the Apple interapplication communication technology that is built into System 7. Applications can have three levels of AppleScript compatibility: They can be attachable, scriptable, and recordable. Note that HyperCard is both attachable and scriptable, but it is not recordable.

HyperCard 2.2 supports the core and required suites for Apple events, and provides its own HyperCard suite. There is message passing between AppleScript and HyperTalk scripts, so HyperTalk handlers can call AppleScript functions and vice versa. Also, note that HyperCard's XCMDs and XFNCs are accessible to AppleScript. In HyperCard, scripts can receive messages as well as send or pass messages to other scripts regardless of the script's language. So a button script written in HyperTalk can send a message to a card script written in AppleScript, which can in turn pass the message to a background script written in UserTalk.

These capabilities dramatically increase your ability to use HyperCard as an engine from which to launch and run other applications and access or analyze data from a variety of sources. If you write a script in an OSA-compatible script, you can have your stacks communicate with other scriptable applications such as FileMaker Pro 2.0, Microsoft Excel 4.0, Microsoft Word 6.0, WordPerfect, Canvas version 3.5, and others.

Keep in mind that HyperCard 2.2 does not include the *AppleScript Language Guide;* if you are planning some serious work with AppleScript, be sure to purchase AppleScript version 1.1 from APDA. (See page 28 for APDA ordering information.)

Building Stand-Alone Applications

As a developer, you may not want to permit users to make modifications to your product, and a long-time mixed blessing of HyperCard was its open design, allowing anyone to open a script and modify it. So Apple created the StackToApp stack extension, which permits a developer to create a stand-alone HyperCard application.

With the extension installed, creating a stand-alone application is easy. When you choose the Save As command, HyperCard has a scrolling menu that offers three file types: stack, custom file type, or application. You can create your own file type, or select Application for a stand-alone stack that does not require HyperCard to run. The StackToApp stack translator places the HyperCard Player into your stack, minus the script editor and other nonessential resources. Note that you need to be running System 7 to create a stand-alone application in HyperCard. The HyperCard portion of such a stand-alone application takes up about 770K and permits the user to run your custom stack without the HyperCard application; the stand-alone application has no authoring features (that is, it has a default user level of 2 with a maximum user level of 4).

You should also know that it is possible to remove a lot of additional resources from a stack if you want to reduce its "footprint." Apple has not yet provided documentation for doing this, but if you are good with ResEdit, you can experiment with extracting a lot of

with any number of commercial products, such as financial packages and databases, Velick is happy keeping Art Stacks the way it is. Velick says, "If I had to start over today, with many more platform options to choose from, I would still choose HyperCard. Art Stacks is successful because it takes advantage of the visual nature of both HyperCard and the Macintosh. HyperCard is what Art Stacks customers use to run their businesses. It allows us to see the images of the art, along with all the data, quickly and easily."

Now, with an installed base of more than 100 users, Velick has a viable business with Art Stacks as his primary product and service. He assists his customers by telephone, making many minor changes to the product constantly. He provides a regular update, usually once a year, to include new features, bug fixes, and updates to the Macintosh system software and HyperCard application.

