



# AppleDirections

## Inside This Issue

Editor's Note: SPA Report: More Than Meets the Eye	2
IndustryWatch: Open for Business	3
Getting Ready for Copland Developer Release: Compatibility Edition	10
QuickTime VR Royalty Eliminated, Licensing Made Easier	11
Macintosh Tools Guide Available Online	11
Apple Enhances E.T.O. and MPW Pro	12
Symantec, Microsoft Release New Development Tools	12
Apple Ships QuickTime Conferencing Kit	13
Apple Products Win Recent Industry Awards	13
Apple Announces Support for JavaScript	13
Pippin Coming Soon to a TV Screen Near You	14
CD Highlights: Tool Chest Edition	15
Human Interface: If You Build It, Will They Come?	16
The Business Case for Component Software	19
Producing "Bug-Less" Software—Part 2: Bug-Chasing Tactics	23
Developer University Schedule	27
The Internet Page	28

## Apple News

### Apple Solidifies Market Share Gains

According to two recent reports, Apple Computer, Inc. continues to hold a top share in the personal computer market.

International Data Corporation (IDC) estimates that Apple, Compaq, and Packard-Bell will end up in a near-tie for the top spot in U.S. personal computer sales for 1995, which grew 22 percent over 1994 unit sales—significantly higher than many analysts' forecasts of 15 percent growth. Recent IDC worldwide data shows Apple in the lead for sales of desktop systems in the third calendar quarter (July through September) of 1995—or Q3 95—and just behind Compaq for the same quarter in combined worldwide desktop and portable computer sales.

In its preliminary 1995 year-end report, IDC projected that Compaq, Packard-Bell, and Apple will each end up selling between 2.6 and 2.7 million personal computers in the United States in 1995. IDC data gives Compaq a slight edge, with 2.669 million units sold (11.7 percent of the market); Packard-Bell (2.656 million units or 11.6 percent) and Apple (2.636 million units or 11.5 percent) are running right behind, and IBM (1.885 million units or 8.2 percent) is placing fourth. Overall, IDC projects that 1995 U.S. computer sales will total nearly 23 million units. IDC's estimates suggest that Apple U.S. 1995 shipments

*please turn to page 10*

## Strategy Mosaic

### Putting OpenDoc to Work in 1996

#### Cyberdog Is First Compelling OpenDoc Solution

*By Gregg Williams, Apple Directions staff*

OpenDoc for Macintosh 1.0 went golden master last November—so what happens now? How does Apple plan to move the adoption of OpenDoc forward?

Apple's approach is twofold. First, Apple Computer, Inc., is encouraging developers to create OpenDoc parts, and over 300 developers have committed to delivering OpenDoc parts in 1996. (For details, see "OpenDoc 1.0 for Mac OS Goes Golden Master!" on page 1 of the December 1995 issue of *Apple Directions*.)

The second part of Apple's plan to promote OpenDoc—which is what this article is about—is to "jump-start" interest in OpenDoc by using it to create compelling solutions in Apple's key markets—home, education, entertainment, and business (especially the publishing, technical, and mobility markets). These solutions will provide immediate benefits to customers and will be open-ended, so as to present business opportunities to developers.

The first of the OpenDoc-based solutions that Apple will release is code-named *Cyberdog*. Once I've described *Cyberdog* (in the first part of this article), I'll be able to say more about Apple's OpenDoc strategy for 1996 and 1997.

*please turn to page 5*

# AppleDirections

Volume 4, 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 (SZABLA)

## Production Editor

Lisa Ferdinandsen (LISAFERD)

## Contributors

Robert Beach, Peter Bickford, Alex Doshier, Victor J. Hnyp

## Manager, Developer Press

Dennis Matthews

## Manager, Apple Developer Periodicals

Mark Bloomquist

## Production Manager

Diane Wilcox

## Prep and Print

Consolidated Publications, Inc., Sunnyvale, CA

© 1995 Apple Computer, Inc., 1 Infinite Loop, Cupertino, CA 95014, 408-996-1010. All rights reserved.

Apple, the Apple logo, APDA, AppleLink, AppleShare, ColorSync, HyperCard, ImageWriter, LaserWriter, Mac, MacApp, Macintosh, MPW, Newton, Performa, Pippin, PlainTalk, PowerBook, PowerTalk, QuickTime, and WorldScript are trademarks of Apple Computer, Inc., registered in the U.S. and other countries. AppleGlott, AppleScript, At Ease, develop, eWorld, Finder, MessagePad, OpenDoc, Power Mac, QuickDraw, and QuickTake are trademarks of Apple Computer, Inc. Adobe, Acrobat and PageMill are trademarks of Adobe Systems Incorporated, which may be registered in certain jurisdictions. NuBus is a trademark of Texas Instruments. PowerPC is a trademark of International Business Machines Corporation, used under license therefrom. UNIX is a registered trademark of Novell, Inc. in the United States and other countries, licensed exclusively through X/Open Company, Ltd. All other trademarks are the property of their respective owners.

Mention of products in this publication is for informational purposes only and constitutes neither an endorsement nor a recommendation. All product specifications and descriptions were supplied by the respective vendor or supplier. Apple assumes no responsibility with regard to the selection, performance, or use of the products listed in this publication. All understandings, agreements, or warranties take place directly between the vendors and prospective users. Limitation of liability: Apple makes no warranties with respect to the contents of products listed in this publication or of the completeness or accuracy of this publication. Apple specifically disclaims all warranties, express or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose.

## Editor's Note

# SPA Report: More Than Meets the Eye

Here's part of a letter I received from a reader last month:

*I enjoy reading Apple Directions' IndustryWatch column; it provides interesting information that helps give some perspective. Unfortunately, it's a little pollyannish in that it seems to usually include news and perspective that shine positively on Apple. However, sometimes there's news that's bad for Apple, that's important to developers. Here's an item that I'd like to see addressed:*

*In the third calendar quarter of 1995 [Q3 '95, July through September], the Software Publisher's Association [SPA] reported a decline in U.S. Macintosh software sales of 27 percent over 1994 sales for the same quarter. . . . This news comes on the heels of the news that Apple's hardware sales were up and that Apple gained market share in the hardware market during Q3 '95.*

*I know enough about marketing that sales are subject to strange influences and cycles, and that it's foolish to get too concerned over a downturn in an overall growth rate. But I think we need to hear [your] "implications/opinions" for this story.*

First off, profuse thanks for sending this. We don't get as many letters as we'd like, and it's great to know that others are paying attention to what's going out there. I encourage any of you to write with other developments—whether or not they “shine positively on Apple.” If they're appropriate, we'll look into them and share them in IndustryWatch or elsewhere.

Now, for your main point: News like this is important; SPA does a careful job collecting and analyzing data, and Apple always takes its reports seriously. But I'm glad you said that we shouldn't get “too concerned” over this one piece of news so I don't have to say it myself (and risk sounding “pollyannish”!). Looking beneath the top-level conclusions of the latest SPA report, there's a lot more than initially meets the eye. To quote the headline of a story about the SPA

report in the December 11, 1995, issue of *Computerworld*, “Soft Mac software sales are deceiving.”

The writer, Lisa Picarelle, says in the article, “Don't let the numbers fool you. Despite the very noticeable decline . . . the Macintosh market is thriving, according to industry watchers.” As you anticipated, she says the software sales downturn can be explained by timing. Third-quarter U.S. sales are usually boosted by heavy educational orders; since Macintosh systems for the educational market ship with bundled applications, schools who buy them don't immediately need to purchase additional software.

Additionally, she points out that the third quarter of 1994 was a very difficult act to follow: It was a “blockbuster quarter for Macintosh business software sales” because Microsoft shipped long-awaited native Power Macintosh versions of Word and Excel.

From this last point, you'd expect that sales of traditional business productivity products would be down, while sales in other categories would be unaffected, and the SPA data supports this analysis. Macintosh spreadsheet, database, and word processor sales were dramatically lower in the United States in Q3 '95 compared to Q4 '94, according to SPA.

Other categories, however, increased—even more dramatically from Q3 '94 to Q3 '95: U.S. Macintosh desktop publishing software sales increased 140 percent—compared to only a 22 percent increase for Windows desktop publishing software sales—while Macintosh personal information management and project management software sales increased 101 percent and 66 percent, respectively. In fact, of the 16 categories tracked by SPA for the report, sales in 10 of those categories either stayed approximately the same or increased during the period.

I think another point or two need to be considered along with the SPA data: First, keep in mind all the positive Macintosh mar-

*please turn to page 14*

## IndustryWatch: News and Perspective

# Open for Business

By the editors of Apple Directions

## Java Becoming Everyone's Cup of Tea

Suddenly Java—the programming language developed by Sun Microsystems that can turn static World Wide Web pages into interactive multimedia documents using video, animation, sound, and other data types—is on the tip of everyone's tongue.

Last month, two major developers committed to supporting Java. Metrowerks, who developed CodeWarrior, the popular Power Macintosh development environment, will provide a suite of Java development tools for the Macintosh platform starting in May 1996. Additionally, Microsoft announced that, as part of its Internet strategy, it would license Java and build support for it into its Web-browsing software.

Java, which is syntactically similar to C++ , lets you write custom mini-applications, or “applets,” that will run on any browser that supports Java, regardless of the underlying operating system. (Currently, the only Web browsers that support Java are various implementations of Netscape Navigator 2.0 and Sun's own HotJava browser.) Java code compiles to a byte-code object-code form that special Java software on the client computer knows how to interpret. Because of this, you can write a single Java applet that will run on Mac OS, Windows, UNIX®, or other computers.

Java is important to the development of the World Wide Web because it promises to free Web page authors from being limited to the data types hard-coded into a particular Web browser. An author who wants to add a new data type—a graphics animation or a new format of audio—to a Web page can do so, and Java makes it possible for the browser software to download and execute an applet that can then access the data type. Through Java applets, Web pages can cause all sorts of dynamic behavior to occur on the computers that access them.

*Implications/Opinions:* Apple recently announced its support for JavaScript, the cross-platform object scripting language that complements Java. (See the news story on page 13 for more about JavaScript). Also, the Macintosh version of Navigator 2.0 isn't expected to ship until early 1996, while HotJava for the Macintosh was announced at Macworld in January. To find out more about Java and how it might fit into your product plans, you can find a white paper about it at the Sun Web page. (See the “Internet Resources” box on page 4.)

## February Apple Directions Online

February's *Apple Directions* will be available by January 15 at the following locations:

AppleLink: path—Developer Support:Developer Services:Periodicals:Apple Directions.

Internet: <http://dev.info.apple.com/appliedirections/adtoc.html>

eWorld: in the Apple area of the Computer Center.

While we're talking about Java, we'd like to clear up one rumor that's been going around by saying that Java does not take the place of OpenDoc. To put it simply, OpenDoc is a cross-platform architecture for component software, while Java is an Internet technology that makes Web sites more versatile; it can't be used for the great many non-Internet-related tasks performed by computers.

Also, from what we've heard, Java is a terrific programming language, but it doesn't take the place of OpenDoc even for Internet application development. Java helps you develop content for Web sites and then get that content to people browsing the Web; OpenDoc—more specifically, the group of Internet-related OpenDoc parts called Cyberdog—can be used to add access to various Internet services (not just Web pages) to your applications and OpenDoc documents.

## Over 24 Million Internet Customers Served

Speaking of the Internet (and who isn't these days?), a recent study by Nielsen Media Research—the same company that tracks television watching in the United States—concluded that approximately 24 million people in the United States and Canada were already using the Internet by fall 1995. That number translates into 11 percent of the population over the age of 16. According to the study, nearly 11 million more people have the equipment that could let them use the Internet, and that 17 million people use the World Wide Web. Also, one-third of North American Internet users are women, and Internet users spend more time on the 'net than they do using VCRs—about five hours each week.

Nielsen based its study on in-depth interviews with more than 4,000 randomly selected households. Previous studies have been considered less reliable because they've tended to inflate the numbers of Internet users, and they've been based on interviewing users online.

*Implications/Opinions:* No question about it: The Internet is open for business. The study also indicates that approximately 2.5 million North Americans have made purchases over the Web, and that Internet-using households tend to be among the most affluent members of society, with annual incomes of more than \$80,000.

## Macintosh Computer Strong on the Internet

Several recent studies of Internet users indicate that people access the Internet using Macintosh computers in numbers disproportionately higher than the Macintosh share of the overall personal computer market. In a survey of Web usage, an Arizona State University researcher concluded that 17 percent of Internet users log on from a Mac OS computer, while only 14 percent use Windows systems. The survey also found that Sun Microsystems' lead as the Internet client of choice is shrinking—the study found that 31 percent of Internet users use a Sun OS-based system, down from 56 percent the year before.

Another survey, conducted by CyberSpace Japan, found that 39 percent of Japanese Web users log on with a Macintosh computer, compared to the 24 percent who use a UNIX system, 26 percent who use DOS/V, and 9 percent who use NEC. NEC's low market share is

particularly surprising given its 50 percent share of the Japanese desktop market.

Also, Mac OS computers outsell Windows machines for use as server platforms, although UNIX systems are still by far the most common choice for World Wide Web servers. Approximately 17 percent of Web servers reside on a Macintosh computer, according to several industry estimates, while it's estimated that Windows systems account for just over 14 percent of Web servers. The Macintosh leads all platforms in creating graphics and media for Web sites, with a 40 percent share of the market.

*Implications/Opinions:* Apple has a strong position among Internet users, but that position is being challenged. The survey results don't reflect the impact of Windows 95 or Microsoft's Internet server strategy. Apple is working intently to hold and improve on that position by shipping products such as the Apple Internet Connection Kit—an easily installed, low-cost bundle of popular Internet-surfing software, including Netscape Navigator, Claris eMailer, Fetch, and more than half a dozen other products. In a recent review of the Apple Internet Connection Kit for the *New York Times*, Peter Lewis said that, because of the kit, "Clearly, for me, the Macintosh was the best way to gain access to the Internet."

On the server side, Apple is enhancing its position as the provider of the easiest-to-use Internet server solution by bundling with its servers software such as the Adobe™ PageMill authoring tool for Web sites. (Incidentally, PageMill—which is available only for the Macintosh computer—shipped more than 30,000 copies in its first month of availability, according to Adobe.)

You can do your part by continuing to deliver top Internet access products for the Macintosh computer and spreading the word that the Macintosh computer provides easy access to the 'net.

### **New PowerPC Chip to Be Exponential-ly Faster**

In December, the startup semiconductor company Exponential announced that it will deliver PowerPC processor-compatible microprocessors with clock speeds over 300 MHz as soon as early 1997. Subsequent versions could run as fast as 600 MHz. When the chips are available, Exponential expects them to be the world's fastest personal computer microprocessors. The company has raised more than \$14 million in venture capital—part of it from Apple Computer—to fund development of the new chips.

To give the chips their incredible speed, Exponential is using so-called BiCMOS technology, a combination of bipolar logic used primarily in mainframe chips and standard CMOS technology employed in many microprocessors. Intel says that the Pentium chip uses BiCMOS technology, but its use of bipolar logic has been limited because the Intel implementation of it is costly, uses a great deal of power, and requires a large die size. Exponential says it will get around these hurdles to delivering a chip that makes extensive use of bipolar logic by using several patented innovations in process technology.

*Implications/Opinions:* We think this should dull some of the cheers coming from the Intel camp about having "leapfrogged" the PowerPC consortium—that is, Apple, IBM, and Motorola—with the Pentium Pro chip. Intel announced it would immediately begin shipping 200-MHz Pentium Pro (formerly called P6) processors with a SPECint rating of

more than 350. This performance compares with a SPECint rating of about 200 for the fastest PowerPC 604 processor currently shipping, and comes close to the integer performance of DEC's Alpha processor, currently the fastest RISC processor on the market.

However, no major PC vendor has announced or made available a desktop system based on a 200-MHz Pentium Pro processor and, despite heavy advertising by a number of vendors for 150-MHz Pentium Pro systems, we don't know of one that is likely to ship in any volume in the near term. Also, its high clock speed and SPECint rating aren't necessarily indicative of what Pentium Pro processors will do for most users: Intel itself admits that Pentium Pro chips have trouble with 16-bit software, running Windows 95 only marginally more quickly than previous Pentium designs and running Windows 3.x more slowly than earlier chips. Windows NT can take advantage of Pentium Pro processing speed; so far there are very few 32-bit Windows 95 applications.

When they're available, Exponential chips aren't expected to have any trouble running Power Macintosh software, which is all 32-bit clean. As of early 1996, developers were shipping more than 1,400 "native" Power Macintosh applications. Also, even before systems based on Exponential chips are available, Apple will build Power Macintosh computers based on the soon-to-be-shipped PowerPC 604e processor, which has a SPECint rating of 250 running at 150 MHz. In its December 1995 issue, *Byte* magazine said about the 604e chip that its "... speculative execution and branch prediction logic, and the improved load/store instruction performance, should endow a desktop system with processing power beyond that of any system based on Intel's new P6." The 604e chip is expected to be available in quantity by mid-1996.

### **Newton Platform Gaining Recognition**

The Newton MessagePad has received a substantial number of favorable mentions lately, including the Best of Show award in the operating system category at COMDEX. Previously skeptical industry analysts, including Stewart Alsop and Kim Brown, are now saying that Newton deserves a second look. Most of these positive reviews are based on the improved connectivity, better performance, and more powerful developer tools that are part of the 2.0 version of the newly released Newton OS.

*Implications/Opinions:* The renewed optimism about Newton is based on solid improvements to the product and a focus on vertical markets—such as sales automaton, health care, transportation, and financial services—rather than the general consumer market. Newton is also benefiting from the fact that high-profile competing products have either quietly faded away (in the case of Casio's Zoomer) or suffered major setbacks (like Microsoft's WinPad initiative). ♣

## **Internet Resources**

- At the Sun Microsystems Web site ([http://java.sun.com/whitePaper/javawhitepaper\\_6.html](http://java.sun.com/whitePaper/javawhitepaper_6.html)), you can find the white paper on Java.

