The Developer Business Report

December 1995

AppleDirections

Inside This Issue

Apple News

OpenDoc 1.0 for Mac OS Goes Golden Master!

Now Available on Internet; Over 300 Developers Commit to Using OpenDoc

By Gregg Williams, Apple Directions *staff*

On your marks . . . get set . . . GO!

OpenDoc 1.0, the open, cross-platform architecture for component software, is now Golden Master for the Mac OS platform, and everything you need to start developing with OpenDoc is available on the Internet.

Apple Computer, Inc., has fulfilled its promise to deliver the final version of OpenDoc 1.0 for the Mac OS in the fall of 1995 and has announced OpenDoc's availability worldwide, first at Apple Expo, in London, England (Nov. 8–11), then at COMDEX (Nov. 13–17) in Las Vegas, Nevada. In addition, Apple has announced its plans to introduce OpenDoc-J (the Japanese version of OpenDoc) at Macworld Tokyo (Feb. 21–24, 1996).

Apple is making an ongoing distribution of OpenDoc-related materials to developers—and anyone else interested in it—through the Internet. Eventually OpenDoc, documentation, sample code, development tools, and other materials will be available. On the Web, go to location http://coretools.apple.com/opendoc/ to find a pointer to them.

please turn to page 8

Strategy Mosaic

Newton—Surviving and Thriving at Age 2

By Gregg Williams, Apple Directions *staff*

As I'm fond of saying, a technology's growth does not occur with mathematical precision, like a straight-line path from point A to point B. Rather, it proceeds in a very organic fashion, with a struggle to survive when it's young, some changes as it adapts to its environment, and—inevitably—some life-threatening mistakes.

Apple's Newton technology is no exception, but I'm pleased to say that it has survived the scrutiny of the marketplace and, at age 2, is growing quite nicely. This article will summarize the improvements to the Newton platform, what Apple Computer, Inc., and its Newton licensees have learned about the true markets for Newton devices, and where the Newton platform may present business opportunities for you.

What's New With Newton

Apple has reshaped the Newton platform in direct response to real-world customer feedback, and the result is the next level of Newton technology, called simply Newton 2.0.

November was the month that Apple introduced Newton 2.0 to the world, at Macworld Australia and London, England's Apple Expo, and at COMDEX in Las Vegas in the United States.

please turn to page 5

AppleDirections

Volume 3, Number 12

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
Peter Bickford, Hadley Brandt, Dave Curbow, Alex Dosher, Elizabeth Dykstra-Erickson, Dean M. Gloster, Bob Megantz, Kerry Ortega, Caroline Rose, Geoff Schuller, Ken Seymens, Michelle Wyner

Manager, Developer Press
Dennis Matthews

Manager, Apple Developer Periodicals
Mark Bloomquist

Production Manager
Diane Wilcox

Prep and Print
Consolidated Publications, Inc., Sunnyvale, CA

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

Apple Directions Express

I wish I could get it sooner! By the time I get it, a lot of the news is old.

I'd like more copies to pass around! A single copy just isn't enough.

Those are the most persistent comments we get about *Apple Directions.* Given the limitations of printed publications—namely, the time and money it takes to print and mail—those have been tough problems to solve. I'm glad to say we've recently designed a solution, one that I hope will make *Apple Directions* significantly more useful to you.

The solution is *Apple Directions*' new free online component, Apple Directions Express, a digest of the latest news from and about Apple. It contains information about new products and technologies, strategic initiatives, market data, competitive analysis, market opportunities, among other subjects. At least every two weeks, the other *Apple Directions* editors and I select the most useful news items, write them up—usually with pointers to World Wide Web and FTP sites for more information—and send them to you for free in a brief e-mail message.

Instead of having to wait until nearly the end of the month for the paper version to reach you in the mail, you get the news instantaneously because it's sent electronically. Also, you can make as many copies—electronic or printed—as you want for whomever you'd like to give it to. We want this information to be sent out as widely as possible, because it's designed to broadcast the advantages of developing for the Macintosh computer.

While Apple Directions Express is supposed to broaden our reach, we've also designed it to make your life easier. You can look at it as the *Apple Directions* Read Me First file; each item is a paragraph or two long, giving you the meat of the news. You can then follow the links shown at the end of the item—many of them taking you to the electronic version of Apple Directions at our Web site (http://dev.info.apple.com/appledirections/adtoc.html)—to get more detail.

Apple Directions Express also lets us report more news. Often, we either run out of room in the paper version, or encounter a news item that will be so old by the time we release the paper version that it's not worth printing. Now, we'll include all those items in Apple Directions Express, and, if they're not going to be written up in that month's *Apple Directions,* we'll put more information on our Web page in a new folder called *Apple Directions Extras.**

As you can see, Apple Directions Express isn't just an electronic rehash of *Apple Directions;* it's a vital part of our effort to tell you everything you need to know about Apple. If you get Apple Directions Express, you'll still need *Apple Directions,* because we can't digest all the contents of the paper version into a concise e-mail. But, no matter what, I think you'll want to subscribe to Apple Directions Express.

Paul Dreyfus
Editor

P.S. Oh yeah, almost forgot: To start getting Apple Directions Express, just send an email to the address adirections@thing1.info.apple.com. In the body of your message, type the string "subscribe <your real name>."

* You'll also want to check Apple Directions Extras for material you won't find in the print version of *Apple Directions,* and for the occasional article we post there in advance of an issue's usual publication date.

December *Apple Directions* Online

December's *Apple Directions* will be available by November 15 at the following locations:

AppleLink: path—Developer Support:Developer Services:Periodicals:Apple Directions.
Internet: http://dev.info.apple.com/appledirections/adtoc.html
eWorld: in the Apple area of the Computer Center.

IndustryWatch: News & Perspective

Evangelical Words and Deeds

By the *Apple Directions* staff

Macintosh Computer Gains Market Share

Yes, you read that right.

Recent market data shows that the Macintosh share of the personal computer market has grown. Worldwide Macintosh unit shipments grew 19 percent faster than overall personal computer shipments during Apple's fourth 1995 fiscal quarter (Q4 '95)—that is, July through September. In its quarter-end financial results, Apple Computer, Inc., said that worldwide Macintosh shipments increased 25 percent over Q4 '94 shipments. (See the related news piece on page 11.) During the same period, overall personal computer shipments grew 21 percent, according to International Data Corporation, the Framingham, Massachusetts, market research firm.

Similarly, the number of Macintosh computers shipped in the United States grew more than twice as fast as overall U.S. personal computer shipments in the second calendar quarter of 1995. According to just-released U.S. Microcomputer Statistics Committee (USMSC) data, the total number of personal computers shipped in the United States between April 1995 and June 1995 increased only 5 percent compared with the same period the year before, while Macintosh U.S. unit shipments increased 12 percent. The actual numbers reported by USMSC are below.

Q2 '94 U.S. PC Shipments

	Q2 '94	Q2 '95	(Growth)
US PC shipments	4,590,000	4,833,000	(5%)
Apple USA shipments	468,000	524,000	(12%)

Source: USMSC

USMSC bases its statistics on actual sales data from 18 major U.S. computer vendors, including Apple. Shipments made by vendors who don't report their numbers to USMSC are estimated by an independent consulting firm, which bases its numbers on microprocessor shipment data, customer research, and channel surveys, among other data.

Implications/Opinions: These increases may be small, but we're impressed that even in the face of the release of Windows 95, which many analysts called the death knell of the Macintosh computer, Apple began to make good on its promise of increasing Macintosh market share. Who knows what will happen, now that the hype about Windows 95 has died.