Educational Products

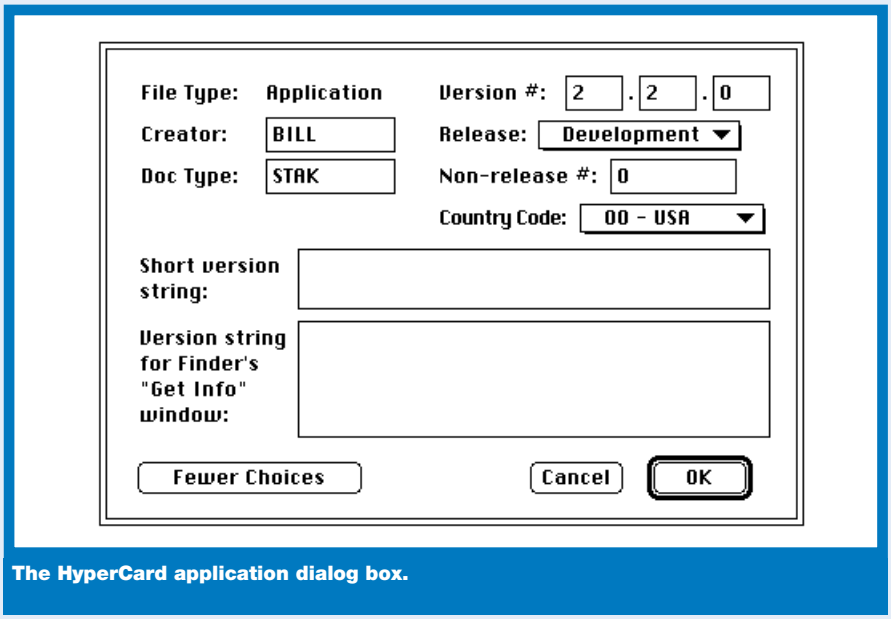
Education is perhaps the area where HyperCard is used most widely. Schools across the United States and around the world have created thousands of stacks, many of them used by just one class. Another category of educational stacks includes products that assist teachers in their work; an example of this is the product line from Peter Desberg, coordinator of the computer-based master's program at the California State University at Dominguez Hills in Southern California.

Teaching With Technology and InterActive CaseWare. Desberg has produced a number of educational products using HyperCard, for students and for teachers. Two of the most interesting are *Teaching With Technology* and *Interactive CaseWare for Teachers.* They help illustrate the range of products that can be created within one industry or field.

Teaching With Technology, published by Allyn and Bacon Publishers (a member of the Paramount group), is an interactive textbook that this year won the award from the Academy of Interactive Arts & Science for the best adult interactive book. Desberg says, "We call it 'bookware,' and it is written totally in HyperCard." The product is designed to introduce technology in a way that encourages the user to question and experiment.

Interestingly, is included with the bundle for ClarisWorks Demo; it is used to simulate a spreadsheet in ClarisWorks. It lets you try things, ask questions, and then takes you right into the application.

Another product of Desberg's is *InterActive CaseWare for*



The HyperCard application dialog box.

resources that may reduce the size of your stack by several hundred kilobytes.

WorldScript Support

HyperCard 2.2 conforms to WorldScript compatibility requirements. This means it is able to handle non-Roman text for languages that are supported by the WorldScript I and WorldScript II extensions.

As long as the relevant script and correct WorldScript extension are installed on a Macintosh computer, HyperCard 2.2 properly displays fields that contain double-byte characters, bidirectional text, or contextual ligatures. This means that HyperCard can now show languages such as Japanese, Arabic, Korean, and Chinese.

ADDMotion II

For version 2.2, Apple licensed the cel animation XCMD ADDMotion II from Motion Works and has bundled it with HyperCard. This provides animation capability for HyperCard users without requiring any other applications. When you install ADDMotion II, a button appears on your Home stack for turning ADDMotion II on and off. The HyperCard File menu is modified to include animation options that you can use to create animated stacks for integrating multimedia and sound features.

Color Tools Stack

Like ADDMotion II, Color Tools is a stack that installs a toggle button in your home stack. With color turned on, you can use the features of ColorTools, which include AddColor and other XCMDs to provide color for any stack. The Color Tools installation adds a color menu to the HyperCard menu bar and a color palette, which you can use to create color objects, place color images, and add color to objects such as buttons and fields. You can also create scripts that add color to the stack while it is running.

Registered users of HyperCard version 2.2 may freely distribute the AddColor XCMD for personal and commercial purposes if credit is given (see the Color Tools stack for details).

QuickTime Tools Stack

HyperCard's QuickTime Tools have been enhanced for version 2.2. The QuickTime Tools stack permits you to install movie buttons in any stack, select a window style and location on a card, and play the movie in a variety of ways.

Teachers—an interactive computer-aided instructional (CAI) product published on CD-ROM, also by Allyn and Bacon. It presents interactive educational case studies using QuickTime, written in HyperCard.