## Strategy Mosaic

## Cyberdog

continued from page 1

## The Three Heads of Cyberdog

Cyberdog is difficult to describe accurately because it is different things to different people:

- *Internet Explorer.* This is a solution (you can't call it an application because it's actually an OpenDoc document) that is immediately useful to anybody. It gives the user an easier, more consistent, and more intuitive way to search and browse the Internet, access Internet mail and newsgroups, and access Internet content.

- *The Cyberdog architecture.* Even though Internet Explorer is definitely about the Internet, the Cyberdog architecture isn't limited to the Internet. Instead, Cyberdog has an architecture that tightly integrates network services into the Mac OS. It just so happens that, to show how useful the Cyberdog architecture is, Apple used it to create an Internet solution (more on that later). In short, the Cyberdog architecture provides a flexible, extensible framework for integrating remotely located information into your software and documents.

You can use the resources available through Cyberdog to give network connectivity to existing container applications or OpenDoc parts. In addition, you can make business opportunities for yourself by adding new parts that extend Internet Explorer or by creating enhanced replacements for existing Internet Explorer parts.

- *A toolbox of parts for creating custom Internet solutions.* Internet Explorer is implemented as a collection of (Cyberdog-aware) OpenDoc parts. Because you can embed any kind of OpenDoc part inside an OpenDoc

document or a container application, both users and developers can use the Cyberdog-aware parts that come with Internet Explorer to create OpenDoc documents that have network connectivity.

## Life Before Cyberdog

Well over a year ago, the popularity of the Internet had begun to skyrocket, but using the Internet—even on Mac OS computers—was too difficult for many users. Apple engineers decided to tackle the problem of making the Internet easier to use, and they further decided to use OpenDoc as the enabling technology. (The choice of OpenDoc was a fortuitous one. Internet software is evolving so quickly that monolithic applications can't be revised quickly enough to meet users' needs; however, Apple's OpenDoc-based Cyberdog will give Apple and third-party developers the ability to evolve their Internet-related products faster than competitors selling monolithic applications can.)

Apple engineers spent five months doing user testing to see how people used Internet access tools and what problems they had using them. They found several problems:

- *Human interface incompatibilities.* Web browsers, Gopher programs, and FTP programs (for example) all have different ways of doing things—specifying where on the Internet you want to go to get your information using site addresses (also called *URLs*, or *universal resource locators*), retrieving information, and saving addresses of interest for later use. Furthermore, these different programs constitute isolated worlds—you can't use one program to access an address saved by a different program.

- *Too much typing.* Internet access centers around site addresses—long alphanumeric

sequences like “http://www.info.apple.com/dev/appledirections/adtoc.html.” Sure, most Internet access programs allow you to save these addresses so you don't have to retype them, but the only way to use the same address in another Internet access program or to send an address to another person is to type the address again. With these programs, there is no way of making an address into an object (like a System 7 alias file) that can be saved, copied, attached to e-mail, and so on.

- *Monolithic Web browser applications.* As good as they are, browsers are far from perfect. Your content almost always shows up in one window—very un-Mac-like. When it doesn't—that is, when the browser calls up a helper application to display a PICT file or a QuickTime movie, for example—the user interface is weakened. (For example, users may close the helper application and end up in the Finder, with the browser hidden under Finder windows.)

A monolithic browser slows and limits innovation. You have to wait for the next revision of the browser application (which takes at least six months) to see new features added. Though you can do a lot with helper applications,

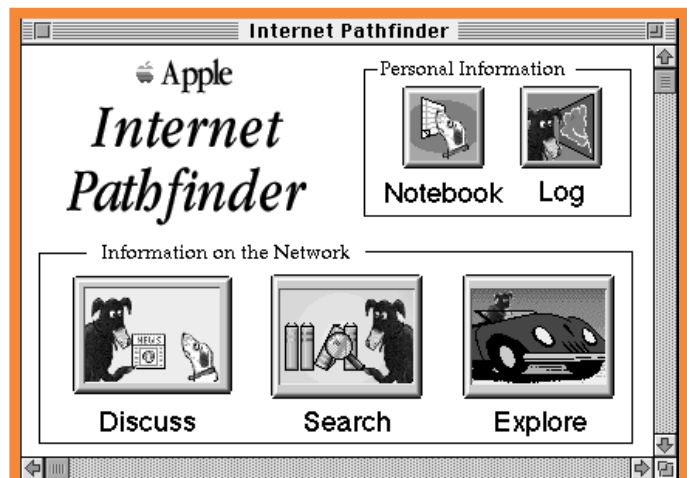
you are also forced to accept certain fundamental innovations that only the browser's vendor can add. What's worse, those innovations and helper applications are limited to the browser application—they aren't accessible from any other Internet access program.

## Internet Explorer

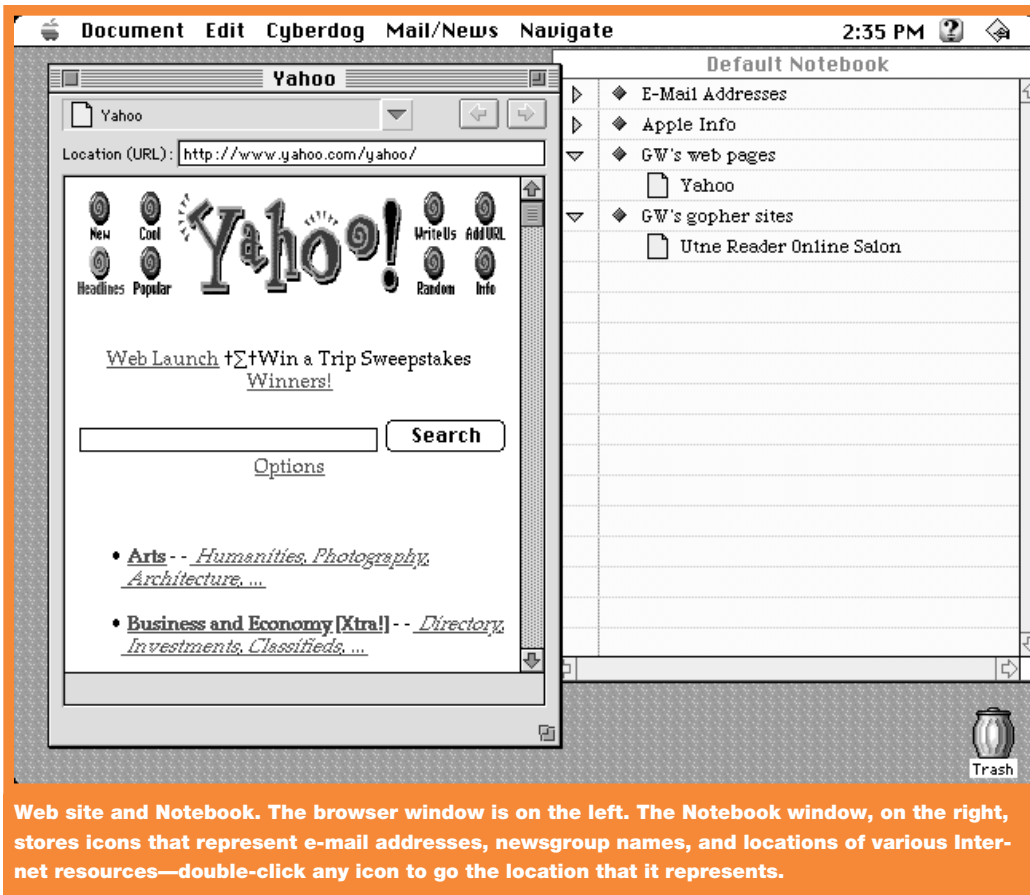
This partial tour of Internet Explorer won't show more than a fraction of the features of Cyberdog, but it should be enough to get you excited about Cyberdog and to show you how Apple has addressed the factors mentioned earlier.

The figure “Internet Pathfinder” on this page shows the main screen you see when you open the Internet Pathfinder icon—which is actually an OpenDoc document. (The figures in this article are from an early version of the product; the shipping version will have different graphics and may be structured differently.)

The bottom three buttons offer users three different ways of interacting with the Internet. The Discuss button gives you a browser for reading Internet newsgroups. The Search button gives you a one-field form asking what you want to search for; it returns a list of everything it found on the



**Internet Pathfinder.** This window (which is actually an OpenDoc document) is the interface through which most users will begin exploring the Internet.



**Web site and Notebook.** The browser window is on the left. The Notebook window, on the right, stores icons that represent e-mail addresses, newsgroup names, and locations of various Internet resources—double-click any icon to go the location that it represents.

Internet—including Web pages, Gopher sites, and newsgroups. (Actually, the search form packs the query off to a Web query page, collects the results from it, and presents them using the Cyberdog interface.) Finally, the Explore button opens a Cyberdog Web browser part that shows an Apple-maintained Web page of pointers to various encyclopedic Web pages.

The top two buttons, Notebook and Log, are two elements that Apple has added to improve the user's Internet experience. The Notebook provides a place for you to store references to all the locations on the Internet that you might want to visit again. (You can have as many Notebooks as you wish—after all, they're just OpenDoc documents.) The Log provides a simple but useful way of looking at the locations you've visited recently. The Notebook and Log windows are always available, regardless of

what part of Internet Explorer you are using.

The figure above shows the result of visiting the Yahoo Web site (through the Explore button) and opening the Notebook. (Note that these results appear in separate windows—very Mac-like.)

### Locations as Icons

One of the major innovations of Cyberdog is that it allows places on the Internet (actually, their addresses) to be manipulated as icons (also called *CyberIcons*); in future versions of Cyberdog, the artwork for a given CyberIcon will tell you what kind of a service it belongs to.

See the Yahoo document-with-folded-corner icon just above the "Location (URL):" label in the Yahoo browser window? By dragging that over to the Notebook window, I was able to add Yahoo to my "notebook" of Internet locations. (If I had dragged the Yahoo "document" onto the desk-

top, I would have created a Yahoo icon on the desktop.) If I open my Log window, I see the Yahoo CyberIcon there as well. (See the figure "The Log window" on page 7.)

CyberIcons behave as you would expect Mac OS icons to. You can click on them in the Notebook and Log windows—which causes the appropriate locations to open. You can also perform a Get Info command on them—which causes an Info window to appear that gives you the location's address.

You can do these same things with the Finder version of a CyberIcon. You can (of course) move such an icon to a folder or copy it onto a floppy disk to give to someone. If you drag it to a text field that supports Macintosh Drag and Drop (such as the Location field of a Netscape Web browser window), the computer will enter the location's address into the text field. Wow!

The Log window also helps users (especially novices) navigate the Internet. "Where am I?", "How did I get here?", and "How can I get back to something I saw earlier?" are common questions that new users often ask. The Log window can help in such cases. Using the pop-up menu in the lower-left corner, users can have the window's contents displayed hierarchically (as shown in "The Log Window" on page 7), alphabetically, or chronologically. Of course, you can double-click on any Log entry to go to that location, and you can drag Log entries as described earlier.

### The Mail's In!

Internet Explorer (and Cyberdog) allow you to send and receive mail. The mail feature conforms to the MIME (Multipurpose Internet Mail Extensions) standard, which means you can send all sorts of things in your Cyberdog mail—including Cyberdog icons. (See the figure "Cyberdog mail" on page 7.)

Here are a few of Cyberdog mail's many interesting features. You can drag address entries between your Notebook and the "To:" and "From:" fields of a Cyberdog mail message. You can define "trays" that incoming mail automatically gets sorted into when it meets certain criteria. Finally, the content of all the mail and newsgroup messages you store locally is automatically indexed by an Apple-developed indexing engine that returns results ranked by relevancy; this allows you to easily and quickly find messages you've stored.

### More Interface Standardizations

The Connect To and Preferences menu commands make the Internet easier to deal with by standardizing the ways you go to specific locations and set preferences. Before Cyberdog, each Internet-access program had a different place for you to specify

the address of the location you wanted to access and a different way of capturing the program's preferences.

By using a single menu command and a paneled approach (see the figure "The Internet Preferences window" on page 8) for preferences and connection information, Cyberdog presents a consistent human interface that is common to every Internet service. Following an "invent as little as possible" philosophy, the Cyberdog preferences scheme is built to work on top of Internet Config, an extension that many Internet access programs use to facilitate the storage and recall of often-repeated information (your name, e-mail address, and news host, for example).

If you decide to add a new service to Cyberdog, integrating it into the Preference and Connect To dialogs is no problem. As part of the new service creation process, you must provide an icon for that service and two OpenDoc parts that implement the Preferences and Connect To panels for your service. The Cyberdog architecture then includes them when a user selects either the Preferences and Connect To menu commands.

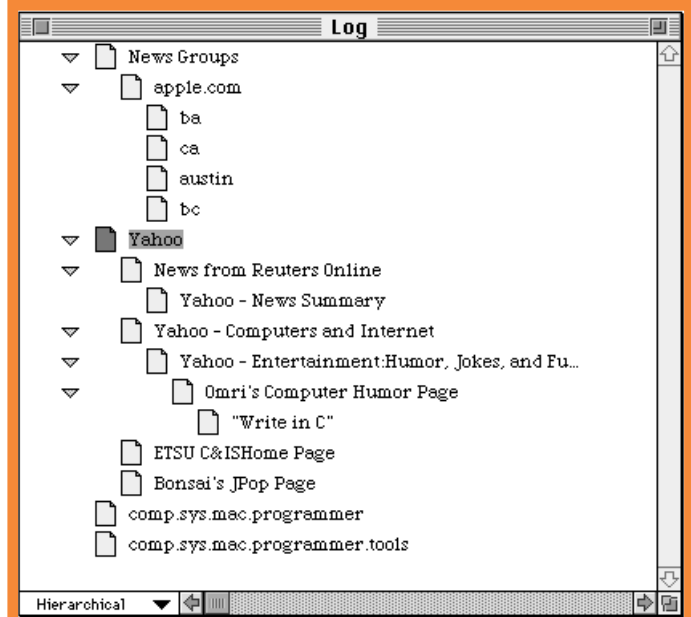
**Cyberdog = Component Software for the Internet**

As you can tell from my description so far, the distinction between Internet Explorer and Cyberdog can become unclear. Perhaps it would help to recall that Internet Explorer is just an OpenDoc document. The Explore button, for example, is just a button (an OpenDoc part called a *CyberButton*) that opens a Web browser part "tuned" to a certain Web page when you click it.

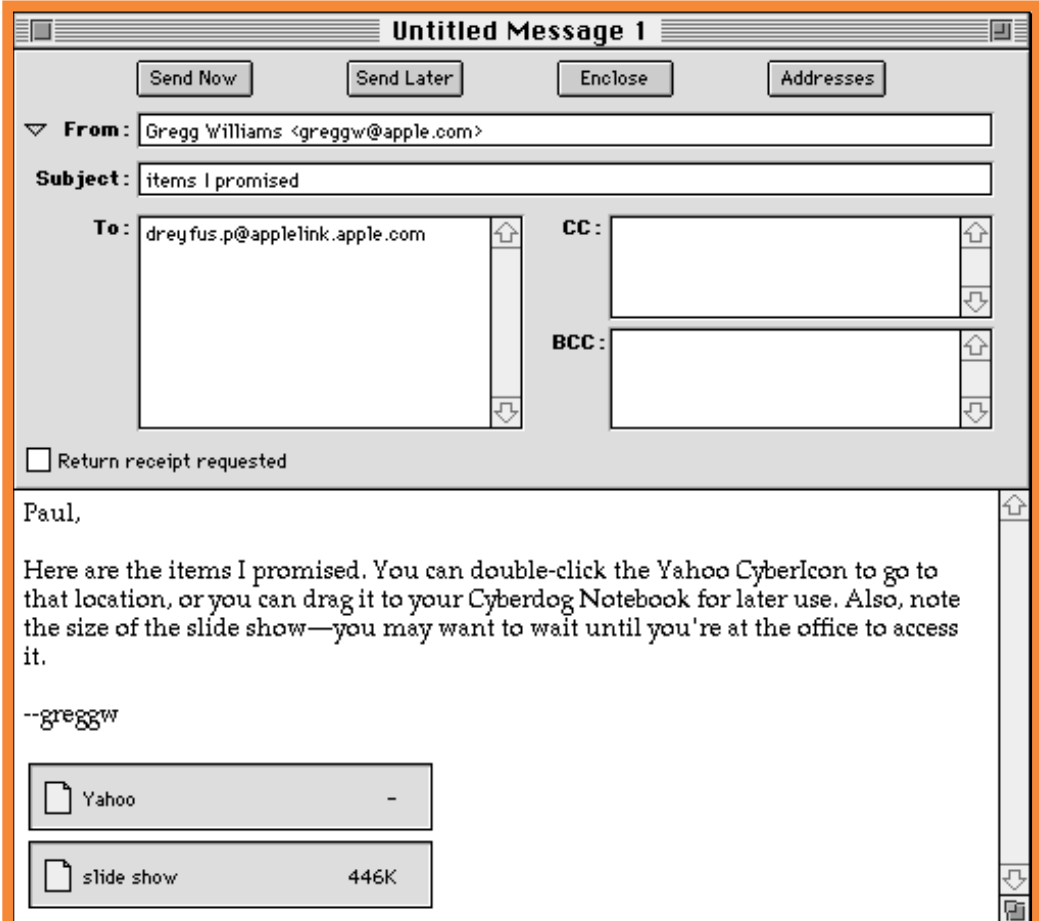
To prove to myself that Cyberdog lives within the OpenDoc environment, I created an OpenDoc document and added an instance of the famous OpenDoc "ticking clock" part to it. I then

dragged a blank CyberButton into the document, dragged a party-hat PICT from the Scrapbook onto the button (which put the artwork on the face of the button), and then dragged the Yahoo CyberIcon onto it (which gave it a behavior that is invoked when the user clicks the button).