Is PC Industry Growth Headed for a Slowdown?

A number of analysts have predicted that the recent fast growth of personal computer shipments will slow over the next 12 months. Dataquest, for example, recently released numbers during a teleconference with Apple representatives suggesting a downturn in PC unit sales growth in the United States, Western Europe, and Japan in 1996. Here's the Dataquest data:

Annual Growth in Total PC Sales

	1995	1996
United States	18.4%	14.8%
Western Europe	29.3%	19%
Japan	58%	37%
Pacific (outside Japan)	25%	27%

Source: Dataquest

Interestingly, Dataquest projects that PC sales in the Asian and Pacific region outside Japan will grow 25 percent this year, and increase even more—27 percent—in 1996.

Implications/Opinions: We can't say whether these projections will hold true, or if a slowdown in PC shipments translates into a slowdown in software sales, as well. Also, remember that predictions are for PC market growth to *slow down,* not for the market to shrink. However, two facts reported in this issue suggest that developing for the Macintosh will remain a safer investment than developing for other platforms: First, as reported in the first item in this column, the Macintosh share of the market is beginning to grow; second the Macintosh side of the market presents a better business opportunity than the Windows market, according to analysts at IDC. (See "IDC Report Shows Mac OS Better Business Proposition" on page 12.)

Intel Retreats (Slightly) With P6 Processor

Intel Corporation recently announced that its forthcoming P6 microprocessor, once the Pentium heir-apparent, will carry the Pentium name when it's released sometime by the end of this year. Intel will call its P6 product Pentium Pro, apparently in recognition of the fact that, for most customers, the chip does not deliver major performance enhancements over current Pentium processors. To take advantage of P6 capabilities, customers must use a 32-bit operating system (like Windows NT) and 32-bit applications. Windows 95 still contains a great deal of 16-bit code, and therefore does not run significantly faster on the P6 chip; currently there are very few available Windows-based 32-bit applications.

Implications/Opinions: Intel says that it will use the Pentium name for its P6 chip because of all the money the company has spent—and the success it's enjoyed—promoting Pentium as a brand; the company also says it will use the Pentium architecture for an entire family of processors. But the facts add fuel to Apple's claim that the x86 architecture is running out of steam, and that RISC is the future of personal computing.

The P6 makes the smallest jump in raw processing capacity over the previous x86 chip (that is, the Pentium) since the introduction of the x86 line. Measured in millions of instructions per second (MIPS), the P6—with a speed of 250 MIPS when running 32-bit software—is only 2.5 times faster than the Pentium (100 MIPS). The Pentium was 5 times faster than the 486 (20 MIPS), which was 3.5 times faster than the 386 (6 MIPS), which was more than 6 times faster than the 286 (0.9 MIPS). Also, according to test results released by Intel itself, P6-based systems running Windows 95 perform only 25 percent faster than Pentium-based hardware, while P6 systems running Windows 3.1 perform 16 percent slower than Pentium systems with the same clock speed.

When it was originally announced earlier this year, the P6 chip was intended to become the next major advance in Intel PC microprocessors. Now, it appears it's only a middling, incremental change that will mostly benefit high-end workstations and servers until Windows NT goes mainstream, or until Microsoft releases a fully 32-bit version of Windows 95. An Intel vice president suggested this might happen by mid-1997.

In the meantime, Apple will continue to build Macintosh computers based on the true 32-bit Mac OS and PowerPC RISC architecture, delivering systems that continue to build on Apple's current price/performance leadership. Also, before "Pentium Pro" chips are in widespread use, you can expect the release (by the end of 1996) of the first Apple computers based on the common hardware reference platform (CHRP) specification being jointly defined by Apple, IBM, Motorola, and a variety of other manufacturers. PowerPC processor–based CHRP systems will not only be powerful and utilize the latest technology, including PCI, but—since they'll run a variety of operating systems (Windows NT, in addition to the Mac OS, and OS/2) they'll also provide a very compelling alternative for people upgrading from Pentium/Windows 95 systems.

Speech Technology: Advantage Apple

You may have noticed lately that a variety of Wintel PC vendors have been showing off their systems' voice capability. For example, IBM uses its own technology to offer the most advanced Wintel PC speech capability with both VoiceType Control, for controlling the user interface, and VoiceType Dictation on higher-end models, for dictating text and numbers. IBM's recognition technology is also used in a CD-ROM title, *COMMAND: Aces of the Deep.* AST bundles third-party software with its computers for both user interface control and text-to-speech. Compaq's SpeechMail capability only allows remote retrieval of voice-mail messages over the phone. Microsoft has shipped a beta version of its speech software development kit for Win32; the final version is due in early 1996.

Implications/Opinions: Lest you be concerned, Apple still has the advantage when it comes to speech technology. Apple's PlainTalk 1.4 provides speech recognition software that runs on PowerPC processor–based Macintosh models and is bundled on the new Power Macintosh 7500 and 8500 models. With PlainTalk 1.4 speech recognition, customers can use spoken commands to execute common desktop tasks like opening files and folders, starting applications, switching applications, and closing windows. Speech recognition technology for Windows systems usually requires that users be trained to speak in a particular manner, pausing after each word. Mac OS speech recognition doesn't require you to speak in a halting manner, or use any other special speech techniques; if commands are spoken clearly, the Macintosh computer will recognize them.

Also, all currently shipping Macintosh computers support text-to-speech, or speech synthesis, which is used in a variety of education and entertainment applications. Using the soon-to-be-available Apple Speech Manager APIs (application programming interfaces), you'll be able to build text-to-speech technology into your applications with (in most cases) just a few lines of code.

On the PC, by contrast, speech does much less. The Microsoft Sound System, for instance, requires a hardware add-in board and can't truly speak text; it can only play back recorded voice clips, something the Macintosh has been able to do, with no add-in card, for ten years. Speech recognition with the Microsoft Sound System also requires add-in hardware and must be trained repeatedly before you can use it. Macintosh speech recognition requires no training.

For more information about Macintosh speech technology, see "The Mac Speaks Back" in the October 1995 issue of *Apple Directions,* which can be found on the World Wide Web at http://www.info.apple.com/dev/ or on AppleLink (path—Developer Support:Developer Services:Periodicals:Apple Directions). In addition, Apple's Competitive Analysis group has prepared a discussion of speech technologies as part of its "Macintosh vs. Windows" series. You can find it on the Web at http://www.info.apple.com/whymac/.

You can also download PlainTalk 1.4 from Apple's FTP site (ftp://ftp.info.apple.com/Apple.Support/Area/Apple.Software.Updates/%21Hot_Downloads/PlainTalk_1.4.1). Watch the Apple FTP site for the Apple Speech Manager, which you'll be able to download for free once it's available.

Compaq Exec Evangelizes the Macintosh Platform

The following event was reported in the October 9 Feedback column of *New Scientist* magazine. We're repeating it here in case you missed it.

The International Data Corporation recently held its annual European Information Technology Forum in Paris. Executives from Lotus, Oracle, Hewlett-Packard, IBM, Compaq, and Microsoft—including Bill Gates—all gave speeches. They all used computerized slides to aid their talks, but during several of the speakers' presentations, slides wouldn't change, or they'd change at the wrong time. When a Compaq senior vice president (who we won't name) got up to give his talk, he announced—according to *New Scientist*—that he "was not going to risk the same embarrassment."

Instead, he showed his presentation on a Macintosh computer.

Again in *New Scientist*'s words: "Compaq's Mac presentation progressed without a hitch. . . . It took a moment or two to appreciate the full significance of [it]. The vice president of a huge company that has built its success on selling PCs that run Windows was admitting to a top industry seminar that he dare not use a Windows PC for his presentation, and was using the rival Mac system instead."