Desberg says, "We generate case studies, which is now a very important part of curriculum development; they can be done interactively with HyperCard. Basically, it is helping teachers improve their teaching skills using interactive multimedia." The case studies are examples of what is called "critical instances" in classrooms—a widely used concept that provides for alternative solutions to various issues, where each user can choose a hypothetical solution and follow it through interactively. Desberg explains, "Teaching by means of critical instances has been used in elementary and secondary classes, but the concept was originally intended for university education classes, as well as continuing education for teachers, called 'in service' training." Because it is based on QuickTime, it allows for more variety and more direct access to information than text-based material.

So why does Desberg use HyperCard for this kind of development? "Because the development time is so quick," he says. "There are other scripting languages, and they are good, but not as good as HyperTalk; the scripting power is much better than in other scripting languages." Desberg feels that other programming languages aren't able to produce the kinds of quick results that a developer can get with HyperCard; he explains, "particularly for me, because I train teachers, and I need results quickly. It's not feasible in the other languages like C or Pascal—it takes too long."

Desberg sees an opportunity for Apple with HyperCard, and says, "I believe that a new version of HyperCard, if it included a number of reasonable but significant improvements, could demand around $250–300 commercially. It's worth it, with its flexibility for animation, sound, and text handling. It's all the things you ever wanted in a multimedia package."

Apple's Commitment

At this point, you may be wondering where HyperCard will go in the future. As always, Apple is loath to discuss unannounced products, and HyperCard is no exception. But as a developer, when you put your product on a particular platform, you need to be assured that the platform will be there in the future, and, in fact, that it will be continually strengthened and improved.

Apple is clearly committed to supporting HyperCard. For starters, more than a year ago, Apple brought it back in-house from Claris. Today HyperCard is the second best selling product from AppleSoft, and it is the fifth most popular software product on the Macintosh computer, according to InfoCorp, Inc. And HyperCard Player is still included in all Macintosh computers sold in the United States. It is clearly a significant and strategic element of Apple's Macintosh software product line.

So you can expect Apple to continue to integrate HyperCard with strategic Apple technologies. In fact, Apple's Peter Christy says, "Apple will increasingly invest in HyperCard and other solution tools. Some of our activities are sort of obvious, such as creating a PowerPC 'native' version of HyperCard. Other directions will be less obvious and even more interesting, including the incorporation of OpenDoc technology and concepts, and, with OpenDoc, the move to support of platforms other than the Macintosh. In all our strategies, ongoing support of today's HyperCard users and developers is a key priority."

Take Another Look

If you have not used HyperCard recently, take a look at it again. Enhanced and expanded, well integrated with many other Apple technologies, not costly, and widely used, HyperCard is still one of the best development opportunities found on any platform.

Here's what comes with HyperCard version 2.2:

- HyperCard application
- sample stacks
- AppleScript run-time software
- AppleScript reference and sample stacks
- QuickTime tools stack
- Color Tools stack (including the AddColor XCMD)
- Motion Works' AddMotion II package
- *HyperCard Companion Products and Services Directory*
- *HyperCard Script Language Guide*
- *HyperCard Reference Manual*
- *HyperCard 2.0 Release Notes*
- *HyperCard Quick Reference Guide*

You can also take advantage of some of the HyperCard training resources available (books, development tools, and more), including the following:

- *The Complete HyperCard 2.0 Handbook,* 3rd ed., by Danny Goodman, (Bantam Books, 1990).
- *HyperTalk 2.2: The Book,* by Dan Winkler, Scot Kamins, and Jeanne Devoto (Bantam Books, 1992).
- Apple HyperCard User Group (meetings held at Apple Computer, R&D building 4, Cupertino, California, every third Thursday of each month. Phone 408-974-1707 for information). ♣

David Gleason is a consultant and free-lance writer based in Palo Alto, California. He is a former localization manager for Apple Computer Europe. His AppleLink address is DAVIDOVICH.