The result is shown in the figure "An OpenDoc document with a CyberButton" (page 8). The clock keeps perfect time (even when the document isn't active), and when you click the party-hat button, the Yahoo Web page opens. Not bad for an "application" that I created in under five minutes! (P.S.—Do you want to be able to do this with your products? This is one of many good reasons to make your application



The Log window. This shows you where you've been in your current Internet session. You can also arrange the locations listed here chronologically or alphabetically.



Cyberdog mail. Since Cyberdog mail is MIME-compliant, you can enclose icons that represent Web sites and other Internet locations.

into a container application or to make the full transition to OpenDoc parts.)

So what does Cyberdog deliver that's better than today's Internet access? Simple—together, Cyberdog and OpenDoc deliver component software for the Internet. A monolithic browser, obviously, suffers from the same limitations and rigidity that plagues monolithic applications. A Cyberdog solution benefits from the advantages of component software.

With Cyberdog, Apple and interested third-party developers can deliver as good a browser as a monolithic application can. But the following are also true: You can design and sell replacement parts for the browser. You can also deliver new Internet services through components—without having to build the infrastructure for it. Customers will be able to add Internet services (including the service you provide) anywhere they want. These are all benefits that accrue because Cyberdog is component software for the Internet.

### Developing With Cyberdog

Unfortunately, discussing the internal workings of Cyberdog is beyond the scope of this article.

Let it suffice to say that Apple has created an extensible, open architecture that you can add to or improve upon. Here are some development opportunities that you may decide to pursue:

- *Replacing Cyberdog parts.* When Apple engineers developed the various parts of Cyberdog—the Web and Gopher browsers, the Notebook, the Log, and other parts—they did so with the intent of providing parts that would be of immediate use to customers but that would, ultimately, be improved upon. As one Apple engineer put it, “We could not be happier if, a year from now, all the Cyberdog parts we built were replaced by much more exciting third-party Cyberdog parts.”

- *Adding new services to Cyberdog.* A lot of Internet services—especially interactive services such as videoconferencing and real-time “chat”—could be added to Cyberdog. And remember, any service that's added to Cyberdog can also be dragged into any OpenDoc document.

- *Customizing Cyberdog for non-Internet networking.* As I said earlier, Cyberdog helps integrate network services into the Mac OS. Perhaps there's money to be made by creating optimized Cyberdog

parts for certain transport mechanisms—an all-Ethernet network, for example—or certain audiences, such as K–12 schools or newspaper companies. I'm sure you can come up with possibilities that no one else will think of.

- *Adding Cyberdog to existing applications.* Cyberdog is about finding and retrieving remotely located data. You might decide, for example, that it makes sense to add Cyberdog access to your high-end image manipulation application. That way, your customers could browse a database of clip art without leaving your application.

- *Turning helper applications into Cyberdog parts.* If you do this, not only will your helper application be more closely integrated into the Mac OS, but it will also be useful in more places—potentially increasing your sales.

- *Analysis tools for data.* Once users find the data they want on the Internet, they'll need tools to “clean” the data up and manipulate it. There'll be a market for software (packaged as OpenDoc parts, of course) to do that.

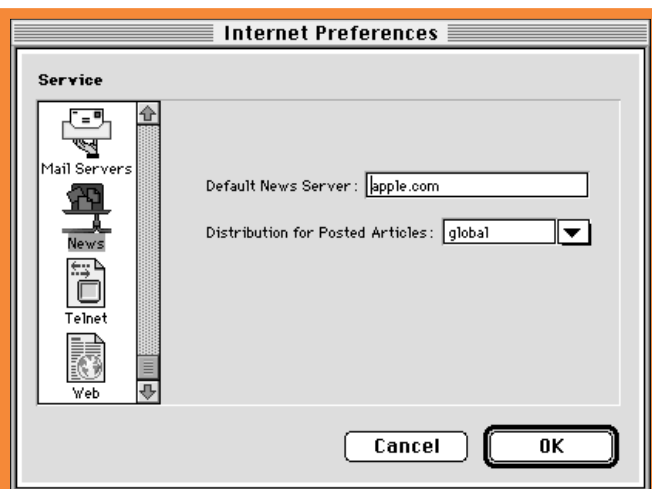
- *Creating innovative front-ends to online services.* An online service or electronic bulletin board accessed through Cyberdog will be easier to use; because it's based on OpenDoc, your customers will find more uses for it than you will be able to imagine.

### Cyberdog Availability

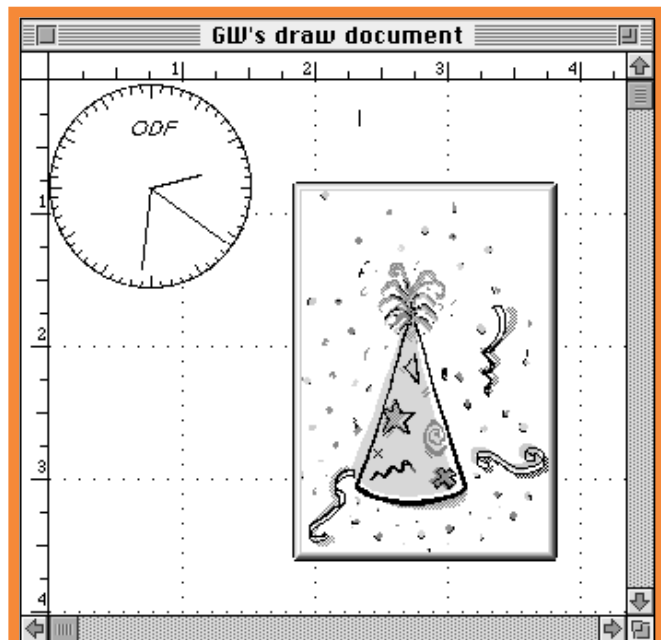
A commercial-quality version of Cyberdog will become available in mid-1996. For the moment, you can get a development version of Cyberdog from Apple's OpenDoc Web page (see the box on page 9 for the site address). If you attended Macworld San Francisco, Cyberdog is also on the OpenDoc DR4 CD, which was handed out at the Developer Central booth. (You can also receive the OpenDoc Developer Release CDs by mail; see the box on page 9 for details.)

### Apple's OpenDoc Strategy

Now that you know what Cyberdog is, I can talk about Apple's overall strategy for promoting OpenDoc. OpenDoc represents a major paradigm shift from



The Internet Preferences window. Cyberdog makes the Internet more usable by providing a single interface for preferences. New services that are added to Cyberdog will appear in the scrolling panel on the left.



An OpenDoc document with a CyberButton. With OpenDoc, you can embed Internet access in a document. This CyberButton, which I modified by dragging content on top of it, displays a Web site when you click it.



monolithic to component-based software, one that will enable developers to create more powerful and easier-to-use software than is possible today. It would be less than realistic to expect this transition to occur overnight, so Apple intends to drive OpenDoc acceptance in two ways: by promoting the development of commercial OpenDoc parts, and by working to jump-start interest in OpenDoc (so as to improve the market that you will be selling into).

I covered the details of doing commercial OpenDoc development in the December news story mentioned at the beginning of this article. In the rest of this article, I'll elaborate on Apple's strategy for jump-starting interest in OpenDoc: Apple expects to promote OpenDoc by using it as the underlying technology to specific solutions that will provide both immediate and

long-term benefits to users and additional business opportunities for developers.

I hope that I've convinced you that Cyberdog is an exciting new technology in and of itself. However, Cyberdog is also important for several reasons:

- Cyberdog is Apple's first OpenDoc-based solution. As such, it demonstrates "proof of concept" and points to future solutions that will follow the same model.
- Cyberdog demonstrates the value of component software as a way to add value to the Internet. People who saw Cyberdog at last November's COMDEX Las Vegas were very enthusiastic about it, and the open, component-based nature of Cyberdog will make it a level platform that will allow developers to innovate faster than they could working through monolithic applications.
- Cyberdog proves the value

of OpenDoc as a development direction. Many people have held the mistaken belief that OpenDoc had nothing to offer them if the software they were using (or developing) had nothing to do with documents. Cyberdog clearly demonstrates the advantages of component software outside the realm of compound documents.

### Apple to Use OpenDoc Internally

One indicator of a technology's importance to a company is whether they're using it internally for their own projects—or, as some industry people put it, "Will Apple eat its own dog food with OpenDoc?"

The official answer to that question is *Yum, yum—you bet!*

Apple plans to deliver several other solutions for target markets in 1997. Apple's target markets include home, education, entertainment, and business (especially the publishing, technical, and mobility markets). Apple's OpenDoc-based solutions will cover one or more of these target markets.

But Apple is also planning even more fundamental uses of OpenDoc. Believe it or not, the next major revision of AppleShare, to be delivered in 1996, will be built on top of OpenDoc. Developers who modify their network-based applications to operate as OpenDoc parts and be compliant to the new AppleShare interface will have products that look like they're part of AppleShare, and they will be able to take advantage of AppleShare services instead of having to write their own. Application-based network services will continue to work as they do today.

In addition, Apple is investigating other uses of OpenDoc at the system-software level (no details are available yet)—and OpenDoc will definitely be integrated into the Copland operating system.

### Getting Customers to Use OpenDoc

Apple is doing several things to get customers to use OpenDoc by mid-1996.

Granted, you can license OpenDoc to include with any OpenDoc parts you sell, but it never hurts to make the process more streamlined. So, in the second quarter of calendar year 1996, Apple plans to install OpenDoc on Power Macintosh configurations for which it makes sense. (When it is inactive, OpenDoc consumes no memory or processor cycles, so adding it to Power Macintosh computers won't decrease their performance for customers who aren't—yet—using OpenDoc.) As the months pass, Apple will be able to say that an increasing number of Mac OS computers—especially in the business and education markets—are "OpenDoc-ready."

Cyberdog should become available in mid-1996, and that will certainly give many customers an incentive to start using OpenDoc. In addition, in the summer time frame, Apple plans to begin shipping with each OpenDoc-ready computer a set of OpenDoc parts that will include a basic container part, viewer parts for key Apple technologies (such as QuickTime, QuickTime VR, and QuickDraw 3D), and some basic editors (certainly a text part and probably others). Apple will also make the viewer parts for key Apple technologies directly available to developers.

The parts that will ship with OpenDoc will provide several benefits:

- It will make OpenDoc immediately useful and will whet customers' appetites for third-party parts.
- It will allow Apple to put its new technologies into customers' hands more quickly—once Apple creates a viewer part for the new technology, customers will be able to add it

## Resources for Cyberdog and OpenDoc

- The Cyberdog Web page (<http://www.cyberdog.apple.com>) is the place to go for information on Cyberdog. If you are an Apple Associate or Partner, you can download a developer-release version of Cyberdog. Apple will release a subsequent version of Cyberdog to the general public somewhat later. Cyberdog currently requires a PowerPC processor-based, Mac OS-compatible computer with 16 MB of memory.
- At the OpenDoc Web page (<http://www.opendoc.apple.com>), you can download OpenDoc for Macintosh 1.0 and numerous sample OpenDoc parts. The page also contains pointers to various other OpenDoc-related sites.
- The OpenDoc DR (Developer Release) 4.0 CD contains version 1.0 of OpenDoc for Macintosh, a developer-release version of Cyberdog, over 40 third-party OpenDoc parts, and documentation, code, and other resources you can use to start developing OpenDoc parts. Apple Associates and Partners will automatically get this CD in the February monthly mailing. You can also get a copy by sending a request to Internet address [opendoc@apple.com](mailto:opendoc@apple.com).
- IBM's Web page (<http://www.austin.ibm.com/developer/objects/od1.html>) is a starting point for information on IBM's OpenDoc efforts.

to their documents.

- It will give you a default environment for your OpenDoc parts, including a container part and basic data parts. Since this set of parts will be shipped with every OpenDoc-ready Mac OS computer, you will be able to count on support for basic data types being present. For example, make your OpenDoc part able to contain other parts, and you'll get text labels and fields for free!

- It will provide very basic levels of functionality, giving you the opportunity to sell full-fea-

tured replacement parts that will meet more of your customers' needs.

- Finally, you will be able to study these parts as examples of good part design, to help you as you design your first OpenDoc parts.

**OpenDoc in '96!**

Certainly, 1996 is the year that OpenDoc will begin to take off! (And not just for Mac OS computers—check out IBM's plans for OS/2, AIX, and Windows 3.1, 95, and NT. (See the box on page 9

for the address of IBM's OpenDoc Web page.) Cyberdog is a very exciting new technology—the first “killer app” for OpenDoc, actually—and it's just the beginning of Apple's use of OpenDoc. If you are an Apple Associate or Partner and are interested in creating Internet-related products, download a copy of Cyberdog from the site listed in the box on page 9.

So what does all this mean for you? Well, it's an old story, but it's one that's still true: As Apple puts increasing emphasis on OpenDoc, the earliest developer

OpenDoc adopters will stand to profit far more than those who enter the market later. With this article, I've given you the outlines of Apple's future support for OpenDoc—with, of course, more details to come as they become available—and you have some decisions to make. We hope you'll join Apple and other fellow developers in contributing to the success of this exciting new technology. ♣

**Apple News**

*Market Share Gains*

*continued from page 1*

grew 22 percent over 1994 shipments—the same pace as the overall industry. For additional data, see the chart “U.S. Personal Computer Sales, 1995 (Projected)” on this page.

In worldwide data for Q3 '95, IDC places Apple first in desktop computer sales, with 1.123 million units shipped (a 9.5 percent share of the market), ahead of Compaq (1,050 million units, or 8.9 percent), IBM (916,000 units, 7.7 percent), and Packard-Bell

(802,000 units, 6.8 percent). Apple desktop sales grew 15 percent faster than the overall industry worldwide during the period, registering a 31 percent increase from Q3 '94 while the overall industry grew at a 27 percent rate. Because of the PowerBook 5300 recall, Apple slipped to sixth place in Q3 '95 worldwide portable computer shipments. In combined Q3 '95 worldwide desktop and portable sales, Apple came in second (1.27 million units, or 9 percent) behind Compaq (1.37 million, 9.7 percent). For additional data, see the chart “Top Ten Worldwide Personal Computer Vendors, Q3 1995 (Actual)” on page 11.

**Getting Ready for Copland Developer Release: Compatibility Edition**

The following is a message from Apple's Copland development team:

Apple is making significant progress in the development of the first widespread seed of Copland, which will be available to all developers in spring 1996 (for our readers in the Southern Hemisphere, that's fall 1996). We want to keep you informed of the progress we are making and inform you of an important consideration related to hardware purchases you may be planning.

As you know, Copland replaces the entire I/O infrastructure of the Mac OS with a robust, flexible, high-performance architecture that requires new drivers for all hardware devices, including Power Macintosh systems. In order to provide more functionality and a qualified developer release, we may need to make some trade-offs in supporting all

Power Macintosh configurations with the Copland Developer Release: Compatibility Edition, coming in spring 1996.

In the spring Copland developer release, we fully hope to support all NuBus™-based and PCI-based Power Macintosh CPUs, but can only guarantee at this time support for the following models: Power Macintosh 6100/60, 6100/66, 7100/66, 7100/80, 8100/80, 8100/100, and 8100/110 computers in both standard and AV configurations; and Performa 6100 series computers.

Many of you are upgrading to the most recent PCI-based systems, and we want to encourage you to keep some of the above-mentioned systems for your initial Copland compatibility testing and development. We will keep you informed if there are any changes to the supported configuration set.

Subsequent Copland developer releases will support a greater number and variety of systems from both Apple and Mac OS licensees. The final release of Copland will support all Apple and Mac OS licensee's PowerPC processor-based systems, as well as systems that conform to the PowerPC Platform (CHRP), and we are working as hard as we can to meet these goals as soon as possible.

**U.S. Personal Computer Sales 1995 (Projected)**

	1995 units*	1994 units*	1995 Market share
Compaq	2.669	2.198	11.7%
Packard Bell	2.656	2.125	11.6%
Apple	2.636	2.150	11.5%
IBM	1.885	1.640	8.2%
Gateway 2000	1.175	.960	5.1%
Dell	1.145	.788	5.0%
Hewlett-Packard	1.023	.445	4.5%

\*millions of units

Source: International Data Corporation

**Top 10 Worldwide Personal Computer Vendors  
Q3 1995 (Actual)**

	<b>Q3 '95 units*</b>	<b>Market share</b>	<b>Q3 '94 units*</b>	<b>Market share</b>	<b>Year-to-year growth</b>
1 Compaq	1.370	9.7%	1.203	10.6%	14%
2 Apple	1.270	9.0%	1.020	9.0%	25%
3 IBM	1.150	8.1%	1.072	9.4%	7%
4 Packard Bell	.802	5.7%	.630	5.5%	27%
5 NEC	.684	4.8%	.407	3.6%	68%
6 HP	.555	3.9%	.213	1.9%	161%
7 Dell	.483	3.4%	.310	2.7%	56%
8 Acer	.432	3.0%	.325	2.9%	33%
9 Fujitsu/ICL	.415	2.9%	.140	1.2%	196%
10 Toshiba	.340	2.4%	.288	2.5%	18%
<b>Total market</b>	<b>14.185</b>	<b>100.0%</b>	<b>11.354</b>	<b>100.0%</b>	<b>25%</b>

\* millions of units

Source: International Data Corporation

**QuickTime VR  
Royalty  
Eliminated,  
Licensing Made  
Easier**

Developing with QuickTime VR—Apple’s technology for creating virtual reality movies and objects—just got cheaper. Apple is no longer charging you royalties for distributing applications created using QuickTime VR. Now, the only fee involved for use and distribution of the QuickTime VR Authoring Tools suite is the cost of buying the authoring tools development kit (U.S. \$495). A license agreement is still required before you can distribute the QuickTime VR run-time software as a player or in an application.