Implications/Opinions: 'Nuff said. If you hear of other evangelical words and deeds like this, from unwitting sources, please let us know about them so we can consider them for future columns. Send e-mail to us at a.directions@applelink.apple.com. ♣

Strategy Mosaic

Newton

continued from page 1

Newton 2.0 will first enhance the Apple MessagePad 120, which will begin shipping in limited quantities in December, and Newton licensees are expected to offer Newton 2.0 compatible products in 1996. Current MessagePad 120 owners will be able to upgrade to Newton 2.0 through a ROM upgrade. Owners of previous MessagePad models will be able to take advantage of a rebate program in early 1996.

(For a summary of the user-level benefits of the Newton 2.0 platform, see the “Newton 2.0 Enhancements” box on this page and the “New Product Information” boxes on page7.)

Equally important is the fact that the Newton platform represents a better business proposition to you. This is the result of the following changes:

- removal of the 1 percent royalty for commercial Newton development (which was announced last September at the Second Newton Platform Developers Conference)
- significant improvements in and reduced costs for Newton development tools
- reduced prices on developer support (including the availability of direct technical support for much less money than before)
- a new modularity in the Newton architecture that enables third-party developer enhancement opportunities

Before I start talking about these points in detail, let me first describe what Apple now knows about the actual Newton markets and audiences.

Vertical Markets and In-House Solutions

The success of Newton devices in vertical markets and as part of in-house solutions, though invisible to the world at large, are very real. The Newton’s size, portability, built-in functionality, ease of use, and ease of development are among the factors that make the Newton platform an excellent choice for vertical markets and in-house solutions.

Numerous businesses have used Newton devices for functions that include mobile data gathering, sales support, and in-the-field reference. Salespeople, health care professionals, teachers, farmers, and repair people are among those who are using customized Newton devices to help them get their work done.

Horizontal Markets—Mobile Professionals

Apart from people who use Newton devices with custom software, many other people use Newton devices with off-the-shelf software. Apple has researched these users and has decided that the most succinct

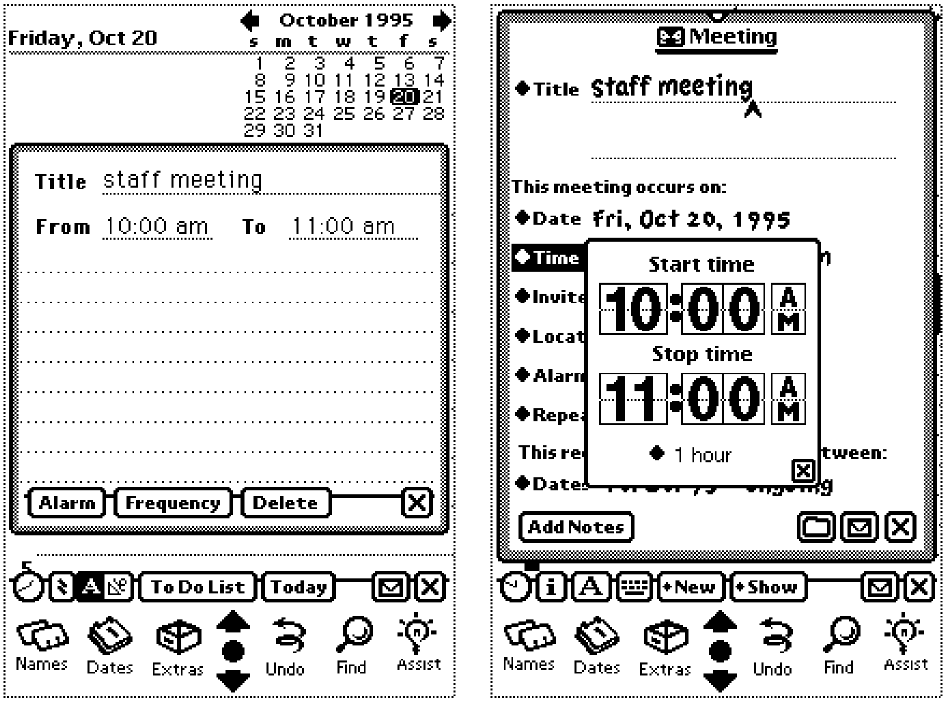
Newton 2.0 Enhancements

Here’s a list of some of the most exciting features that users will see in Newton 2.0 devices:

- *More functional built-in applications.* Most of the built-in applications have more features or have been redesigned to be more intuitive to use. For example, the Notepad now supports three different kinds of stationery—notes (as in Newton 1.x), checklists, and hierarchical outlines.
- *Better handwriting recognition.* The dictionary-based cursive handwriting recognition that was in Newton 1.x devices has been improved, partly through the addition of a larger internal dictionary. Newton 2.0 also includes a new, separate recognizer for printed letters (where the user picks the stylus up between letters). This printing recognizer recognizes individual letters and so is not limited to recognizing only those words that are in the dictionary.
- *New “comb-style correction picker.”* If a Newton 1.x device fails to recognize a word, you can double-tap the word and get a list of alternative word guesses. If none of those alternatives is correct, you have to type the correct word using an on-screen keyboard. To these options, Newton 2.0 adds a new correction picker that shows the spelling of the current word in a one-letter-per-box format. With this picker, you can make corrections by writing letters in individual boxes, which is more direct than finding and tapping individual keys in a keyboard layout.
- *Better human interface.* The Newton 2.0 human interface contains both subtle and obvious human interface improvements that make Newton 2.0 devices more intuitive to use. (For one such example, see the figure “What a Difference a Rev Makes” on page 6.) These range from better use of pickers (pop-up menus) and icons to a more visible text-insertion point in the form of a caret (^). Better yet, in free-form text situations (for example, the Notes application), any text you enter appears at the caret, not where you physically wrote it on the page.
- *Ink as a first-class citizen.* When you write something on a Newton device with recognition turned off, the Newton device stores it as “ink”—that is, as a series of writing strokes. In Newton 1.x devices, ink acts more like an arbitrary bitmap than it does as written information.

In Newton 2.0 devices, ink text is treated just like recognized text. For example, ink text is supported in virtually all input fields, and ink text “wraps” just like recognized text when you insert or delete words before it. Many users like ink text for “on-the-fly” notetaking and for situations where they don’t need the searchability that recognized text allows.

- *Faster NewtonScript execution.* Since this feature increased execution speed, Newton-based applications run faster and are more responsive to user input on Newton 2.0 devices.
- *Landscape mode.* The Extras Drawer now includes a Rotate button, which causes the Newton 2.0 device to switch between portrait mode (the only mode of Newton 1.x devices) and landscape (or “sideways”) mode. This works for all built-in applications and for any third-party application that you write (or rewrite) to respond to the Rotate button. Many Newton 1.x users have found it easier to read and write memos in landscape mode, and the extra width of landscape mode makes it feasible for Newton 2.0 devices to receive fax documents.
- *Keyboard input.* Apple will be selling a lightweight, full-size keyboard that attaches to any Newton 2.0 device and allows users to enter data through the keyboard. This, coupled with the landscape mode, makes it feasible for mobile Newton 2.0 users to do serious notetaking and letter writing while away from their main (or laptop) computer. In addition, when a Newton 2.0 device is attached to a desktop computer, users can similarly enter data into the Newton 2.0 device through the computer’s keyboard.



Friday, Oct 20

October 1995

Title staff meeting

From 10:00 am To 11:00 am

Meeting

Title staff meeting

This meeting occurs on:

Date Fri, Oct 20, 1995

Start time 10:00 AM

Stop time 11:00 AM

1 hour

What a Difference a Rev Makes. The human interface improvements between Newton 1.x and 2.0 devices number in the dozens, but this example gives an indication of how great the difference can be. The screen shot on the left is the dates slip (for filling out the details of a meeting in the Dates application) on a Newton 1.x device; the one on the right is the same dates slip on a Newton 2.0 device.

way of describing them is as "mobile professionals."

Mobile professionals are *not* a small target audience. In the United States alone, 71 percent of the country's 44 million professional workers (that's over 31 million people) spend more than 20 percent of their time away from their desks.

These people need what Newton devices have to offer. First, corporate "downsizing" has resulted in fewer support staff members, and Newton devices can provide many of the same services that support staff usually provide. Second, staying informed and in touch are essential to professionals' success, and Newton devices help with that, too.

Here are what Apple sees as the three main benefits of Newton devices to mobile professionals:

- keeping them organized (using applications such as the built-in address book, scheduler, to-do list, and In/Out Box)
- helping them communicate (in various ways, including the ability to send and receive fax documents and paging messages)
- making it easy for them to use their Newton devices in conjunction with their desktop Mac OS or Windows-compatible computers

Improvements to Development Tools

Now that I've talked about who Newton users actually are, it's time to talk about how Apple has improved the Newton platform. Though the many improvements that the user sees are important and exciting (see the box "Newton 2.0 Enhancements" on page 5), I'll concentrate here on improvements of interest to developers.

Newton Toolkit. The primary tool for serious Newton development is, of course, Newton Toolkit (NTK). NTK 1.5, released last summer, includes some much desired features, such as the ability to profile your program's performance and to compile critical parts of it (Newton programs are normally stored in NewtonScript, which is interpreted). Apple also made NTK more affordable by reducing its price to $299 in the United States (down from $799 for Newton Toolkit and Newton Book Maker). For more information, see the article "New Newton Toolkit Offers Compiler, Reduced Prices," in the September 1995 issue of *Apple Directions,* or read the article on the World Wide Web at location http://www.info.apple.com/dev/appledirections/sep95/newtontoolkit.html.

Newton Toolkit 1.6 adds new features to NTK 1.5:

- It supports Newton 2.0 devices.
- The Macintosh version is PowerPC processor "native."
- It offers additional support for debugging.
- It includes Book Maker, the Desktop Integration Libraries, and NewtApp (discussed below) in its U.S. $299 price.

The Macintosh version of NTK 1.6 will be available in December, and a beta Windows version will be available in the first calendar quarter of 1996.

Newton Book Maker. You can use this tool, available as part of NTK 1.6, to create electronic, hypertext-linked books, documentation, and reference materials, using little or no programming.

Desktop Integration Libraries (DILs). One frequent customer request is for the ability to transfer data directly between a Newton device and a favorite application (instead of doing so indirectly through the Newton Connection program). Apple can't do the work for you, but it can make the process of modifying your applications as easy as possible. To help you with this, Apple engineers created the Desktop Integration Libraries. For details, see "New Newton Connectivity" in the August 1995 issue of *Apple Directions,* or read the article on the World Wide Web at location http://www.info.apple.com/dev/appledirections/aug95/newsnewton.html.

NewtApp. Apple engineers have created the first software framework for building Newton-based applications. NewtApp is a body of NewtonScript code for Newton 2.0 devices that simplifies the creation of applications that collect, display, and edit soup-based data. ("Soups" are the primary data-storage structures of the Newton platform.) It's a good framework—Apple engineers used it to build the Newton 2.0 Names, Notes, and In/Out Box built-in applications!

Other Enhancements for Developers

In addition to the improvements in developer tools, the new Newton 2.0 platform contains enhancements that make your job easier and offer you new development opportunities:

• *Stationery.* If your application supports this new Newton 2.0 feature, other developers can add new features to your application. (The built-in Notepad, Names, and In/Out Box applications support stationery, which means there will be a market for stationery add-ons for those applications.)

Stationery makes two kinds of enhancements possible. First, it allows you to add new data types to an application—for example, you could add spreadsheet stationery to the Notepad. Second, it allows you to add new ways of viewing or printing existing data—for example, you could add additional ways of viewing data in the Names application (for instance, by name and e-mail address only).

• *Enhancements to views.* Newton 2.0 includes a number of new features for views. With Newton 2.0, you can create views that support the dragging and dropping of data between views. Since Newton 2.0 devices can now be used in either portrait or landscape mode, you should write (or rewrite) your Newton-based application to re-display itself when the user switches modes. In addition, Newton 2.0 includes new NewtonScript methods, messages, and constants that allow you to work more easily with your application's views.

• *Newton communications architecture.* Apple has appreciably improved the Newton communications architecture, making the communications code you write simpler and more versatile. For example, you can now add a Print or Beam command to your application in as few as six lines of code. Also, the modular communications architecture isolates you from the actual communication method used, so the code you write today will work with little or no change when used with new communication methods that become available for Newton devices.

• *Other platform improvements.* Newton 2.0 includes a number of new protos—the "building blocks" of code that you use to build a Newton-based application quickly. In addition, a new implementation of data soups operates faster than in Newton 1.x. You also have more ways to manipulate bitmaps in Newton 2.0, as well as more control over sound playback, handwriting recognition, and drawing. Finally, in the realm of communications, Newton 2.0 includes a significantly improved routing interface, as well as new transport and endpoint interfaces.

Why Develop for Newton?

Consider the advantage of Newton devices over both larger laptop computers and smaller fixed-function PIMs (personal information managers). You should develop for the Newton platform if you see a situation in which a Newton device offers the best solution.

Newton devices offer PIM, data-capture, and communications capabilities in a package that is lighter, easier to use, and more reliable than a laptop computer. In addition, Newton devices offer open-ended functionality (which PIMs can't) through the use of add-in software.

Newton devices can also communicate more richly than PIMs can. They can connect to online services and both send and receive e-mail, fax documents, and paging messages. Using the optional portable Newton keyboard and its landscape mode of display, people can use Newton devices to read and write business letters, which is one of the functions most often performed on notebook computers.

Opportunities for Developers

I can think of four broad opportunities for commercial developers. First is for you to develop Newton-based solutions for mobile professionals—especially applications that keep such users organized and productive or help them communicate. Second, you may find a business opportunity in an enhancement that hooks into the open-ended Newton 2.0 architecture.

A third opportunity is to use the Desktop Integration Libraries to give your customers an easy way to exchange relevant data between your application and a Newton device; Now Up-To-Date and Now Contact are two applications that already do this.

Finally, you may want to extend your desktop-computer-based application by using a Newton device. One good example of this is Pocket Quicken, a Newton-based application that Intuit developed to complement its

New Product Information

In upcoming months, Apple will release the following Newton-related products:

• *Apple MessagePad 120 with Newton 2.0.* This is the same Apple MessagePad 120 hardware as is available today, but with new ROM chips that implement the Newton 2.0 platform. This new version of the Apple MessagePad 120 will ship in limited quantities beginning in December 1995.

• *Apple MessagePad 120 upgrade.* Apple will also make a ROM upgrade to Newton 2.0 available to Apple MessagePad 120 owners. Owners will be notified of the details of this upgrade when they become available.

• *Newton Toolkit 1.6.* This new version of NTK contains all the features of NTK 1.5 plus additional debugging tools and support for the Newton 2.0 platform. The Macintosh version will be PowerPC processor native. A beta Windows version will be available in the first calendar quarter of 1996.