Additionally, Apple announced the following changes to its QuickTime VR licensing policy:

- You can now license QuickTime VR run-time technology with applications you distribute through the Internet. The first company to license this technology, Time Online, is distributing its World Series baseball title at the *Time* magazine World Wide Web site with the QuickTime VR run-time engine embedded in a MacroMedia Director-based application. See the “Internet Resources” box on page 14 for the location of the *Time* magazine Web site.

- The QuickTime VR Player (for Macintosh or for Windows) is also freely available for licensing for distribution on CD-ROM, although Apple reserves electronic distribution rights to the player, which will remain available online only from Apple’s QuickTime VR Web site.

- A distribution license agreement from Apple is no longer required for you to include QuickTime VR panoramic or object movies on Web sites and CD-ROM discs. You can now electronically disseminate QuickTime VR without including the

QuickTime VR run-time software. If your Web site includes QuickTime VR panoramic and object movies, Apple recommends that you build in a link to Apple’s QuickTime VR Web site so viewers can easily download the QuickTime VR player. The QuickTime VR Web site is currently the sole electronic distribution source for the QuickTime VR player for Macintosh or Windows. The QuickTime VR Web site includes installation instructions along with other interesting examples, and helpful information about using the software.

Content developers can now distribute the following without paying a royalty (although a distribution license agreement is still required):

- QuickTime VR panoramic and object movies
- An application embedded with the QuickTime VR run-time technology
- Apple’s QuickTime VR Player, or simple viewer application, on CD-ROM

Also, Apple recently lowered the price for the QuickTime VR Authoring Tools Suite bundled with MPW Pro from U.S. \$695 to U.S. \$595. The stand-alone QuickTime VR Authoring Tools Suite remains at U.S. \$495. The price for the three-day Developer University training course “Multimedia Development with QuickTime VR” has been decreased to U.S. \$900 (from U.S. \$1,500). For information about the Developer University course, see the Developer University schedule on page 27 of this issue, or call the Apple Developer Hotline at 408-974-4897.

QuickTime VR is available through APDA; see page 32 for APDA ordering information. For licensing information, contact Andrew Hammond at Apple (408-862-0576). Additional information about QuickTime VR is available at the QuickTime VR World Wide Web site; for its location, see “Internet Resources” on page 14.

**Macintosh Tools  
Guide Available  
Online**

Think of the *Guide to Macintosh Development Tools* as the “Smart Yellow Pages” of Macintosh tools. This 60-plus page document provides you with overviews of more than 200 Macintosh tools, from component frameworks to “middleware” to development tools for Web pages. Using this document, you can quickly learn about the Apple and third-party tools available to help you perform a specific set of programming tasks. Then, using the Web site listing at the back of the document, you can download data sheets and demos (if available) for the most promising selections.

This second edition of the *Guide to Macintosh Development*

*Tools* features two new sections: "The Transition to OpenDoc" and "Internet Development." In the OpenDoc section you'll find succinct descriptions of OpenDoc tools, technologies, benefits, and resources, as well as information on when your favorite tools will support OpenDoc development. The Internet section provides you with the most complete listing of Macintosh Web page development tools anywhere, including some soon-to-be-released Java "applet" development tools. (Applets are small, platform-independent, object-oriented Internet applications created with Sun's Java programming language. Because of the ease with which they can be "dropped into" Web pages, they're expected to significantly enhance the Internet browsing experience.)

This document also features brief essays on the Mac OS platform's strengths in different development situations. The section on why the Macintosh is great in client/server environments, for example, could provide useful proposal material to systems integrators or VARs (value-added resellers) pitching Macintosh-based solutions.

Whether you're an application developer, solution integrator, client/server developer, or multimedia author, this guide provides you with information to help you make informed decisions on tool purchases—decisions that will ultimately result in faster development and better software.

The *Guide to Macintosh Development Tools* will be included on the next Developer CD Series release, and it is also posted at one of Apple's Web sites (see the box on page 14 for the site addresses).

## Apple Enhances E.T.O. and MPW Pro

Apple Computer, Inc., recently announced the availability of release 19 of its E.T.O.: Essentials • Tools • Objects and MPW Pro development products. New software components included in these products makes it easier for developers to adopt key Mac OS technologies and to develop higher-performance applications while shortening time to market.

### New Features

Release 19 of E.T.O. and MPW Pro include the following new features:

- MrC/MrC++ 1.0, highly optimizing C and C++ compilers that developers can use to create faster "native" Power Macintosh applications. Performance improvements vary with the application, but some test cases have shown that programs built using these new compilers run up to 25 percent faster than the same programs built using other popular compilers.

- MacApp 3.3, a significant upgrade to Apple's popular object-oriented application development framework. It features many performance enhancements and includes integrated support for system technologies such as Open Scripting Architecture (the foundation on which AppleScript is built), Macintosh Drag and Drop, and PowerTalk mailers.

- Power Macintosh Debugger 2.0, an improved version of Apple's standard one- and two-system debugger.

- The first release of Code Fragment Manager for 680x0 Macintosh systems (CFM-68K), run-time software that enables developers to create shared

libraries and drop-in code modules for 680x0-based applications in much the same way they do today for native Power Macintosh applications.

- The first release of SOM-objects for Mac OS, an object-oriented programming technology for building, packaging, and manipulating binary class libraries. SOM adds value to CFM-68K because SOM can be used to develop an object-oriented programming interface to a shared library. SOM is also important because it must be present for OpenDoc to work on both 680x0- and PowerPC processor-based Mac OS computers.

- SC/SCpp, new C and C++ compilers that generate 680x0 code and support the new CFM-68K run-time model.

- An early development release of MrPlus, a new performance tuning environment that can help accelerate native Power Macintosh applications. MrPlus can monitor run-time execution of an application, then rearrange code to gain optimal usage of the PowerPC instruction cache and to reduce the number of virtual-memory page faults. MrPlus can also act as a profiling and code coverage tool.

### Lower Prices

Apple has lowered the price of both E.T.O. and MPW Pro to make the tools more accessible for all developers interested in writing applications for 680x0- and PowerPC processor-based Mac OS computers. The price for the E.T.O. Complete New Subscriber Package has been lowered to \$595 (from \$795).

This package includes printed manuals for major components, three CDs containing the software (*MPW Pro*, *Symantec C++ for Power Macintosh*, and *Inside Macintosh*), and automatic delivery of the next two E.T.O. releases for free. Releases occur

in April, August, and December each year. The price of a one-year E.T.O. subscription renewal has been lowered to \$195 (from \$295).

The price for MPW Pro has been lowered to \$195 (from \$295). This includes the software on CD and two printed introductory manuals. Other manuals are included in electronic form on the CD; you can also order printed manuals for a separate charge. Current MPW Pro customers can upgrade to MPW Pro release 19 for \$75.

All prices listed here are for the United States market. International customers should contact their local APDA offices.

To order either product, customers can call one of the following numbers: 800-282-2732 (United States), 800-637-0029 (Canada), or 716-871-6555 (international).

## Symantec, Microsoft Release New Development Tools

Symantec Corporation recently shipped the Symantec C++ Subscription Version 8.0 Release 4. This is an improved version of Symantec C++ for Power Macintosh version 8.0. It includes Apple Computer's new MrC and MrC++ compilers, which compile Power Macintosh applications that run up to 25 percent faster than other leading compilers. (See the previous story, "Apple Enhances E.T.O. and MPW Pro.")

Symantec C++ Subscription Version 8.0 Release 4 also entitles the purchaser to receive the next two updates of the product free of charge. It's available for an estimated retail price of \$399. For

more details, you can get the Symantec press release from the site listed in the "Internet Resources" box on page 14.

Microsoft Corporation recently released the Microsoft Visual C++ Version 4.0 Cross-Development Edition for Macintosh. This product enables developers to recompile Windows-based applications to run on either 680x0 or PowerPC processor-based Mac OS computers. The resulting Mac OS programs have a "native Macintosh look" and run on any Mac OS computer that uses System 7 or later. This cross-development software executes on an IBM-compatible computer running the Windows NT operating system.

The Microsoft Visual C++ Version 4.0 Cross-Development Edition for Macintosh is priced at "approximately \$1,999," according to a Microsoft press release. For more details, you can view the press release on the Web. (See "Internet Resources" on page 14 for the site address.)

---

## Apple Ships

### QuickTime Conferencing Kit

Apple's QuickTime Conferencing Kit is now available worldwide through Apple authorized resellers. The kit allows you to participate in videoconferencing sessions with AV Macintosh and AV Power Macintosh computers from anywhere in the world. You can also use it to take PICT "snapshots" during conferencing sessions, record video sessions as QuickTime movies, and share and mark up images, sounds, and QuickTime movies through a shared window and a technology called "whiteboard collaboration."

The QuickTime Conferencing Kit contains the Apple Media Conference software application, the QuickTime Conferencing System Extension, the QuickTime Conferencing 100 Camera, and all necessary cables and documentation. The kit costs U.S. \$289, and requires an AV Macintosh or AV Power Macintosh computer, System 7.5 or later, 16 MB of RAM, a PlainTalk Microphone (which ships with most AV Macintosh and AV Power Macintosh systems), and a connection to an Ethernet network and/or the Internet through a TCP/IP connection.

The kit can also be upgraded to use high-speed ISDN connections with the QuickTime Conferencing ISDN Kit; with the ISDN upgrade, customers can participate in long-distance and cross-platform videoconferences with any other H.320-compliant desktop videoconferencing system. This allows for a larger video image size, higher video frame rate, and the ability to conduct video conferences outside the local area network with one or more participants, whether they're using a Mac OS or Windows system.

The QuickTime Conferencing ISDN Kit is fully compliant with the ITU standard H.320, the international standard for video compression and decompression. Apple is also continuing to work toward incorporating the new ITU T.120 standard into the QuickTime Conferencing product line, which will allow true application sharing and cross-platform collaboration.

The QuickTime Conferencing ISDN Kit includes a NuBus expansion card, which provides the ISDN connection as well as an H.320-compliant hardware codec (for video compression and decompression); QuickTime Conferencing H.320 software; a telephone handset; and all necessary cables and documentation. The

NuBus ISDN Kit was scheduled to be available starting in January for approximately U.S. \$1750. A PCI version of the kit is expected to be available by spring 1996.

You can license the QuickTime Conferencing core technology and APIs (application programming interfaces) to develop your own videoconferencing applications. For information about the technology and licensing, see the Mac OS Software Developers' Kit #4, or contact Apple Developer Support by sending an AppleLink message to DEVSUPPORT

---

## Apple Products

### Win Recent

### Industry Awards

During December 1996, QuickDraw 3D, the Macintosh Performa 5215CD computer, the Color LaserWriter printer, and ColorSync 2.0 all received awards from computer industry publications.

As part of its annual Eddy Awards for computer products, *MacUser* magazine named QuickDraw 3D its Breakthrough Technology of the Year for 1995 and gave the Macintosh Performa 5215CD computer the Mac System of the Year Award. QuickDraw 3D is the recently released Mac OS technology for rendering three-dimensional images and incorporating them into applications, enabling Power Macintosh users to work with 3D objects just as they would text, sound, PICT files, or other standard Macintosh data types. *MacUser* said about QuickDraw 3D that it is a "... perfect example of what Apple does better than anyone: create plug-and-play software that enhances every user's experience."

*Publish* magazine awarded 1995 IMPACT awards to the Color LaserWriter 12/600 PS and ColorSync 2.0. *Publish* magazine bestows IMPACT awards on those products and technologies that significantly advance the future of desktop publishing. The Apple Color LaserWriter 12/600 PS printer, which had its debut in August, delivers near photographic-quality color printing coupled with the highest quality black text at extraordinarily low cost, thanks to new technology that resulted in Apple's filing for 42 separate new patents. The printer includes ColorSync 2.0 color-matching technology, which matches color between imaging devices, mapping from one device's profile to another to create accurate color output. It's designed to work smoothly with all Mac OS, UNIX, and Windows-based computers.

---

## Apple Announces

### Support for

### JavaScript

Apple has joined over 25 other companies in supporting JavaScript, the open, cross-platform object-scripting language recently announced by Netscape Communications and Sun Microsystems. JavaScript enables the creation and customization of applications on enterprise networks and the Internet. Ike Nassi, Apple's vice president of system software development said that "JavaScript will be a key component in building interactive applications for the Internet, and we are looking forward to working with Netscape to encourage adoption of JavaScript on Macintosh platforms."

JavaScript complements Sun's Java programming language, which is used for adding multimedia and

interactive objects to Web sites. JavaScript is an object scripting language designed for creating live online applications that link together objects and resources on both clients and servers. While Java is used by programmers to create new objects and “applets,” JavaScript is designed for use by Web page authors and enterprise application developers to modify the properties and behavior of Java objects running on either the client or the server.

For example, a multimedia weather forecast applet written in Java can be scripted by JavaScript to display appropriate images and sounds based on the current weather readings in a region. Or, a server-side JavaScript script might pull data out of a relational database and format it in HTML (that is, hypertext mark-up language) “on the fly.”

Netscape and Sun plan to propose JavaScript to the W3 Consortium (W3C) and the Internet Engineering Task Force (IETF) as an open Internet scripting language standard. To encourage its adoption, the companies intend to make a source-code reference implementation of JavaScript available for royalty-free licensing.

Java is available to developers

free of charge. The Java Compiler, Java Developer’s Kit, and related documentation are available from Sun’s Web site (see the “Internet Resources” box on this page for the location). Java source code can be licensed for a fee; licensing details are available at the Sun Web site.

## Pippin Coming

### Soon to a TV

### Screen Near You

During December’s PC Outlook Conference, Apple CEO Michael Spindler used a prerelease Pippin device to access the Internet using Netscape Navigator as well as to play a variety of Mac OS CD-based software titles.

Pippin—expected to be released this spring in Japan and in the second half of the year in the United States by Apple licensee Bandai Corporation—is the inexpensive multimedia player and Internet access device based on the Mac OS and QuickTime multimedia technologies. Its release is expected to expand the

market for your Mac OS multimedia titles among customers who aren’t yet ready to buy a full-blown personal computer but who want access to powerful multimedia and networking technologies.

Used in conjunction with a television screen, Pippin uses the RISC-based PowerPC 603 microprocessor. Its features include high-resolution graphics and text optimized for TV video output; a quadruple-speed CD-ROM drive; full stereo sound output; serial ports and PCI peripherals; and a full range of networking capabilities, enabling Internet browsing. Pippin plays multimedia CD-ROM

titles as well as audio CD and Photo CD titles; also, all Pippin CD-ROM titles can be played on suitably configured Power Macintosh computers. ♣

## Internet Resources

- The QuickTime VR example from Time Online is located at <http://pathfinder.com/time/special/baseball>.
- For Apple’s *Guide to Macintosh Development Tools*, go to Web site <http://www.devtools.apple.com/general/devtoolsguide2.html>
- The QuickTime VR Web site is located at <http://qtvr.quicktime.apple.com>.
- For the Symantec press release, go to the site at <http://sunweb.symantec.com:80/compinfo/news/products/mcpp84pr.html>.
- For the Microsoft press release, go to the site at <http://www.microsoft.com/VCPINFO.HTM>.
- The Sun Web site (for more information about Java) is located at <http://java.sun.com>.

## Editor’s Note

*continued from page 2*

ket share data released by International Data Corporation and Dataquest (among others) and Apple itself over the past several months. I won’t repeat it all here, since we’ve given it prominent play in the last several issues, but suffice it to say that Apple picked up 20 to 25 percent of its 20 million-plus Macintosh customer base in 1995 alone, enjoying record, or near-record, quarters throughout the fiscal year. So, when the *Computerworld* writer says “the Macintosh market is thriving,” she knows what she’s talking about.

Also, SPA perennially wishes that it could gather more complete data about the software industry. SPA researchers are somewhat at the mercy of the software companies who are willing to volunteer sales information for SPA’s quarterly reports. Because many companies aren’t even in touch with SPA, and others aren’t willing to share their data, SPA’s reports can’t be said to encompass the entire indus-

try. (By the way, Apple has actively supported SPA for many years, and it’s in all our interests for you to do the same; I’d suggest you contact SPA and find out how your company can help ensure that SPA’s research is as complete as possible. You can call them in Washington, D.C., at 202-452-1600, or visit the SPA Web site at <http://www.spa.org> for more information.)

None of this gets around the fact that SPA’s top-level conclusion (and easiest-to-remember sound byte) is that overall U.S. Macintosh software sales were down in Q3 ‘95. However, as with any market data, it’s important to look behind the most-visible and most-obvious points to see what’s really going on.

*Paul Dreyfus*  
Editor

# Technology

## CD Highlights

## Tool Chest Edition, February 1996

Thanks to being squeezed between the European Developer Forum on one end and the holidays on the other, this month's disc is a bit smaller than usual. The Japanese and Chinese Language Kits make their reappearance; the next set of kits were not quite ready to ship at press time, but will appear soon. The New System Software Extensions folder is history, its contents provided (with voluminous additions and updates) on last month's Mac OS SDK Editions. In its place, I've piled up 230 MB of the latest printer drivers. And now, this month's featured packages. . . .

### AppleGlott 2.4b2

AppleGlott 2.4b2 is a text translation tool used by Apple localizers to localize Apple software. AppleGlott extracts the text from a resource file and dumps it into a text file for translation. Once the text is translated, AppleGlott replaces the original text in the resource file with its translation. In addition to the regular translation, AppleGlott handles updates from one version to another, and copies over resizable items. AppleGlott 2.4b2 also

- supports 'RMAP' resources for remapping to a different template
- can treat the data fork as a resource
- works with an environment-relative batch file
  - uses P0nn retry logic
  - features file-specific templates in a `_FileInfo` folder
  - provides auto-association by name
  - features bug fixes for size fields and speed zone (see the What's New folder)

### ColorSync 2.0.1 GM

In response to bug reports from several internal and external project groups, ColorSync 2.0.1 has been released to fix several bugs. See the document ColorSync 2.0.1 Release Note for details.