• *Newton Backup Utility.* This utility, which will be free to anyone who purchases or upgrades to an Apple MessagePad 120 with Newton 2.0, will allow Newton 2.0 device owners to perform Newton backup, restore, and package-install functions using their Macintosh computers.

• *Newton Press.* This Macintosh software program, available in December, will allow users to convert many different types of documents into Newton "books" that users can review on any Newton device. Newton Press makes this process very simple—users drag the documents onto the Newton Press icon; the program automatically translates the documents and downloads them into the Newton device. Apple hopes that this program will encourage many Newton owners to use their Newton devices (instead of a notebook computer) to review documents and batches of e-mail while away from the office.

• *Newton Keyboard.* This is a thin, lightweight keyboard that Newton 2.0 device owners can plug into the Newton device's serial port. Using this keyboard, Newton owners can enter text from the keyboard anywhere that the Newton device accepts handwritten input.

desktop-based Quicken money-management application.

The Newton platform also offers similar opportunities to in-house developers, system integrators, and consultants. If you work in one of these areas, you know that you need to provide a custom solution quickly—probably one that can easily be modified. The rich functionality of both the Newton platform and its development tools make development faster and easier than it is on any other PDA (personal digital assistant).

In addition, the Newton platform has the best developer support infrastructure of any PDA platform. Newton development training is available through Apple Developer University, and the Newton platform has more experienced programmers and consultants available for hire than any other platform.

In the United States and Canada, Apple also offers a variety of support services through its three developer support programs, the Newton Associates, Associates Plus, and Partners Programs (at $250, $500, and $1500, respectively, in the United States). Apple recently created the Associates Plus Program to offer limited one-to-one technical support to developers who do not need the full support of the Newton Partners Program.

The Newton Associates program is available in Europe, and the Newton Partners and Associates Plus programs are available on a limited basis in Europe.

For more information on these support programs, United States and Canadian developers can contact the Developer Support Center at 408-974-4897, AppleLink address DEVSUPPORT, or Internet address devsupport@applelink.apple.com. Developers worldwide should contact their local Apple offices to see what services are available.

Newton 2.0: A New Hope

I'm reminded of Williams's First Law of Technology Adoption: When contemplating the adoption rate of new technologies, think in terms of *half-decades.* Such is the case with the Newton platform. Though it has not caught on as Apple had hoped, it has survived the initial tests of the marketplace and is a better, more versatile platform for it. The Newton platform has more industry buy-in and infrastructure than any competing platform, and Apple is committed to supporting and extending it in the long term.

The Apple MessagePad with Newton 2.0 is a very elegant device, and as a Newton user (and erstwhile developer) of long standing, I can see that the Newton 2.0 platform brings out much of the promise that lay implicit but unrealized in the Newton 1.x technology. I look forward to seeing where developers like you will take it. ♣

Apple News

OpenDoc

continued from page 1

In addition, Apple will issue the Developer Release 4 OpenDoc CD beginning in January 1996. This CD will include OpenDoc 1.0, documentation, sample code, and sample components built by third-party developers. Apple will distribute this CD through Macworld Expo in San Francisco, the Apple developer programs, and other channels.

Developers Commit to Using OpenDoc

Though OpenDoc will not change the nature of personal computer software overnight, it will do so eventually. (For a discussion of the advantages of component software, see "OpenDoc *Is* Cross-Platform" in the November 1994 issue of *Apple Directions* and "OpenDoc Your Mind" in the December 1994 issue.)

However, over 300 developers see the advantages of OpenDoc and have committed to bringing out commercial or in-house OpenDoc components in 1996. Established companies that have committed to using OpenDoc include ACIUS, Adobe Systems, B&E Software, Berkeley Systems, Bowers Development (AppMaker), Casady & Greene, Inc., Claris Corp., DeltaPoint, Inc., Global Village Communications, Metrowerks, Novell, Oracle Corporation, Peachtree Software, Inc., Pictorius, Route 66, Symantec, Taligent, Inc., theta Group, and Virtus Corp.

However, it's important to note that roughly 25 percent of the developers who have committed to OpenDoc are small developers who believe that OpenDoc—especially in its beginning stages—will give them an opportunity to "stand out in the crowd," to establish themselves as experts in a new field, and to create a viable business from their areas of expertise.

Development Tools

You can develop OpenDoc components using any of the major C++ compilers. Apple also supplies you with several things to help:

- *PartMaker.* This program generates the source code for an "empty" OpenDoc component. PartMaker makes it much easier to create a component by freeing you from writing the code that is the same for every component. (For details on using PartMaker, see "OpenDoc Programming Made Easy" in the November 1994 issue of *Apple Directions.*)
- *OpenDoc Development Framework (ODF).* This is a framework that creates OpenDoc components, similar to the MacApp framework, which creates Macintosh applications. Unlike MacApp, however, ODF is a cross-platform framework that greatly simplifies the creation of separate Mac OS and Windows-compatible components from the same source-code base. Apple has been supplying developmental (that is, pre–Golden Master) releases of ODF and will continue to release improved versions.
- *Container-Application Library (CALib).* This library of code makes it easier for you to convert your application into a container application—that is, a conventional application that creates documents that can accept OpenDoc components. CALib helps you convert existing applications to participate in the world of OpenDoc components.
- *Source code.* Apple is supplying a lot of sample code that illustrates many aspects of creating different kinds of components.

In addition, other companies will be creating development tools that will give people other ways of creating OpenDoc-related software. Among those are the following:

- In January, the CodeWarrior CW8 CD will contain a preliminary version of new PowerPlant classes that aid in the construction of OpenDoc components. (PowerPlant is Metrowerks' software development framework. You can currently use PowerPlant to create components, but these new classes will make it easier to do so.)

• Oracle Power Objects 1.1, due in early 1996, will generate applications that will be component applications.

• Pictorius, the company that makes the Prograph CPX visual, object-oriented development environment, says that it will enhance Prograph CPX to produce OpenDoc components as well as stand-alone applications.

What You Should Do

Because of the advantages of component software and the new business opportunities it opens, OpenDoc will become increasingly commonplace in software development in the years to come. The question is not *whether* you will adopt OpenDoc, but rather *when* and *for what.*

During the past two years, *Apple Directions* has carried various articles that stressed the necessity of your becoming more knowledgeable about OpenDoc, so that you can make an informed decision about how it fits into your company's future. Now that OpenDoc 1.0 has gone Golden Master, you have no excuses for not doing so. Here are some things you should consider doing:

• Download OpenDoc and documentation from the World Wide Web.

• Write your first sample parts using ODF, PartMaker, or sample code.

• Look at the sample code that Apple supplies to see how a seemingly monolithic application can be implemented as a suite of OpenDoc components.

• Subscribe to the opendoc-interest newsgroup, which is a valuable channel for information on and discussion about OpenDoc. To do so, send e-mail to Internet address listproc@cilabs.org, with the message "subscribe opendoc-interest <insert-your-name-here>" in the body of the message.

• Use CALib to make your current applications into container applications.

• Consider working with Apple. If you're planning an OpenDoc product for commercial or in-house release in 1996 and are not in Apple's database of OpenDoc developers, send an e-mail to opendoc@apple.com. Apple evangelists may contact you if there are any possibilities of Apple comarketing or demonstrating your product.

Apple will begin shipping OpenDoc with Macintosh computers in 1996 and has plans for providing some exciting new business opportunities centered around OpenDoc. We hope you'll start using OpenDoc now, so you'll be ready for the opportunities that will follow in 1996 and beyond.

Apple, IBM, Motorola Announce CHRP

Last month, at COMDEX in Las Vegas, Apple, IBM, and Motorola jointly announced the availability of the CHRP (Common Hardware Reference Platform) hardware specification, which will allow any hardware vendor to create computers that will run the Mac OS and several other operating systems. Details were not available when this issue went to press, but you can find more information on the World Wide Web by going to the Apple Directions Express "current" page, at location http://www.info.apple.com/dev/appledirections/adextras/adextras.html.

Apple to Move Communications to Internet

Apple recently announced that it will be moving its communication and information system directly to the Internet. This means that Apple Developer Relations will not be making the transition to eWorld at this time, as previously planned.

The company's Apple Online Services organization, reorganized under the new name Apple Internet Services (AIS), will continue to develop and market eWorld, moving it toward a more Internet-based service and expanding its current 115,000 subscriber base. Apple will continue to bundle eWorld with its computer systems.

Put as concretely as possible, here's what this news means for developers:

• You'll no longer be required to use AppleLink if you're a member of the Apple developer programs; you can use the online system of your choice. As long as you keep Apple informed of your e-mail address, you'll continue to have the same level of communication with Apple you've experienced until now.

• Apple will not be converting AppleLink accounts to eWorld. (And AppleLink services will not go away by the end of the year.)

Apple's Developer Relations recently posted a message on AppleLink, eWorld, and the Web explaining the current details of Apple's move to the Internet and how it affects you. We've reprinted the message below, with the details that most affect you.

Dear Developer,

By now, you are probably aware that Apple Computer has decided to take our communication and information system aggressively to the Internet. We think this is the right long-term strategy, as the rest of the world is also moving aggressively to the 'Net. While we initially envisioned a communications migration from AppleLink to eWorld to the Internet, we have decided to accelerate the move directly to the 'Net. This means that we will not complete the migration from AppleLink to eWorld for Apple employees and our business partners. We wanted to clarify for you what this means to you as a developer.

First of all, you are not required to have an AppleLink account to be a member of Apple's developer programs. You can choose the online system that best fits your communication needs.

Secondly, we have been working hard to make most developer-related materials available on both the Internet and AppleLink. And, of course, we also deliver confidential information, tools, and system software on your Developer CDs.

So, depending on your needs, you can get timely developer information from Apple by making one of the following choices: 1) keep your AppleLink account, and we will continue to update the Developer area until our long-term Internet solution is available, 2) keep/join eWorld, where there is access to the Internet from the Web browser, or 3) choose another online service that provides you access to the Internet through a Web browser.

You may have other eWorld/AppleLink-related questions. Some are specific to being a developer and others may relate to Apple's overall online strategy. We have tried to bring together the issues we think are most on your mind in a Question and Answer format at the *end of this link. Read on for the details.*

Sincerely,
Apple Developer Relations

Q: If I decide to drop AppleLink, how do I inform the Developer Group of the new e-mail address I plan to use to communicate with Apple?
A: We would prefer for you to let us know your e-mail address. If you are a U.S. or Canadian developer, send a message to DEVSUPPORT from your new e-mail account. Be sure to identify your company and, if possible, your old e-mail address. This will allow us to provide faster turn-around on subsequent e-mails arriving from your new ID, and for you to receive group communications and updates in the future. If you are a developer in another country, please notify your local Apple office via e-mail of any changes.

Q: What will I miss from Developer Relations if I drop AppleLink and go to the Internet?
A: We have been working hard to make most developer-related materials available on the Internet, including the Hardware Purchase Program price list. We will continue to aggressively move the remaining developer content. (The majority is already on the Developer CD Series.)

Q: I have kept my AppleLink account mostly to access the Hardware Purchase Program price list (U.S.). Where can I get it on the Internet?
A: It can be found at the address http://www0.info.apple.com/dev/ddph.html. Call Developer Relations at 408-974-4897 for an ID and password.

Q: What will I miss from the Developer Group if I drop AppleLink and migrate to eWorld?
A: You will find on eWorld the Developer Corner, a public forum for Macintosh developers to meet and discuss Macintosh-related developer issues. All of the content that is on the Internet can be accessed through eWorld's Web browser. It is important to note, however, that you will not be able to send enclosures (disk files) to DEVSUPPORT or other AppleLink users directly from eWorld.

Q: I still want to join eWorld as a regular consumer. Can I now get my AppleLink ID as my eWorld screen name?
A: We "reserved" AppleLink IDs on eWorld in order to facilitate the migration. Now that we're foregoing the move to eWorld as our official business communication system, we will release the reserved AppleLink IDs; we'll let you know shortly exactly how and when that will happen. If you decide to join eWorld, there's a good chance that you'll get to use the same name; however, there are no guarantees, as names are assigned on a first-come, first-served basis.

Q: What should I do if I already migrated to eWorld in anticipation of the official migration?
A: We'll continue to develop a seamless integration between the eWorld "Town Square" and Apple's presence on the Internet. As a member of eWorld, you'll be able to take full advantage of all that eWorld and the Internet have to offer, including Apple's developer content on the Web. It is important to note, however, that you will not be able to send enclosures (disk files) to DEVSUPPORT or other AppleLink users directly from eWorld.

Q: If I am a new developer and I believe that I need AppleLink, how do I get a developer AppleLink account?
A: To set up a new AppleLink ID that is coded for developer visibility and qualifies for discounted developer pricing, you should fill out a Developer AppleLink Account Request Form (DAARF). You can obtain this form by calling Developer Relations at 408-974-4897 or by sending an e-mail to devsupport@ applelink.apple.com. For international developers, the form is in your application packet.

Q: Does Apple have any products that will assist me in getting connected to the Internet?
A: Yes, the Apple Developer Tools catalog sells the Apple Internet Connection Kit. This kit brings together all the applications you need to access the Internet directly including Netscape Navigator, Apple Internet Dialer, eWorld, Adobe™ Acrobat Reader, Aladdin Stuffit Expander, Fetch and Anarchie, E-Mail software and NewsWatcher. To order the Apple Internet Connection Kit call 800-282-2732 (U.S.), 800-637-0029 (Canada), or 716-871-6555 (international).

Q: Will Developer Relations still be updating information on AppleLink? on eWorld?
A: For the short term, we will continue to update information on AppleLink and our public postings on eWorld. We will evaluate this over time.

Q: Will there be a pricing reduction for external AppleLink customers?
A: There is no anticipated change at this time; however, we'll continue to pursue methods of reducing our costs and pricing and will let you know at a later date if there is a change to the current AppleLink pricing.

Q: Will Apple employees migrate off of AppleLink eventually?
A: It is still important for Apple to seek a more effective and innovative approach for official business communication using the Internet and World Wide Web. For now, we'll continue to use AppleLink.

Q: How long will it take to move to a different solution? When will it be ready? What will it look like?
A: This question can be answered later in 1996, subject to planning efforts now underway at Apple. For the latest information check our World Wide Web page at http://www.info.apple.com/dev/developerprograms.html.

Q: Where can I get more information or ask questions about this?
A: You may write to ASK.EWORLD on AppleLink if you have questions that aren't addressed in this or other documents that are posted on AppleLink. If you have questions specifically about developer-related issues, then you may write to devsupport@applelink .apple.com.

Apple Announces Record Quarter

So much for the idea that Windows 95 would wipe the Macintosh computer off the face of the Earth. Between July and September this year, not only did Apple Computer, Inc., set new revenue and Macintosh shipment records, it also gained market share. (See the first item in IndustryWatch on page 3 for more about the market share story.)