### Gestalt Selectors List 3.2

This document lists all selectors known to the creator of the Gestalt Selectors List for use with the Gestalt Manager. They can



*Tool Chest Edition, February 1996*

include selector codes installed by Apple system software or by your software. The information in this list is useful for programmers who use the Gestalt Manager with their software (even using externals, as with HyperCard, 4th Dimension, and so on).

Please note that this is not an Apple product. It is provided on an "as is" basis. Apple Computer, Inc., is not responsible for any problems you may encounter in its use.

## Inside This Section

**Human Interface: If You Build It, Will They Come?**

16

### GXOverrideTool

GXOverrideTool is a message override tool source file for QuickDraw GX. It creates ".r" files (for override resource) and ".a" files (for jump table) for developers who program drivers and extensions. Note that the tool does little sanity checking, so beware. Examples of the dash options and command lines understood by the tool when parsing the source file are available in the DocumentationAndSample file.

### SetDefaultDTP

This application demonstrates how to send the Finder an Apple event that causes a new default desktop printer to be selected under QuickDraw GX. The user interface consists of a dialog box that collects a string that is the name of a desktop printer. When the Default button is pressed, the application creates an Apple event and sends it off to the Finder. If the Printing Finder Extension (part of QuickDraw GX) is installed and a valid desktop printer was specified, that printer becomes the default printer.

### Speech Recognition 1.5a1

This package contains version 1.5a1 of Apple's new Speech Recognition Manager. This is alpha software and documentation—not yet ready for wide distribution or release in products. When finished, version 1.5 of the Speech Recognition extension will embody the first officially supported developer release of the Speech Recognition Manager.

The draft documentation included with this note, in the file Speech Recog Draft

*please turn to page 18*

## Human Interface

# If You Build It, Will They Come?

By Peter Bickford

There's a reference guide for psychologists called *DSM-IV*. This is basically a huge tome that lists all the currently known psychological ailments and their trademark symptoms. Periodically the guide is revised (thus the "IV" in the current edition), adding new ailments as they are classified, and getting rid of others as general opinion decides that a particular behavior may make you a drag at parties, but is not technically a sickness.

Since the guide is overdue for an update, I thought the editors might want to consider a new ailment for inclusion: "Internet Psychosis." Similar to mass hysteria, it's the irrational belief that absolutely everyone from school children to forest rangers needs to get onto the Internet if they're to have a hope of succeeding in the future. The chronic condition is also characterized by the perceived need for people to have their own Web page, lest the world somehow forget they exist.

The Simpson trial, Windows 95, and New Coke had nothing on the Internet when it came to the amount of sheer hype involved. Business people in particular are tantalized by the prospect of reaching millions of well-heeled potential customers for almost nothing. But before you crack open that HTML primer and fire up your Apple Internet Server Solution for the World Wide Web (a lengthily named, but very fine product from your friends at Apple) you need to do some planning to make sure your Web site lives up to its well-stated potential.

What's a "Web site," anyway? Technically it may be "an informational presence on the World Wide Web." But if you want anyone to visit your Web site, you'd better make sure it's more than that. Today, the Web is a Turkish bazaar of information filled with a million people hawking random trinkets of data and vying for users' attention. Your ultimate goal should be to have users say, "Well that's all very interesting, but whenever I want X, I go to your site." The keys to achieving this are to give your Web site a distinct identity, to make it easy for people using your site to find what they're looking for, and to ensure that the whole experience is enjoyable enough that they come back.

## Find a Metaphor

Right now, the Web is dominated by people who like technology for its own sake. These are the folks who are capable of saying to themselves, "Oh yeah! The Web is just the Internet with richer message formats and hypertext." Technology is seen in terms of other technologies, and there's the general assumption that the unexplored may hold great rewards. That's why technophiles are so jazzed about the Internet, envisioning all the wonders that a new world of connected computers might bring. They'll get hooked up just so they can be there when the future reveals itself.

But the Internet's ultimate success is going to rely on attracting The Rest of Us. To win these folks over, you need to be able to

explain what the technology is and what it will do for them, using examples from real life. Instead of trying to sell The Rest of Us a PDA, start by touting it as a really great personal organizer. Similarly, don't portray your Web site as an ambiguous "informational presence"—model it as an online magazine, library, or storefront. This metaphor forms the core of your site's identity. It lets potential users know what to expect from your site, as well as what benefits they'll be getting out of using it. It also puts you way ahead of the sites whose basic metaphor is "a bunch of stuff I thought I'd post because it seemed interesting."

Designing your site with an appropriate metaphor also gives you a way to know what content you should be focusing on and how often it should be updated. For instance, if you're building a storefront-style site for Wombats-R-Us, your users will expect to find product data, pricing, availability, and other information that they could get from a store visit. It's great to throw in bonuses, such as editorials on the state of the Wombat industry, but the major success of your site will rely on living up to the basic metaphor—in this case, delivering a quality Wombat shopping experience. Similarly, magazine-style sites will be judged on the timeliness and quality of the various editions, library-style sites on the breadth and depth of their collection, and so on.

## It's in Here Somewhere . . .

There's a regular scavenger hunt held on the Internet to see who can find various bizarre pieces of information: How many people sailed on Columbus' second voyage? Who was the department head of Psychology at Stanford in 1978? And so on. Contestants are only allowed to use information found on the Internet, with the point being both to have fun and to prove that darn near anything can be found *someplace* on the 'net.

It's all very impressive stuff, but it reminds me of picking through the overstock bins at a record store. Somewhere amidst the thousands of random discs may be that elusive single of Camouflage's *This Smiling Face* that I've been wanting, but I'll probably never know—it's simply too much work to search for it. Meanwhile, the store owners are desperately trying to get rid of their overstock, marking it down time and again in an effort to convince customers to wade through and buy something. One Camelot Records store owner took another approach: He added alphabetical dividers to his overstock section, drastically cutting down on the time required for customers to find a given disc. His store carried the same overstock as the others in his chain, at the same prices, and in truth, he didn't have the greatest store location. Still, this owner managed to sell over four times more overstock than the chain's average, simply because he made it easier for customers to find a given disc.

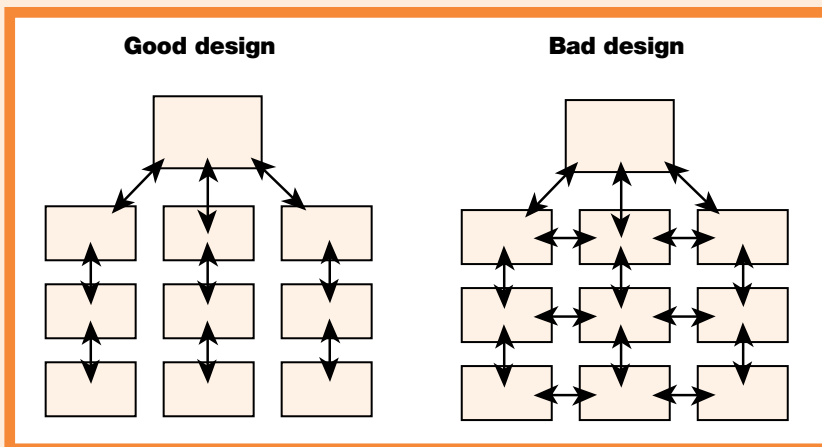
Once you've attracted users to your Web site, make sure they don't leave in frustration because they get lost or can't find what they're looking for. Ideally, you should provide two major sorts of navigation: a high-level directory so users can browse for items in a



given category, plus a detailed search for specific, known items. The high-level directory can be based on your site's basic metaphor and can take the form of a magazine table of contents, store directory, or list of services. It should be one of the first things users see when they enter your site, and should be available at all times. Because this directory will be your user's primary means of navigating the site, make sure that it can be displayed quickly, without downloading 100K of graphics first. And if you do use graphical maps, make sure that the various sections are outlined in black or otherwise set off so that they will be recognized as clickable areas.

To help your users keep their bearings, display the current area's name or logo prominently on your site's various pages so that it becomes something of a signpost. You should also make the various pages of an area adopt similar graphical looks to help unify them.

When laying out the paths for users to navigate your system, try to keep things as simple as possible. There are many possible navigational schemes for a site, including the linear path, network model, and the ever-popular "just click on the blue words and maybe you'll go someplace interesting" scheme that is in use on many Web sites today. A more reasonable alternative for most sites is the straight "tree" structure where users can choose a topic they are interested in, then work their way down to more specific subtopics. If the users feel they're on the wrong path, they simply go back up the tree and



choose a different main topic. A diagram of this scheme is shown on the left side of the figure. Try to avoid the situation shown on the right, in which users can proceed directly from a subtopic of one main topic to the subtopic of an altogether different main topic. This tends to be horribly confusing since users can find themselves several layers deep inside a Web site without having gone in the "front entrance" of that subtopic. It's a bit like walking out of your house's bathroom and finding yourself in the bedroom of an entirely different house.

Once your navigation scheme is in place, test it with several users. You're on the right track if users can find their way around the site even when you've hidden the "go back" button on their Web browser. Your Web site's navigation scheme should be self-contained and shouldn't need to rely on a browser application's ability to "go back to the last place I knew what I was doing."

### Make the Experience Enjoyable

By now, you've gotten users to visit your site and find their way around. Now all you need to do is make sure they have a good time while they're there. This is the part that most Web design books cover in depth, so I'll stick with giving a few design hints:

- *Design for the future.* HTML (hypertext markup language, if you really need to know) is evolving quickly to provide richer control of styles, colors, backgrounds, and formatting. Given the low fidelity of HTML 2.0, it may be worth taking a chance and incorporating some of the proposed 3.0 standards, as well as extensions offered by Netscape and Java. (See IndustryWatch on page 3 for more about Java.) Whatever variants of HTML you use, test your design with any browser applications that you expect a sizable number of your visitors to be using. Ideally, you should design your pages so they're at least readable by users of older browsers, although they may not get the full graphical effect.

- *Watch the graphics.* Not everyone has 10-Mbit lines to the Internet, and the state of the analog modem art is likely to be pegged near 28.8 Kbits for some time. Although I don't suggest you stick with text-only pages, make sure that you aren't loading them down with so many graphics that the drawing times become intolerable. Small, effective graphics are ultimately more pleasing to users than 24-bit extravaganzas that require 30 minutes to download.

- *Background tricks.* Many of the current Web browsers let you use special tricks to get great-looking backgrounds with little overhead. Netscape Navigator, for instance, lets you tile a small graphic to give a fully rendered background. You can also change the background color directly using a text command in your HTML page. This looks especially striking when you use several background change commands in a row to give a "fade-up" or "fade-out" effect upon entering a page.

- *Use moderation.* The tendency of many first-time Web page authors is to overdo things. Don't flash headlines, use shocking-pink text on a blue background, or change font sizes in every other line of text. Let the user concentrate on what you

have to say, not how loudly you're saying it.

- *It's worth hiring a graphic artist.* Back at school, while we were learning to use programming languages to communicate with computers, our graphic artist colleagues were learning to use images to communicate ideas to people. Building a good Web page requires both skills, and not many of us are as good at one as we are at the other.

*Till next time,*

*Doc*

*AppleLink: THE.DOKTOR*

*Got an interface gripe? Send it to Doc at AppleLink address THE.DOKTOR or Internet address [the.doktor@applelink.apple.com](mailto:the.doktor@applelink.apple.com)*

## CD Highlights

*continued from page 15*

11/21/95.pdf (in the SR 1.5a1 Documentation folder), is an Adobe Acrobat file. The documentation describes version 1.5 of the Speech Recognition extension and the API and Toolbox it embodies. (Version 1.5 contains many features that were not available in version 1.4.1 or earlier versions of the extension.)

Input from early adopters of Apple's speech recognition technology has greatly improved the Speech Recognition Manager API. If you find bugs in the extension or documentation, please let us know at AppleLink address PLAINTALK or Internet address [plaintalk@applelink.apple.com](mailto:plaintalk@applelink.apple.com).

See the document READ ME FIRST! for more details.

### **WorldScript Updater 7.5.1**

This folder contains the 7.5.1 WorldScript Installer. This installer will install files needed by WorldScript technology for users running Mac OS version 7.5 and later. Users of

the Japanese and Chinese Language Kits or applications that depend on WorldScript technology should install these files, since they improve the stability and performance of your system. QuickDraw GX and Power Macintosh users should upgrade to these files, because they provide important improvements over previous versions. See the WorldScript Installer Read Me document for details.

### **Coming Next Month**

Possibilities for next month include the QuickTime Conferencing SDK, a new System Update, and the usual vast expanse of technical documentation.

*Alex Dosber*  
*Developer CD Leader*

## APPLE DIRECTIONS EXPRESS

*At-a-glance, up-to-the-moment business news from Apple  
Apple Directions Express—the Apple Online Business Bulletin  
Sent to you instantaneously—and free—over the 'net*

Apple Directions Express is the official online business bulletin from Apple Computer, Inc., and the editors of *Apple Directions*. At least every other week, we send you a single, concise e-mail providing you with important, up-to-the-moment news and competitive analysis from inside and outside Apple.

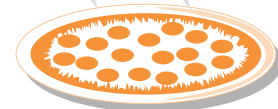
Tired of searching the 'net for Apple-related material? Apple Directions Express not only gives you the news, but points you to Internet locations for complete information about new Apple products and technologies, strategic initiatives, market opportunities, and other subjects that are important to the business of Macintosh development.

**SUBSCRIBE now to receive this valuable service. THERE'S NO CHARGE!**

Send e-mail to [adirections@thing1.info.apple.com](mailto:adirections@thing1.info.apple.com).

In the subject field of your message, type the string "subscribe <your real name>".

We'll send you a free pizza over the Internet  
if you act NOW!  
<well, we would if we could>



# Business

## Developer Outlook

# The Business Case for Component Software

By Robert Beech, President, Pharos Technologies

I'll say it straight out. I'm one of those who believe software components will forever change the way software is constructed. Software development is headed toward a "Detroit" model of mass customization, where many different solutions are rapidly assembled using prefabricated parts, multipurpose assembly environments, and a limited amount of customization.

I should quickly acknowledge that I'm not a trade columnist who can make predictions like this and never lose a night's sleep. To the contrary, I'm betting the future of Pharos Technologies, the successful, multimillion dollar company I founded, on the development and licensing of

software components for specific vertical markets. If I'm right, you can read about our hot stock offering in a few years. If I'm wrong, well . . . everything in my past experience leads me to believe I'm right!

First, let me share some industry observations that I've made during my ten years in the software development business. Then I'll explain how these experiences support my strongly held belief that software companies need to prepare for the component revolution.

### The Holy Grail: Custom Software With Less Risk

Since 1986, Pharos Technologies has been in the business of providing customer-specific software to enterprise clients. I've often referred to Pharos as a software

expeditionary force, hired by our clients to take them across rugged software terrain to a specific destination. Pharos has built a reputation for handling difficult "software expeditions" using the latest technologies and innovative strategies. Over the years, we've built most of our software using third-generation language (3GL) tools, a necessity given our clients' proprietary requirements and uncharted territories.

Our clients appreciate the software we deliver, but they don't like the higher costs and longer delivery times associated with fully customized solutions. We're often asked, "Can't you give me what I want without having to resort to high-risk software development?" I've spent the last few years searching for the right answer to that question.

## Inside This Section

**Producing "Bug-Less" Software—Part 2: Bug-Chasing Tactics** 23

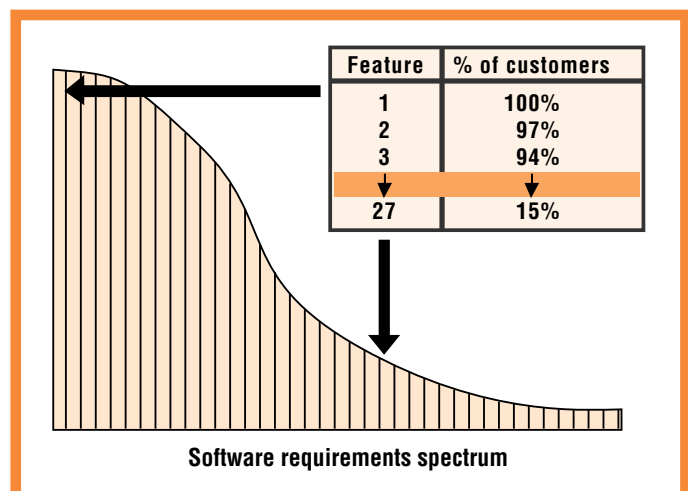
important difference is that software components can be more easily modified and customized.

Components represent an opportunity to reuse finished software parts in different combinations, satisfying both shared and unique requirements of specific markets. With the advent of CI Labs' OpenDoc component software standard, which Apple Computer, Inc., is co-developing, and Microsoft's OLE standard, you will soon be able to combine and reuse components that have been fabricated by different development sources.

Component software has inherent appeal. In an age of mass customization, who can argue against the value of prefabricated software parts? Yet, the compelling business case for components—the one that weaves together the highest value proposition for component developers and their customers—has not been clearly articulated. The case exists, but understanding it requires appreciation of the "feature requirement spectrum" that defines any software market.

### One Size Fits One—The Business Case for Component Software

Components can best be described as the software equivalent of prefabricated building blocks. Compared with software objects, components are larger modules, built from a number of tightly integrated objects and delivering a much higher level of functionality. Properly designed components can be readily assembled to provide a partial or complete software solution. In many ways, software components are analogous to integrated circuit microprocessors on the hardware side of the business. An



Curve of feature commonality. Each vertical bar represents the percentage of customers who share the need for that specific feature.

### Curve of Feature Commonality

If all of the various customer requirements could be identified

for a given software market, a graph could be constructed whose curve would show, in order of highest to lowest, the percentage of customers who share each of the identified requirements. The resulting requirement spectrum and its “curve of feature commonality” (shown on page 19) set the stage for analysis of the value proposition for components.

On the left side of the curve are the software requirements that are shared by most customers in a given market. For example, in the markets needing sales force automation systems, it is highly likely that 100 percent of all customers would require a contact management feature.