In Apple's fourth fiscal quarter, 1995 (Q4 '95)—that is, July to September—the company posted record revenues of $3.0 billion dollars, 20 percent more than the $2.49 billion revenues the same period a year ago. Customers purchased more than 1.25 million Macintosh systems during the quarter, a 25 percent gain over the same period of the year before. For the year, profits were $424 million, up 37 percent over fiscal year 1994, and total revenues were $11.1 billion, up 21 percent from total fiscal year 1994 revenues. Computer sales outside the United States accounted for a larger part of Apple's revenues than they did in fiscal year '94; 47.6 percent of net sales came from outside the United States, up from 45.7 percent the year before.

"Demand for Apple products has been stronger than we've ever experienced in the company's history," said Apple CEO Michael Spindler. "In response, we have ramped unit shipments to record levels. Our better than 25 percent increase year-over-year in unit shipments during the quarter is our best in two years."

The company's increase in revenues and unit shipments for the quarter was driven by sales of Power Macintosh systems. Total shipments of Power Macintosh products accounted for more than 70 percent of units shipped during the quarter.

Apple Guide Now Supports Mac OS Systems 7.0, 7.1

The latest version of Apple Guide, the Mac OS technology for creating intelligent online help systems, supports system software versions 7.0 and 7.1. Apple Guide 2.0 was just made available at Apple's FTP site and on AppleLink; you can download it for free. The new version will let you create guides that can be used by a larger market. It's also faster than previous versions, and provides a variety of bug fixes and other enhancements.

For more information about Apple Guide, read "Apple Guide—Intelligent Help For Your Business" on page 25 of the October 1995 issue of *Apple Directions,* which you can find on the Internet at http://dev.info.apple .com/appledirections/october/.

You can download Apple Guide 2.0, along with Apple's licensing agreement, from the following locations:

- Apple FTP site—ftp.info.apple.com/Apple. Support.Area/Developer_ Services/System_Software_ Extensions/Apple_Guide_ Authoring_Kit/
- AppleLink—Developer Support:System 7.5 Talk:Apple Guide Discussion

To help your Apple Guide development efforts, Apple has also posted at the FTP site the latest version of Guide Maker, the application you use to compile and test Apple Guide files.

Apple Customers Most Loyal in Industry

A recent International Data Group (IDG) study determined that Apple customers remain the most loyal in the industry. According to the study of both home and business users, Apple and Hewlett-Packard tied for the lead among customers who were asked if they would repurchase their current systems. Of the Apple and H-P customers participating in the study, 70 percent said they would be "very likely to repurchase" their systems.

Macintosh customers proved themselves the most passionate about their computers, 39 percent of them describing themselves as "missionaries" for the brand. Hewlett-Packard placed second in this category, with 25 percent of their customers saying they were missionaries.

Other results of the study include the following:

- Packard-Bell customers were the least loyal of those studied, only two of five saying they would be very likely to buy Packard-Bell systems again.
- IBM now has more brand power in the home than in the business market.

You can obtain a copy of the study findings by calling 415-676-3030 and asking for the *Brand & The Buying Process* study.

IDC Report Shows Mac OS Better Business Proposition

According to a recent report by International Data Corporation (IDC), *The Opportunities and Cost for the Macintosh Market,* the Mac OS market offers developers higher profits and less competition than the Windows market.

"The major costs in R&D are head count and capital equipment," said IDC senior vice president John Gantz. "And while these costs tend to be the same for each platform, the results [of this report] show that the larger number and diversity of peripherals and machines in the Wintel environment make secondary costs, such as testing and technical support, higher than for the Macintosh."

Other key points from the report were the following:

- The introduction of the PCI bus into the Mac OS architecture and the availability of Macintosh-compatible machines will ensure that the Mac OS platform base will grow as fast as the Windows platform base; each installed base will double in size over the next five years.
- Because Mac OS customers typically buy and use more software packages than Windows owners and those packages are generally sold at higher prices, Mac OS software generates almost 75 percent more software revenue per machine than Windows-based software.
- In a survey of 70 hardware and software cross-platform developers, IDC found that because software and hardware for Windows-based computers are more difficult to test and support than software and hardware for Mac OS computers, development costs for Windows-based products average 50 percent higher per dollar of revenue.
- Because of the need to advertise in more places to reach particular audiences and the extra costs of supporting two-tier distribution, marketing costs for Windows-based products average 54 percent higher than for Mac OS computers.

IDC is a leading provider of computer industry analysis. For more information on *The Opportunities and Cost for the Macintosh Market* or any of IDC's other market-research services, contact Linda Rich at 508-935-4389.

Report Confirms Lower Support Costs for Macintosh Platform

If you need ammunition to support the use of the Mac OS platform (in your own or a client's company), a recent survey of technical support costs for corporate computing just completed by Gartner Group Consulting Services came to two conclusions that you can use to support your argument:

- First, companies supporting both Mac OS and Windows-based computers did not have higher support costs than companies supporting Windows-based computers alone.
- Second, companies using Mac OS computers had lower support costs than companies using Windows-based computers.

According to Dr. H. Walter Johnson, Gartner Group vice president of Consulting Services and author of the study, "The major conclusion [of the report] is that, while conventional wisdom might suggest that standardization on a single platform would lower technical support expenses, there is no premium associated with supporting both Macintosh and Windows."

Other conclusions offered by Gartner Group include the following:

- The higher the percentage of Mac OS computers, the lower the technical support costs.
- Support costs for Mac OS computers are approximately 25 percent lower than those for Windows-based computers.
- On the average, fewer than three days of training per user per year tends to increase support costs per user.

This data comes from a report entitled *Technical Support Costs in Dual-Platform Computing Environments: Managing Diversity.* The report analyzes data collected from a broad range of U.S.-based companies with varying mixes of Mac OS computers (up to and including Mac OS version 7.5.1), and Windows-based desktops (up to and including Microsoft Windows version 3.11). The research was conducted in the first half of 1995, prior to the public release of Windows 95.

Apple Takes Commanding Lead in Education Market

Quality Education Data (QED) has great news for you if you're an educational software developer. According to the education market research firm's new study, *1995–96 Technology Purchasing Forecast,* 63 percent of the computers used in U.S. public schools are Apple-branded. Apple's share of school PCs grew four full share points in the past year, up from 59 percent.

The news is even better if you specialize in educational multimedia products. Today, 77 percent of all school districts currently using multimedia computer systems use Macintosh systems, according to QED.

The QED report also predicts that Apple will continue to be the leader by a wide margin in the education market. School districts reported that of the systems they plan to purchase in the 1995–96 school year, 58 percent of them will be Macintosh computers. This is more than three times the runner-up in this catetory—IBM—whose systems educators say will make up 18 percent of their purchases during the current school year.

Non-named brands make up the third largest segment of planned purchases, at 12 percent. Planned purchases for other name-brand systems were lower, including Compaq at 7 percent and Dell at 4 percent. The average district plans to buy 28 Macintosh computers, 8 IBM computers, and 10 other Windows/Intel systems during the 1995–96 school year.

The QED research also shows that educators are embracing the power of PowerPC microprocessors over Pentium. Districts reported that 23 percent of the computers they'll purchase this year will be Power Macintosh systems, while 8 percent of their purchases will be Pentium-based systems.

You can order copies of the QED report by writing to Quality Education Data, 1600 Broadway, 12th Floor, Denver, CO 80202, or by sending e-mail to qedinfo@qeddata.com.

Apple Licenses StarNine PowerTalk Gateways

Apple recently licensed several PowerTalk gateways for System 7.5 from StarNine Technologies and is making them available for no charge over the Internet. The StarNine gateways include Mail*Link Internet for PowerTalk, Mail*Link MS for PowerTalk, and Mail*Link QM for PowerTalk, and they give Mac OS users access to Internet mail, StarNine Mail, or C.E. Software's QuickMail. If you've incorporated the PowerTalk Mailer into your application, your customers will be able to communicate using those mail systems from within your application (if they're using System 7.5 with the gateways).