On the right side of the curve are the requirements shared by the fewest customers. At the far right, the curve identifies requirements that are unique to individual customers. In the sales force automation example, a single customer might desire a specialized quotation generator that produces complex quotes based on a proprietary set of expert rules.

A simplistic way of describing the spectrum would be to characterize the requirements on the left as “shared” and the ones on the right as “unique” (shown in the “Differentiation value gradient” figure on this page). The shared requirements on the left represent the lowest differentiation value to a customer. These are mainstream requirements that are shared with other customers in that market space.

The highest differentiation value goes to the requirements that are shared by the fewest customers—the ones on the far right that uniquely position a company within an industry and give it a major advantage in the marketplace.

### Historical Software Strategies

Historically, software strategies have been based on positioning

within this differentiation value gradient (shown in the figure on page 21). The suppliers of off-the-shelf, one-size-fits-all software try to meet the combination of shared requirements necessary to hit the sweet spot for price and volume. Any customers who require features that are well outside the mainstream won't be satisfied. Off-the-shelf software, by nature, rarely progresses into the area of high differentiation value.

Fourth-generation language (4GL) tools provide customers with the ability to move closer to a complete solution. However, the primary attraction of 4GL tools is ease of custom application building, which often comes at the expense of performance and lower-level control over the technology. Eventually many customers come up against these limitations as they attempt to push the performance and functionality of 4GL applications to accommodate all of their requirements. In the end, they must often settle for less than what they really want.

It has been the domain of third-generation languages (3GLs), such as C and C++, to deliver the complete spectrum of customer requirements. Unfortunately, programming in C and C++ is a complex undertaking that carries significant risks and disadvantages. The most frequently heard complaints regarding 3GL efforts are high cost and slow time to delivery. Of the two, the latter is often viewed as the highest risk, since business conditions change so fast that many 3GL efforts become bogged down in the quicksand of change orders and are eventually abandoned. Customers who choose to fund a 3GL effort usually have very compelling, mission-critical requirements whose high business value justifies the risks involved.

The challenge to software developers is this: How do you

give customers everything they want without subjecting them to the significant risks of a fully customized, 3GL software development effort? It is precisely this challenge that component software is prepared to meet.

### The Case for Components

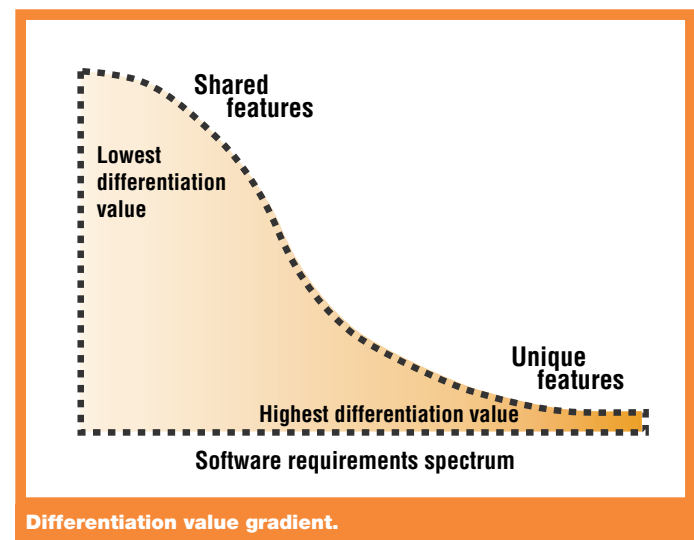
A look at the chart on page 22 shows the case for a components strategy. This chart contrasts the major characteristics of the component and customized portions of the curve. The shared requirements on the left will support the R&D investment in prefabricated software components that can be reused within customer-specific projects. Component sets can be readily assembled on a customer-by-customer basis using the appropriate combination of universal, market, industry, and niche-specific components, plus a multipurpose assembly environment. With proper market targeting, these component sets, or frameworks, will encompass up to 70 percent or more of the requirements for any specific customer project. The remaining percentage will require custom code to meet a customer's unique and proprietary requirements, providing the business incentive for justifying a custom coding effort.

Together, the two portions form a new value proposition that

is compelling for both the developer and customer of component-based software.

From a developer's standpoint, the opportunity to rapidly deliver high-value, one-size-fits-one solutions—while also earning license revenues and higher net margins through component reuse—is a highly attractive business proposition. Over time, a developer's in-house component library, market targeting, customer-specific knowledge, and exclusive partnering relationships will help form significant barriers to competition. Enlightened developers will use this attractive opportunity to invest in component factories and license the component output through several OEM (“other equipment manufacturer”) channels—including, as appropriate, their own integration and customization services. (See the “Channels to end-customers” figure on page 22.)

Customers win big as well. Of highest value to customers is the opportunity to dramatically decrease time to delivery while improving the flexibility and reliability of tailored software that will meet all requirements. This includes the critical ability to keep up with rapidly changing business conditions. Lower overall costs, in the form of component license



fees and reduced maintenance expenses, provide the icing on the cake.

### Ideal Markets

It is important to note that not all markets are ideal candidates for component-based software. In some markets there is little need for differentiation between customers, as almost all requirements are shared. This type of requirements profile does not drive the need for various component combinations and is best suited for one-size-fits-all solutions that move sequentially through traditional upgrade versions.

The ideal markets for component-based software are ones in which there are a large number of shared and unique requirements. Just a few examples of such segments are health care administration, sales force automation, and electronic retailing. The shared requirements in these types of segments promote volume reuse of components. The less common requirements—the ones with the greatest differentiation value—promote various component combinations plus limited customization.

The answer to the question I presented in the beginning of this

article—How do you get custom solutions with far less risk?—is through component-based software. Ultimately, demand for component-based software will

grow rapidly as more customers realize its advantages and move their software strategies toward the high-value portion of the requirement spectrum (shown in the chart on page 23).

car brands. In today's competitive environment, it is essential for car companies to reuse parts as much as possible to improve quality and drive down costs.

Solution providers, such as systems integrators, will realize similar benefits from the reuse of software components. Today, component factories such as Pharos are busy specifying, fabricating, testing, and bundling software parts for specific markets. These high-quality, high-performance components will be licensed to solution providers as comprehensive packages coupled with multipurpose assembly environments. Solution providers will rapidly assemble the bulk of each customer solution from finished software parts, and then, if neces-

**Component-based software will help enable the entire spectrum of customer requirements, while dramatically reducing the time, cost, and development risks normally associated with custom software.**

## Component Software Resources

If you're interested in learning more about software components and related technologies, you can find numerous publications and Internet resources. Here are a few of our favorites.

### Publications

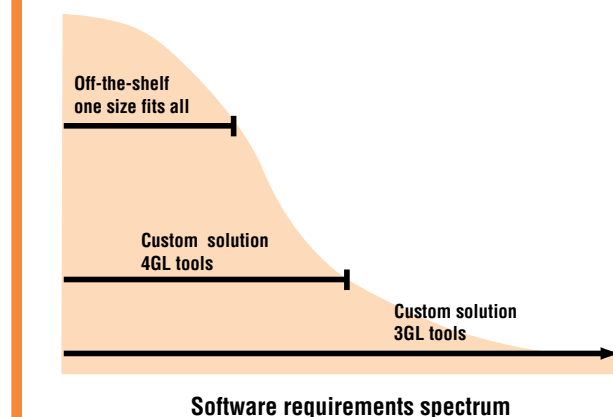
- *The Essential Distributed Object Survival Guide* by Robert Orfali, Dan Harkey, and Jeri Edwards (ISBN 0-471-12993).
- *Object-Oriented Programming Using SOM and DSOM* by Christina Lau (ISBN 0-442-01948-3).
- *The Essential CORBA: System Integration Using Distributed Objects* by Thomas J. Mowbray and Ron Zahavi (ISBN 0-471-10611-9).
- *Inside OLE* by Kraig Brockschmidt (ISBN 1-55615-843-2).

### Internet Resources

- Apple's OpenDoc Web site (<http://coretools.apple.com/opendoc/>) is where you can download the latest version of OpenDoc, as well as everything you need to begin creating OpenDoc components, containers, and extensions.
- The CI Labs Web site (<http://www.cilabs.org>) is run by Component Integration Laboratories, Inc. (CI Labs), the nonprofit organization administering the technological specifications and foundation technology for using the OpenDoc component software architecture.
- The Object Management Group (OMG) Web site (<http://www.omg.org>) is run by a nonprofit consortium dedicated to promoting the theory and practice of object technology.
- IBM's Object Technology Web site (<http://www.torolab.ibm.com/objects/>) is a place to find out more about IBM's OpenDoc-related products and programs.
- News:news:comp.soft-sys.middleware.opendoc is an Internet newsgroup that can keep you informed of the latest component software news.

### Parts Assembly vs. Custom Coding

Today, individual cars are assembled using finished parts that have been tested extensively and perform according to strict quality specifications. The cost of R&D and testing of these parts is more than covered in their volume reuse within and across multiple



**Historical software strategies. Historically, 3GL tools were necessary to satisfy all of the requirements for a specific customer.**

sary, tailor the solution using a limited amount of custom code.

This assembly approach contrasts sharply with that of traditional 4GL tools that rely upon a proprietary shell and scripting language to craft self-contained run-time solutions. Traditional 4GL solutions are often hard to extend beyond the functional boundaries of the proprietary shell. Component-based software, however, can be extended easily.

**An Interactive Retail Case Study**

Pharos is currently focused on the development of a comprehensive set of OpenDoc components for use in interactive retailing, within traditional and nontraditional retail locations. A late change to our feature requirements called for incorporation of voice recognition to accommodate interactive shoppers who, for reasons of convenience or disability, will not interact by means of a keyboard or mouse. In a traditional 4GL tool, such as Macromedia Director, incorporation of voice recognition would involve writing a 3GL extension, gluing it to the 4GL environment, then changing every location in the 4GL script that must call the extended functionality.

With our component approach, incorporation of voice recognition was straightforward.

In one morning a Pharos engineer was able to write an OpenDoc part that functions with Apple's PlainTalk voice-recognition technology. He then dropped the finished PlainTalk part onto an OpenDoc container that already held a combination of other parts from our interactive retailing parts package. That package includes a QuickTime VR part that allows a consumer to move around a 360-degree interior panorama of a virtual house or retail outlet, "pick up" products, zoom in on those products, and rotate them in 3D space for all-angle viewing. The QuickTime VR part was designed to receive messages, through a central dispatcher part, from several other parts that could affect the viewing location within the QuickTime VR image. In essence, the VR part understands how to handle dispatcher messages that correspond to moving around or jumping to rooms inside a house or retail outlet.

When the PlainTalk part was dropped into our OpenDoc container—without having to relaunch—we could immediately issue voice instructions that corresponded to the same internal messages that were already understood by the QuickTime VR part or any other part affected by QuickTime VR location changes. The PlainTalk part simply directed its messages at the

central dispatcher part, which then fired them off to any other parts that were "listening" for such messages. There was no need to glue in the extended functionality—it was fully integrated when it was dragged and dropped onto the OpenDoc container. There was also no need to open any of the previously existing parts and make changes to their code. The implications of extending functionality like this, without having to resort to messy code rework, are significant and underpin the whole notion of economical, mass customization of software solutions.

**New Opportunities in Component Software**

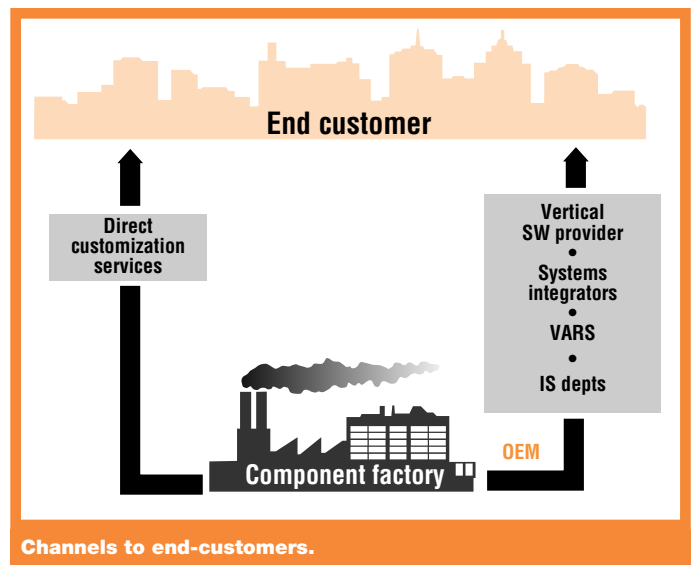
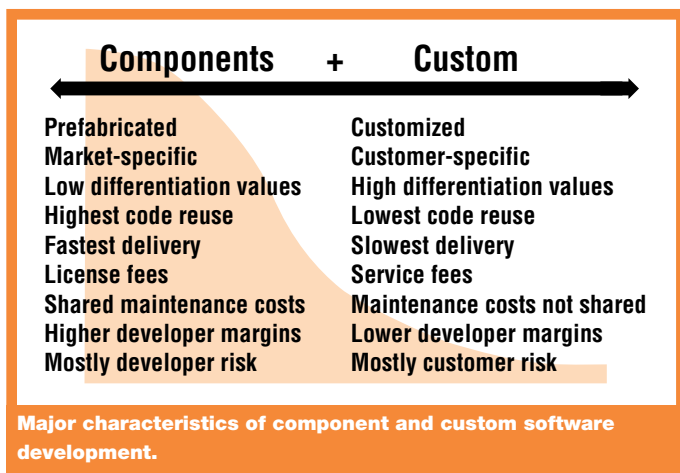
A number of new opportunities are emerging in the component software industry. Here are several functions that may fit your current or future capabilities and interests:

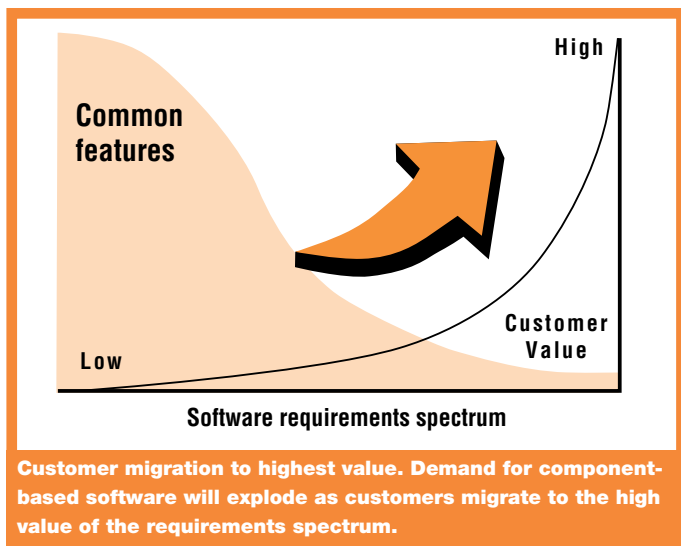
- doing mass customization of component-based solutions
- fabricating and licensing vertical components (Pharos' plan)
- fabricating and licensing horizontal components
- being a contract developer to vertical and horizontal component fabricators

- being a quality assurance lab for testing individual software components

Developers of horizontal applications should consider licensing their technology in the form of universal components that can be used in a wide variety of component-based solutions. Those with subject expertise in specific markets, industries, or niches are ideally suited to supply components to those categories. Beyond niche markets is the end-customer category, which will be the domain of solution providers who have reoriented their services toward mass customization of component-based solutions.

As an example, Pharos has chosen to fabricate components for vertical markets. One effort involves development of an extensive set of components that can be used to assemble interactive retailing solutions. Our marketing plan calls for us to license these components as a package, along with a multipurpose assembly environment, to solution providers who are focused on the interactive retailing marketplace. Using our components and assembly environment, solution providers will be able to rapidly deliver high-value, interactive retailing solutions, customized to the needs of individual customers.





### The Bottom Line

In any given industry, customers of software solutions have many shared requirements, plus a limited number of unique requirements—specific to individual customers—that are of high business value. Historically, these customers have been forced to choose a fully custom solution to satisfy all of their requirements. Component-based software will help enable the entire spectrum of customer requirements, while dramatically reducing the time, cost, and development risks

normally associated with custom software. As a result, customer demand for assembly and customization of components will generate enormous market opportunities for software developers and solution providers who are quick to embrace software component technologies. ♣

*Robert Beech (beech@pharos.com) is the president and founder of Pharos Technologies, Inc., a software component manufacturer located in Cincinnati, Ohio.*

### Business Feature

## Producing “Bug-Less” Software— Part 2: Bug-Chasing Tactics

By Victor J. Hnyp, President,  
Prosoft Labs

Great software testers almost always exhibit two personality traits—tenacity and an uncanny sixth sense for finding bugs. Though we can’t help you in the tenacity department, we can heighten your bug-finding intuition by sharing some bug-hunting techniques that we’ve developed over the last ten years.

My company, Prosoft Labs, identifies and eradicates bugs for a living. We’ve tested commercial software for publishers of Mac OS and Windows-based applications, utilities, and cross-platform multimedia products. We also write Macintosh software. Through these efforts, we’ve developed an effective methodology for producing software with the least possible number of bugs—in other words, “bug-less” software.

This is the second part of a two-part article that will hopefully help you create more reliable software. While Part 1 presented

the 10,000-foot view of the software testing process, Part 2 zeros in on some of our tried-and-true bug-chasing tactics. If you’re new to this process, this article will help you become a better tester of your own products. If you’re an expert software tester, you’ll benefit from our extensive list of bug-finding tests.

### A Case Study

To help explain some technical aspects of software testing, let’s assume you’re the producer of a fictional product, BuggyChecks 1.0. This simple checkbook management program was initially released full of bugs, and it’s your job to create the 1.0.1 bug-less version.

BuggyChecks includes all the requisites of a standard application: It has a graphical user interface, as well as processing, output, and storage-and-retrieval capabilities. It enables users to write checks, enter balance adjustments, void checks, and print checks—everything that can

be done with a bank checking account. Users of this product need a keyboard, mouse, video monitor, storage device, and printer. So what types of testing should you conduct on this program to make sure you don’t have to create a “1.0.2” version?

### Boundary Testing

Consider the following: John Doe has been using BuggyChecks 1.0 for several weeks, and everything has been working fine. One day, while entering checks, John answers the phone, continuing to enter checkbook figures into the program. Due to an inadvertent auto-repeat of a keystroke, he accidentally enters a figure that’s 12 digits long! After pressing the Return key, he’s presented with a dialog box stating that a system error has occurred, and the program shuts down. Alas, the checkbook file was not closed properly, and all the data has been corrupted. Your customer has lost several months of checks!

Obviously, BuggyChecks 1.0 was not “boundary tested.” Boundary testing entails entering legal and illegal data to see how the program reacts.

Legal data is expected data—valid numbers placed in numeric fields and alphabetic characters placed into alphabetic fields. (Keep in mind that—depending on the field—decimal points, spaces, backspaces, or other special characters might be considered valid characters, and should be tested as such.) Illegal data is unexpected data—for example, numbers higher or lower than the designated range, control characters, too few characters, too many characters, and so on.

In order to “fully test” BuggyChecks for the real world, you would have to check every possible combination of keys and values that each field could support or reject. Obviously, this is impossible. However, most data errors are found near the “boundaries of validity.” The two types of boundaries that you should test are

character boundaries and value boundaries.

Character boundaries are at the outer range of a character set. For instance, a numeric positive integer field in your program should only accept the digits 0 through 9. In order to test the field for character boundary mistakes, you would only need to test the characters 0 (hex \$30) and 9 (hex \$39) to make sure they are accepted, and "/" (hex \$2F) and ":" (hex \$3A) to make sure they are rejected. Similar boundary characters exist for uppercase alphabetic characters, lowercase alphabetic characters, printable characters, and so on. Make sure that Delete, Enter, and Return keypresses also work.

Value boundaries are boundaries determined by minimum or maximum character counts, or minimum and maximum values. For instance, if the maximum entry value in BuggyChecks is determined (by design) to be +/- 9,999,999.99, then we should test the amount field with +9,999,999.99 and -9,999,999.99 (to make sure these values are accepted), and +10,000,000.00, and -10,000,000.00 (to make sure these values are rejected either for length violation or for value violation). For good measure, also check 0 (zero), since this can sometimes produce "divide by 0" errors in a program. Another good check is to just press Return, not entering anything.

Don't forget to try completely illegal characters, such as Command-key sequences, the Escape key, function keys, and so on. Also, try entering characters before the program is ready for them (before a prompt appears). Press random keys and force random mouse clicks when the program is sorting, loading, or doing anything that causes the watch cursor to be displayed.

Just for fun, launch a commercial application and try some of the tests just mentioned. Randomly roll your fingers all over the keyboard. Chances are, you'll cause the application to crash or hang.

### **Our Checkbook Program Meets a Rich Customer**

In most cases, programmers write code that validates user input against some limit, such as +/- \$9,999,999.99. However, the entry limit and the internal limit (also known as an *internal boundary*) of the program are usually not the same. For instance, if a programmer uses long integers to store dollar values (using an implied decimal point), the value of the BuggyChecks application's internal limitation is approximately +/- \$24,159,191.03. This value is considerably higher than +/- \$9,999,999.99, so we should be safe, right?

Wrong! If keyboard entry is limited to \$9,999,999.99, then we won't have any problems with a single entry. But we will have problems if multiple entries sum up to be outside the +/- \$24,159,191.03 range. For example, if the user deposits three paychecks, each one being \$9,000,000.00 (this person is rich!) the total of the deposits would be \$27,000,000.00. This is outside the range of our poor little checkbook's storage medium (the long integer), and we get a system error.

This is an over-simplified example, but it proves a point. In order to claim BuggyChecks to be bug-free, we would have to test every possible number added to every other possible number, and every possible combination of keystrokes a user could ever type. A good test plan will have many test cases that exercise boundary values. A design or programming specification is extremely helpful to a test lead engineer in accomplishing this task.

### **Functionality Testing**

Functionality testing, the most common type of testing, is performed to make sure a program meets its specification. It's the tester's job to validate basic functions—that selecting "Save" from the File menu brings up a Save (or Save As) dialog box, and that pressing Command-Q exits the application. The best way to perform functionality testing is to go through a specification page by page to verify that the program does exactly what it's supposed to do. This applies to user manipulation of the program, data entry variables, and decision path execution.

Functionality testing, however, should go further than validation against the product specification. A program that accurately fits a bad specification is still a bad program. A good tester should keep a sharp eye out for things that don't appear "Mac-like." This is especially true for cross-platform applications that have been ported to the Macintosh platform. Things to watch out for range from the simple (is "Command-Q" shown next to the Quit command in the File menu?) to the archaic (does pressing Escape in a dialog box momentarily highlight the Cancel button and close the dialog box?). Apple's *Human Interface Guidelines* publication is an excellent reference for producing test cases of this type.

Here are some other kinds of functionality testing:

- Error-recovery testing concentrates on what happens after an error occurs. Does the same error message keep coming up on the screen in an endless loop? Can the program get back to a known, good state? Does a valid error message cause other, invalid messages to be displayed?

- Correctness testing is a form of functionality testing in which the program is audited to make sure it's producing the desired results. Do calculations produce

the proper figure? Do rounding errors occur? Does a file contain the expected sequence after sorting?

- State transition testing is another type of functionality test. The program is forced to go from one state through several other states in sequence. For example, a sequence could consist of loading a file, exporting it through Apple events to some other program, sorting the file, then closing it again. Do the states happen in the proper order? Can the states be forced to happen out of order through keyboard entry or mouse clicks, or through switching the program into the background? What happens when an unrecoverable error pops up in the middle of the sequence?

In tandem with functionality testing, some test teams perform documentation testing. As the name suggests, this testing verifies that the user manual reflects exactly what the program does, and the program does exactly what the manual says.

### **Configuration Testing**

Configuration testing takes a lot of time and equipment, and as such, it's the most expensive type of testing. Even companies with in-house test labs often hire outside companies to perform configuration testing. Your program must run on various combinations of machines, monitors, CPUs, networks, and hardware-related drivers. Various peripherals need to be attached and removed to see if the program supports them properly. For games or databases that require a CD-ROM, configuration testing should include various brands and speeds of drives and drivers to make sure they don't cause video or audio skip.

### **Compatibility Testing**

Compatibility testing, for the same reasons as configuration



testing, is usually left to outside test labs. For these tests, a program is run with various versions of the operating system, with various common INIT files, extensions, and control panels installed. Other compatibility tests include importing and exporting with other applications through the Clipboard, reading and writing of PC-format files, checking for support of industry standard sound, picture, and video formats, and testing for peaceful coexistence with other applications.

Application launchers, such as Apple's At Ease product, often cause problems when used with applications that are meant to be used alone. This is because the launcher patches system calls to prevent folder navigation, menu bar access, file deletion, or other things that could get a novice computer user in trouble. Unfortunately, these system patches are outside the control of the application programmer, and sometimes the patch makes it impossible to change an applica-

tion's default folders, preference file, and so on. Another common problem area is application compatibility when Apple's launcher utility is activated.

Virus protection software, such as Symantec's SAM, also should be checked for its effect on applications. This is especially true during the installation of a software package.

### Printer Testing

There are so many problems with printer output among various printer drivers that you should develop specific test cases for printing. Test cases should include laser printers, black-and-white and color ink jet printers, dot matrix printers, poster printers, slide printers, and so on. In many cases, what looks fine printed on a LaserWriter printer is unreadable when printed on an ImageWriter printer. This type of problem is most likely to occur in printouts that create forms or use reverse ink color (white on black, shading, and so on).

It's important to exercise numerous print options, such as print quality setting, start page, stop page, landscape and portrait orientation, and so on. Also, make sure that scaling the output up or down works properly.

### Stress Testing

Stress testing is where you force your program to operate under extreme conditions. Common types of stress tests include the following:

- *Low memory.* A program is launched with progressively smaller Finder memory partitions. When the program starts to get low on memory, a warning dialog box should be displayed instructing the user to save work and close windows. If memory gets too low for the program to continue operation, it should exit gracefully to the Finder.

- *Low storage space.* Conduct a series of tests with decreasing storage memory. Make sure a user can insert a floppy disk or some other media to prevent loss of data. DiskFiller is a commercial utility that can help you quickly (yet temporarily) deplete disk space.

- *Heap scramble.* The Mac OS is notorious for the Toolbox's memory handling—any time new memory is allocated, there's a chance that the application's heap will be reshuffled. MacsBug (available through Apple's *Developer CD Series* and various online services) can help find memory movement problems. Use MacsBug's HS (heap scramble) command or a commercially available third-party tool (such as Onyx Technologies' QC utility) to force memory to move around within the application. That way you can pinpoint any unlocked resources, pictures, sounds, or general memory handles that will cause bus errors after the items they point to are no longer there.

- *Write to nil.* This problem isn't as catastrophic as it once

was. However, a program should never write to location \$0000 in memory. This is usually a prelude to a more severe bug, such as disposal of unallocated handles or pointers. Again, use MacsBug's SS (Step Spy) command or third-party tools to trap this kind of behavior.

- *EvenBetterBusError.* This handy system extension is distributed by Apple on the Developer CD Series. This tool traps improper memory usage (including improper addresses, writing to nil, and so on) at a much higher level than normal program operation would produce. Caution: Many shipping applications don't survive very well when this extension is loaded and active!

- *DoubleTrouble.* This utility makes sure that a programmer doesn't dispose of resources, handles, or pointers more than once. It's handy for finding logic flow and decision path errors in code. "Double disposal" of memory is extremely hard to track down because the problem is never immediately apparent. Usually the double-disposal problems don't crop up until later in the program's execution, after the double disposal occurred. DoubleTrouble is also distributed on the Developer CD Series.

- *Multiple monitors.* Test each application at least once with more than one monitor connected to the test machine. Move windows around from monitor to monitor. Force update events. Zoom the windows in and out. These actions often produce problems, especially in programs that use offscreen ports and graphics worlds (gWorlds). Game applications are notorious for failing this test.

- *Monitor bit depth.* Randomly change the monitor bit depth through the Monitors control panel, and check the program for loss of color information. Be sure to check the 1-bit mode (black-and-white setting). Switching

## A Trick for "Trapping" Bugs

Here's a trick for invoking "heap scramble" commands, breakpoints, and other MacsBug commands early in a program's execution.

Almost all Macintosh applications initialize various Toolbox managers as one of the first steps after launching. To enter breakpoints and MacsBug commands early in the launch process, do the following:

1. Drop into MacsBug while still in the Finder (*before* launching your application).
2. Type the command "ATB InitMenus:g", which installs an A-Trap break on the InitMenus trap call.
3. You will be back in the Finder. Launch the program as usual.
4. As soon as the program calls InitMenus (usually one of the first few Toolbox calls in any application), MacsBug will be invoked.
5. Type in any breakpoints or commands you wish MacsBug to invoke.
6. To continue program execution, type "g", which means "go again."
7. When you no longer want MacsBug to break into every program you launch, drop into MacsBug one last time and enter "ATC:g", which stands for "A-Trap Clear."

monitor bit depth changes the memory requirements of a program, so be prepared for low memory messages or crashes.

- **Virtual memory.** Again, test in a fashion similar to your low memory series. These tests are more effective when used in conjunction with other stress conditions, such as heap scrambles.

- **Modern Memory Manager.** (Power Macintosh programs only.) Power Macintosh computers don't have the 32K segmentation limits that 680x0 machines have; thus they have a separate, optimized memory management system. For best results, do your tests with virtual memory turned on.

- **PowerPC vs. 680x0.** Even programs that aren't "fat" or PowerPC "native" can behave differently on Power Macintosh computers than on 680x0-based machines. Test applications on both types of computers and compare the results. You will most likely encounter more "bus error" and "illegal instruction" messages on Power Macintosh computers. Concentrate on the values of items that are defined

by the programmer as "unsigned," such as unsigned characters, short integers, and long integers. Some Power Macintosh compilers allow out-of-range (high bit set) values in these fields when they've compiled the application for 680x0 models, but produce strange results when they've compiled the application for PowerPC processor-based machines. To make matters worse, a programmer may not know that there's a problem, because the program compiles and links with no warnings!

#### Ad-Hoc Testing

Ad-hoc testing, also known as *mainstream usage testing*, is where a test engineer doesn't follow a predefined plan, but randomly runs through various portions of the program. This often uncovers bugs that may elude a strict, regimented test plan.

Experienced testers produce good results through ad-hoc testing because it gives them a chance to exercise a program in the areas where they feel it's weakest. Each

tester should be encouraged to do some amount of ad-hoc testing during each test cycle.

#### Performance Testing

Performance testing is any type of testing that is conducted in the hopes of identifying and eliminating speed bottlenecks. For instance, a program that requires communication with another program over a network should be tested in increasingly heavy network traffic conditions. At some point, traffic will interfere to the point where messages get lost or timed out. Once quantified, the condition can most likely be fixed by re-engineering hardware or code. In the worst case, the condition can be documented as a known limitation.

#### Software Production— More Art Than Science

Programming is not an exact science. Research by DeMarco and Lister shows that professional programmers average 1.2 software defects for every 200 lines of code. This means a typical software product, which contains about 200,000 lines of code, will (in theory) contain more than 1,000 programming errors. Studies conducted by Microsoft on larger projects show that it takes an average of 12 programming hours to find and fix a software defect!

Software testing is not an exact science either. No matter how much testing is done on a product, there is no way of knowing that all bugs have been found and documented. When multiple versions of the Mac OS, dozens of Macintosh models, hundreds of peripherals, and thousands of lines of code get together, there are millions of opportunities for error.

The trick to software testing is not to perform every test imaginable, but to perform the tests that will uncover the meanest, hardest-to-find bugs. I hope the information in this article has given you some insights into test strategies

that will help you reach your ultimate goal—bug-less software. ♣

*Victor Hnyp (victor@prosoft-labs.com) is president of Prosoft Labs, an engineering firm based in Pleasanton, California (510-426-6100) that specializes in the testing of Mac OS-based and Windows-based hardware and software.*

*A recommended source for further software testing information is the book *Testing Computer Software, second edition*, by Cem Kaner, Jack Falk, and Hung Quoc Nguyen (published by Van Nostrand Reinhold, New York).*

## Common Software Bugs

Error type	Percentage of errors
Content problem	17
Memory problem	14
Can't survive stress	14
Does not fit Apple GUI guidelines	9
Errors calling third-party libraries	8
Does not match specification	7
Dual monitor problem	6
Color table problem	5
Error recovery problem	4
Memory leak	4
Standard library errors	4
Other problems	8

# Listings

## Developer University Schedule

Developer University (DU) offers a broad range of Mac OS and Newton programming instruction through hands-on classes and self-paced training products. Classes are offered in Cupertino, California, and through selected third-party trainers.

The following is a list of upcoming DU course offerings, including when and where they're offered and how much they cost.

### Advanced C++ 5 days/\$1,000

March 4–8                      Cupertino, CA

### Apple Events/AppleScript Programming 5 days/\$1,500

January 29–February 2      Portsmouth, NH  
February 12–16              Cupertino, CA

### Creating Apple Guide Help Systems 4 days/\$1,200

February 12–15              Cupertino, CA

### Creating OpenDoc Parts 5 days/\$1,500

February 19–23              Cupertino, CA  
March 18–22                 Cupertino, CA

### Macintosh Debugging: Strategies & Techniques 3 days/\$900

January 22–24                Cupertino, CA  
March 13–15                 Portsmouth, NH

### Multimedia Development with QuickTime VR 3 days/\$900

February 20–22              Cupertino, CA  
March 19–21                 Cupertino, CA

### Newton Programming: Essentials 5 days/\$1,500

February 12–16              Cupertino, CA  
March 11–15                 Cupertino, CA

### Newton Programming: OS Enhancements 5 days/\$1,500

January 29–February 2      Cupertino, CA  
February 26–March 1        Cupertino, CA

### Programming with MacApp

On demand—call DU Registrar for more information

### Programming with QuickDraw 3D 3 days/\$900

January 15–17                      Cupertino, CA

### Programming with QuickDraw GX 4 days

On demand—call DU Registrar for more information

### QuickStart Mac OS Programming 5 days/\$1,500

January 29–February 2      Cupertino, CA  
March 4–8                         Cupertino, CA

### Scripting with AppleScript 2 days/\$600

January 22–23                      Cupertino, CA  
February 26–27                 Cupertino, CA

### Writing Reusable Code 3 days/\$900

February 5–7                        Cupertino, CA

To register for a class or to get a complete course description by fax, call the Developer University Registrar at 408-974-4897.

Course descriptions can also be found electronically at the following locations:

- **AppleLink:** Developer Support:Developer Services:Apple Information Resources:Developer Training:Developer University
- **eWorld:** Computer Center:Apple Customer Center:Apple Developer Services:Developer Information:Developer University
- **Internet:** <http://dev.info.apple.com/du.html>
- **America Online:** Computing:Computing Forums:Development:Mac Development Q&A:Developer University ♣

## The Internet Page

This feature is devoted to informing you about where you can go on the Internet for online information about Apple Computer, Inc.; its products, technologies, and programs; Mac OS and Newton programming; and other subjects that pertain to the business of computer product development. You'll find this feature particularly helpful when you view it at the *Apple Directions* Web page (located at <http://dev.info.apple.com/>). There, all the names of the locations listed in this article are linked to the sites themselves; clicking the names will take you directly to the relevant Internet locations. We'll update this feature every month, based both on what Apple is doing on the Internet and on your feedback.

### Apple Developer Services and Products

This section describes World Wide Web sites maintained by Apple Computer.

#### Apple Developer Services and Products

<http://dev.info.apple.com/>

This is probably the most important World Wide Web page for you. It contains Apple Directions Express with live links to other Internet locations and the online versions of *Apple Directions* and *develop*, the Apple Technical Journal. It also links you to a variety of other sites that give you access to the gamut of Apple's online developer support services.

#### Apple Directions Express Newsletter

<http://dev.info.apple.com/appledirections/adexpresscurrent.html>

Apple Directions Express is our biweekly e-mail digest of business news and information from Apple, sent to you biweekly over the Internet and posted at our Web site. It includes pointers—live links at our Web site—to Internet locations and other sources for more detailed information. We think of it as a front-end to Apple's huge Internet resource bank as well as other sources of business information about Apple.

Subscribe now by sending e-mail to [adirections@thing1.info.apple.com](mailto:adirections@thing1.info.apple.com). In the subject field—*not* the body of the message—type the string "subscribe <your real name>".

#### Guy Kawasaki's List Server

Guy calls his list server the EvangelList; it's for official and nonofficial Apple evangelists who want to hear and help spread the good word about Apple. For information on how to join, send an e-mail message to [macway-request@solutions.apple.com](mailto:macway-request@solutions.apple.com) for an automatic reply. (Any message will work.)

#### Apple Computer

<http://www.apple.com/>

This site contains links that will let you go to just about all the other Internet sites maintained by Apple, even the ones listed separately here.

#### Mac OS

<http://www.info.apple.com/macOS/>

You can go here for the latest information on the Mac OS, including details about Copland, white papers on new Mac OS technologies, marketing and strategic information, and other items to help you develop new Mac OS products.

#### Technotes

<http://dev.info.apple.com/technotes/Main.html>

This is the Web site for the Apple technical notes series. It contains all Technotes—new and old—as well as author's guidelines in case you want to contribute your own technical notes.

#### Apple Pacific

<http://www.info.apple.com/pacific/>

This page contains information about Apple offices and developer support in the Pacific region, including Japan, Australia, Canada, and Latin America.

#### Apple Europe

<http://www.euro.apple.com/>

This is the front door for information about Apple activities—including developer services—in Europe, with pointers to Internet sites for specific countries. Sites are currently established for Apple Norway, Apple Italy, Apple Germany, and Apple Benelux.

#### Newton

<http://dev.info.apple.com/newton>

Go here to learn about the Newton platform, including information about Newton 2.0—the new version of the Newton OS—and the more powerful suite of development tools that ship with it.

#### OpenDoc

<http://www.opendoc.apple.com/mainpage.html>

This site features the latest OpenDoc release, documentation, and tools, all available for free downloading, and Developer Showcase, from which you can download and sample actual OpenDoc parts! This is the place to go for the OpenDoc 1.0 SDK.

#### Third-Party Products

<http://dev.info.apple.com/thirdparty/>

Apple Fellow Guy Kawasaki set up this Web page to list your hardware and software products. Fill out the form located at the site to add your products; that way, everybody on the 'net can find out about what you're up to.

#### Apple Multimedia Program

<http://www.amp.apple.com>

If you're a multimedia developer or considering getting into multimedia, you'll want to check out the information on this page about Apple's multimedia technologies. It also includes the Apple Multimedia Program Member Showcase, a searchable database of multimedia developers.

#### The Macintosh Advantage

<http://www.apple.com/whymac/>

The official source for official Apple ammunition to fight the war against Windows 95, including the extensive series of Windows 95 vs. Macintosh Updates, prepared in the wake of the Windows 95 release.

**Macintosh PowerBook and Mobile Computing**

<http://www.info.apple.com/gomobile/>

This site contains complete information about PowerBook computers and the full line of Apple mobile computing solutions.

**PowerTalk**

<http://dev.info.apple.com/evangelism/powertalk/>

This site contains useful resources for PowerTalk programmers. It currently contains the StarNine gateways recently licensed by Apple: Mail\*Link Internet for PowerTalk, Mail\*Link MS for PowerTalk, and Mail\*Link QM for PowerTalk, which give Mac OS users access to Internet mail, StarNine Mail, or C.E. Software's QuickMail. You can download the gateways at no charge.

**QuickDraw 3D**

<http://www.info.apple.com/qd3d/>

This Apple site contains everything you need to know about QuickDraw 3D, including QuickDraw 3D applications you can "test drive."

**Power Macintosh**

<http://www.info.apple.com/powermac/powermac.html>

<http://www.info.apple.com/ppc/ppchome.html>

These are two useful sites for information about Power Macintosh computers.

**QuickTime**

<http://quicktime.apple.com>

This site contains news and technical and marketing information about QuickTime.

**QuickTime VR**

<http://qtvr.quicktime.apple.com>

You can find samples of QuickTime VR products here, as well as information on how Apple's virtual reality technology works and how you can incorporate it into your multimedia products.

**QuickDraw GX**

<http://www.info.apple.com/gx/gx.html>

Look here for information on QuickDraw GX as well as links to other non-Apple sites.

**Apple Education**

<http://www.info.apple.com/education>

Here's where you'll find information about Macintosh computers for the education markets. You can also use online forms located at this site to request product specifications, information about the Apple Education Series (bundled products), and technical support from Apple engineers.

**Macintosh Application Environment**

<http://www.mae.apple.com>

This site contains a sample of the Macintosh Application Environment (MAE), software that lets UNIX workstations run Macintosh applications, as well as technical information about the latest release, MAE 2.0.

**Pippin**

<http://support.info.apple.com/pippin/>

The Pippin Web page contains technical information about designing products that will work with Apple's PowerPC processor-based, low-cost CD playback device.

**eWorld on the Web**

<http://www.eworld.com/>

Go to this location to find content and services from eWorld, Apple's online service.

**Apple Internet Servers**

<http://www.apple.com/documents/otherappleservers.html>

Once you've exhausted the obvious Web sites just listed, this page will give you ideas about where else to go on the Internet to find the information you need. This page includes lists of other Web sites as well as Gopher and FTP sites.

**Non-Apple Sites**

We can't guarantee the information the following sites contain, since they're not created by Apple, but we think you'll find them useful and interesting. They're listed alphabetically, by site address.

**Internet Search**

<http://home.mcom.com/home/internet-search.html>

This site gives you access to InfoSeek, Lycos, WebCrawler, and other Web search engines. If you use the Netscape browser, you can reach this location just by clicking the Net Search button.

**Macintosh Vendor Directory**

<http://rever.nmsu.edu/elharo/faq/vendor.html>

This is a directory of companies that make and sell products for the Macintosh computer.

**Brad's WebSTAR/MacHTTP**

<http://www.ape.com/webstar/>

This site provides a database of all the Macintosh computer-based Web sites its owner can find, so far nearly 1,000 entries strong. It also lists outstanding Macintosh sites, as well as some losers that refuse to move to a Macintosh server solution.

**Macintosh Programming Tools**

<http://www.astro.nwu.edu/lentz/mac/programming/tools.html>

This site is a terrific source for Apple and non-Apple Macintosh programming tools.

**CI Labs**

<http://www.cilabs.org/>

Component Integration Laboratories—or CI Labs—is the industry consortium supporting the development of OpenDoc component software. Visit their home page for a great deal of OpenDoc content.

### MacTech Magazine

<http://www.class.com/MacTech/URLs.html>

This site contains a useful list from *MacTech* magazine of Internet locations on a variety of subjects, most of them pertaining to the technical aspects of Mac OS development.

### Nathan's Everything Macintosh Page

<http://www.cs.brandeis.edu/~xray/mac.html>

This is a treasure trove of Macintosh information; it contains a thorough listing of Apple and other corporate sites that pertain to Mac OS development as well as games, e-mail mailing lists, periodicals, a listing of FTP sites, software archives, and even Apple II information.

### Digitool (Macintosh Common Lisp)

<http://www.digitool.com/>

This site contains information on Macintosh Common Lisp (MCL) 3.0 and other MCL products, as well.

### The Ultimate Macintosh Page

[http://www.freepress.com/myee/ultimate\\_mac.html](http://www.freepress.com/myee/ultimate_mac.html)

This site contains more Mac OS information and software than you could possibly imagine exists. We think you'll particularly enjoy the software archives and games sites, from which you can download real-live software and play with it.

## New This Month/From Our Readers

This list contains Internet "stuff" (for lack of a more descriptive term) we've just become aware of, thanks to *Apple Directions* readers inside and outside Apple. Know of a particularly useful site? Whether it's a Web page, a list server, an FTP site, or a newsgroup, let us know about it and we'll consider adding it to this feature next month. Send your suggestions to the Internet address [a.directions@applelink.apple.com](mailto:a.directions@applelink.apple.com).

### Key Apple Developer Relations Contacts

<http://dev.info.apple.com/adrcontacts.html>

Intended mostly for use when standard Apple feedback mechanisms aren't working, this Apple site provides a list of key Apple Developer Relations contacts and how you can reach them.

### Apple International Developer Services and Products

<http://dev.info.apple.com/intl.html>

The Apple Web page for developers outside the United States, this site currently contains the current list of Apple international Apple Developer Services locations and contacts and a link to the Developer Services European FTP mirror site.

### Apple and the Internet

<http://product.info.apple.com/productinfo/tech/wp/internetwp.html>

This site contains the Apple Internet white paper, which describes what Apple is doing to open the Internet to Macintosh customers and why the Macintosh is the best platform for use with the Internet, both as a browser and a server.

### Apple FTP Sites

<http://dev.info.apple.com/ftpmain.html>

[ftp://ftp.info.euro.apple.com/Apple.Support.Area/Developer\\_Services](ftp://ftp.info.euro.apple.com/Apple.Support.Area/Developer_Services)

Go to these sites to download Apple software and documentation. The first address is for Apple's main FTP location; the second is a mirror site of the main location, maintained specifically for European developers.

### QuickTime Live!!

<http://live.apple.com>

This is Apple's site for showing multimedia broadcasts of live entertainment, including images, videos, sound, and QuickTime VR. The first event to be shown was Bill Graham Presents San Francisco New Year's Eve on December 31, 1995, featuring a variety of bands such as Santana and the Gin Blossoms. Future events to be shown here will include the Sundance Film Festival in January.

### The SALON Online Magazine

<http://www.salon1999.com>

To quote the Apple press release, "Some of San Francisco's top journalists have launched SALON—a new World Wide Web site that aims to bring lively, smart and engaging discourse about the arts, books, and ideas to the Internet." This literary/arts/cultural "e-zine" is sponsored by Apple Computer, Adobe Ventures L.P., and retail bookstore experts Borders, Inc.

### Mac\*Chat Newsletter

<http://www.cts.com/browse/xxltony>

This is an online newsletter directed primarily at Macintosh customers to help them make the best use of their Macintosh systems. You'll find it interesting, and your customers will find it even more interesting. Direct them to it as just another benefit of Macintosh ownership. Information about Mac\*Chat is available at the above Web site. To subscribe, send e-mail with the string "SUBSCRIBE MACCHAT" in the body of the message to [listserv@vm.temple.edu](mailto:listserv@vm.temple.edu).

### OpenDoc Part Ideas

[http://www.eng.uci.edu/~sroussey/NetVision/software/od\\_parts/](http://www.eng.uci.edu/~sroussey/NetVision/software/od_parts/)

This is a non-Apple repository of not only OpenDoc software but also ideas for software that could be implemented as OpenDoc components. You can read other people's ideas for OpenDoc components or submit your own. Visiting this page is a great way to feel like a part of the OpenDoc community.

**guideWorks**

<http://www.guideworks.com/>

This non-Apple site contains so much information about Apple Guide that you can think of it as the Apple Guide home-away-from-home page.

**PC Fairy Tales**

<http://www.icsi.net/~crfrank/newpcTales2.toc.html>

Check this page out! An enterprising, very pro-Macintosh NASA employee put it together to debunk common Macintosh myths. There's a ton of good data here to help you do the same.

**Kaidan Digital Imaging Solutions**

<http://www.kaidan.com>

Here's a reader-recommended site that will be especially interesting to

QuickTime VR developers. The site contains information about add-on lenses and QuickTime VR camera mounts for QuickTake cameras.

**Metrowerks**

<http://www.metrowerks.com/>

Go here for information about Metrowerks' CodeWarrior PowerPC development environment.

**User Group Connection**

<http://www.ugconnection.org/vendors/vendors.html>

This site contains resources and services to market your products to Apple's most influential and enthusiastic users: user groups. The site includes an online user group locator, a listing of user group Web pages, a monthly newsletter, and excerpts from a new book called *How to Market With Computer User Groups*.

**Pictorius**

<http://www.pictorius.com>

This is Pictorius Incorporated's Web site; it contains information about Pictorius Prograph CPX, the visual programming, native PowerPC environment for application programmers, and Peregrine, the Prograph-based environment for client/server database developers.

**Quinn's Human Interface Subtleties**

<http://redback.cs.uwa.edu.au/Quinn/WWW/HumanInterfaceSubtleties.html>

This non-Apple site lists the many human interface subtleties that continue to make the Macintosh user experience richer and easier. There's good ammunition here that you can use when anyone claims that some other operating system is just as good as the Mac OS.

**The QuickDraw GX Fan Club**

<http://www.ixmedia.com/quickgx/quickgx.html>

To quote the page itself: "The QuickDraw GX Fan Club has been set up to share information about QuickDraw GX and to encourage its use and development. There are pages devoted to each of the three primary elements of the GX technology: printing, typography, and graphics. If you haven't made the leap yet, there is a page devoted to reasons for using QuickDraw GX."

**Abacus Concepts, Inc.**

<http://www.abacus.com>

Check out this page to see how one of your colleagues has made good use of Apple Guide, Apple's intelligent online help architecture. Abacus is currently celebrating the tenth anniversary of its Macintosh statistics package, StatView; the latest release makes thorough use of Apple Guide.

**BroadCast Software Distribution**

<http://www.broadcastsoft.com>

BroadCast is a unique new software product that lets you distribute software products electronically. It's intended to give you a way to reach customers yourself without having to work with other distributors. Visit the

BroadCast Web site for information about the product as well as a free evaluation copy of it.

**Pages That Employ QuickTime VR**

The following pages use Apple's QuickTime VR cross-platform virtual reality technology to provide virtual reality experiences for their customers. Check them out to see some of the cool ways people are using QuickTime VR.

<http://pathfinder.com/time/special/baseball>

This site shows the 1995 World Series in QuickTime VR, brought to you by *Time* magazine. *Time* was the first QuickTime VR run-time technology licensee.

<http://sfasian.apple.com>

Here's where you can view a QuickTime VR version of the Asian Art Museum of San Francisco exhibit of Mongolian art, "Mongolia: The Legacy of Chinggis Khan."

<http://www.crowneplaza.com>

Holiday Inn uses this site to show QuickTime VR tours of two of its hotels, the Crowne Plaza Ravinia in Atlanta and the Crowne Plaza Toronto.

<http://www.honda.com/cars/odyssey/>

This site contains QuickTime VR panoramas showing the interiors of Honda Motor Company's new car models.

<http://www.gildamarx.com/>

Go to this site to see Gilda Marx Fashion Design's samples of QuickTime VR objects displaying its line of athletic body suits.

<http://www.interart.net/>

InterArt maintains this site, which contains QuickTime VR real estate tours of actual properties that are for sale.

### The Complete Conflict Compendium

<http://www.islandnet.com/~quill/c3data.html>

This site's owners, Quill Services Ltd. and MacSymum POWER Systems, have the goal of listing all software conflicts on the Macintosh computer and the cures for them. Forms at the site let you report conflicts and register your e-mail address so you can be notified if others report conflicts having to do with your products.

### MacHack

<http://www.machack.com/>

This site is maintained by the MacHack annual hackers conference, which features, among other events, an all-night hack contest, usually with very interesting results. At the 1995 conference, attendees decided on the ten issues they'd most like Apple Computer to address. You can find Apple's responses to those issues, results from the hack contest, and information about the next conference at this Web site.

### Macintosh Help Wanted

<http://www.memphisweb.com/mathew/default.html>

<http://www.memphisweb.com/nammac/default.html>

Need to find programmers and others to work on developing Macintosh products? Go to these locations for help. The first is called MATHEW, which stands for *Macintosh Talent, Help Wanted*. All companies and job seekers can post want-ads there for free, and the database is searchable by city, state, and expertise. The second site is a free directory of individuals and companies available for short-term tasks or contract work, also searchable by city, state, and expertise.

### Cult of Macintosh

<http://www.utu.fi/~jsirkia/mac/>

This is the site of another "everything Macintosh" compendium of information for Macintosh lovers. It contains dozens (maybe more than 100) links to other sites containing everything you'd ever want to know about the Macintosh computer, as well as sites with tools, free software, and so on.

### Why Should I Buy a Macintosh?

[http://www.dsu.edu/~bitzm/why\\_buy\\_mac/index.html](http://www.dsu.edu/~bitzm/why_buy_mac/index.html)

This site's name says it all.

### Douglas Adams

<http://www.umd.umich.edu/~nhughes/dna/stories/adamson95.html>

Go here to read Douglas Adams's highly subjective thoughts on Windows 95. Lotsa fun!

### Nisus Software

<http://www.nisus-soft.com/~nisus/>

We list this site partly because of its clever layout. The page looks like a Macintosh desktop; clicking the icons on the desktop takes you to Nisus's various Web postings. Just for fun, click the Trash icon and see where you end up! ♣

---

**APDA Ordering Information** To place an APDA order from within the United States, contact APDA at 800-282-2732; in Canada, call 800-637-0029. For those who need to call the U.S. APDA office from abroad, the number is 716-871-6555. You can also reach us by AppleLink at APDA or by e-mail at [APDA@applelink.apple.com](mailto:APDA@applelink.apple.com). More detailed APDA ordering information is available at the following locations:

- Internet: <http://www.info.apple.com/dev/apda.html>
- AppleLink: Developer Support:Developer Services:APDA
- eWorld: in the Developer Corner of the Computer Center