Apple released PowerTalk in October of 1993, and it's now an integral part of Macintosh System 7.5. With PowerTalk, Apple was the first operating system vendor to provide built-in e-mail and collaboration services. PowerTalk provides Mac OS users with a universal electronic mailbox to most communication systems they may use. In addition, PowerTalk allows users to send mail from within any PowerTalk-enabled application, digitally "sign" documents, and share files over an AppleTalk network. When coupled with PowerShare, Apple's e-mail and directory server, PowerTalk becomes a powerful electronic collaboration system for workgroups.

Currently Macintosh System 7.5 includes PowerTalk personal gateways for direct dial-up, fax, pager, and CompuServe, in addition to demonstration copies of the StarNine products. The new StarNine modules include the following:

- Mail*Link Internet for PowerTalk, which enables customers to access their Internet or UNIX® mailbox using SMTP and POP3 protocols
- Mail*Link MS for PowerTalk, which enables users to access their StarNine Mail or Microsoft Mail for AppleTalk mailbox and directory
- Mail*Link QM for PowerTalk, which enables users to access their QuickMail mailbox, address books, and name server

The gateways will be included in future releases of the Mac OS.

Until then, you can download them from Apple's PowerTalk Web site (http://dev.info.apple.com/evangelism/powertalk/). Additional information about the gateways can be found at StarNine's World Wide Web site (http://www.starnine.com).

QuickDraw 3D Gets Faster

Underscoring its commitment to QuickDraw 3D graphics software—and making it easier to develop QuickDraw 3D graphics—Apple Computer, Inc., has introduced the QuickDraw 3D Accelerator Card. The Peripheral Component Interconnect (PCI) card, coupled with QuickDraw 3D, accelerates 3D performance up to 12 times on Power Macintosh systems. If you develop for the game or technical markets, or if you develop multimedia products that rely on 3D images, the card will give you workstation-quality 3D graphics features and performance on the comparatively lower-priced Power Macintosh computer.

QuickDraw 3D is Apple's cross-platform application program interface (API) designed to take advantage of PowerPC RISC technology for creating and interacting with electronic 3D graphics and animations. The Accelerator Card enhances this capability enabling 3D developers and other customers who create or work with 3D images to work in real-time with much larger and more complicated models. The 7-inch QuickDraw 3D Accelerator Card plugs into the PCI slot of any PCI-based Power Macintosh or Macintosh-compatible computer running Mac OS version 7.5.2 or later and system enabler 1.1 or later with 16 MB of memory. The card does the following:

- cccelerates QuickDraw 3D rendering so you can accomplish more in less time
- renders 10 million trilinearly filtered, mip-mapped texture pixels per second and up to 120 thousand triangles per second
- accelerates Gouraud shading, texture mapping, transparency, constructive solid geometry, and other imaging techniques
- utilizes trilinearly filtered and mip-mapped textures at all times for crisp object definition and high-quality resolution, making object edges appear defined and realistic at any scale
- includes high-performance SRAM memory with the ability to display up to 12 texture maps simultaneously in each open window
- outputs an alpha channel, allowing you to quickly transfer images from one application to another
- provides per-pixel object sorting for easier manipulation of realistic-looking transparent objects
- accelerates multiple frame buffers, eliminating the need to buy a separate acceleration card for every system frame buffer

You can install multiple cards for added performance of rasterization-intensive tasks; two cards installed in the same system can work up to twice as fast as can one card working alone. The card doesn't require the use of video connectors, and it works with any application that supports QuickDraw 3D.

The Apple QuickDraw 3D Accelerator Card is expected to be priced at an Apple Price of $399 (U.S.), and it's planned to begin shipping in volume by the end of November. Details about QuickDraw 3D development, and how you can get started, can be found in "QuickDraw 3D—The Future is (Virtually) Here" in the October 1995 issue of *Apple Directions,* which you can find on the World Wide Web at http://dev.info.apple.com/appledirections/oct95/gamedeveloper.html. ♣

==> ==> ==> ==> ==> ==> ==> ==> ==> ==> ==> ==> ==> ==> ==> ==> ==>

==> ==> APPLE DIRECTIONS EXPRESS ==> ==>

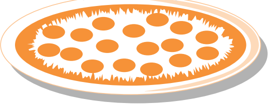
==> ==> ==> ==> ==> ==> ==> ==> ==> ==> ==> ==> ==> ==> ==> ==> ==>

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 on-line business bulletin from Apple Computer, Inc., and the editors of *Apple Directions.* At least every other week, we send you a single, concise email 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.
In the body 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>

Technology

Inside This Section

develop Issue 24: Something for Everyone

Issue 24 of *develop,* Apple's technical journal, contains articles on subjects at all levels of complexity, ranging from simple topics (such as the design of alerts) to the more esoteric (like device data transfer). Whether you're new to developing for the Apple platform or an old hand at it, you should find something interesting and helpful in this issue. It contains the following articles:

- "Speeding Up whose Clause Resolution"—Although the Object Support Library will resolve complex AppleScript clauses for you, if you take on some of the work yourself the performance gains can be dramatic.
- "Getting Started With OpenDoc Storage"—OpenDoc storage is a departure from what you're used to: It needs to support storing different kinds of data, written by different part editors, in the same file or container.
- "Sound Secrets"—These less obvious features of the Sound Manager will help improve your application's use of sound.
- "Guidelines for Effective Alerts"—Alerts and modal dialog boxes are common (and commonly misused) elements of the Macintosh interface. This article elaborates and expands on the guidelines for their consistent and correct usage.

please turn to page 24

CD Highlights

Reference Library Edition, December 1995

We've done some additional work on this month's library of technical documentation in Adobe Acrobat format. Keywords have been added to most documents in the core set (*Inside Macintosh*, Macintosh Technical Notes, Developer Notes, and *develop*); for details about searching in general and how to use keywords to limit your search results, see the prototype Contents Catalog in the About This CD folder.

develop issues have been broken out by article, like the *Inside Macintosh* books, to make search results more informative and allow retrieval of individual articles from online services; references to individual articles can also be found in the Subject Index folder. Macintosh Technical Notes have a new champion, Tom Maremaa of DTS; check out his new formatting and organization, and send your feedback to AppleLink address DEVFEEDBACK or Internet address devfeedback@applelink.apple.com.

And, for those who may not be horizontal scroll bar fans, we've included a first attempt at an entire book (the *QuickDraw 3D Reference*) in HTML format. Please have a look, and send us your feedback at dpfeedback@applelink.apple.com.

The Reference Library Edition contains all of this, and much, much more; here are this month's new and revised packages.

Apple Guide 2.0 GM Update

This package contains the run-time software for Apple Guide 2.0, which now supports system software versions 7.0 and 7.1. Also included is the book *Apple Guide Complete* in Acrobat format and the latest version of Guide Maker, the authoring environment for creating Apple Guides.

AppleGlot

AppleGlot 2.3 is a text translation tool used by Apple localizers to localize Apple software. AppleGlot extracts the text from a resource file and dumps it into a text file for translation. Once translated, AppleGlot then replaces the original text in the resource file with the translation. In addition to the regular translation, AppleGlot handles updates from one version to another, and copies over resizable items. AppleGlot 2.4b1, also provided on this CD, adds several additional features.

Gestalt Selectors List 3.1

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

Note: This is *not* an Apple product. It is provided on an "as is" basis. Apple is not responsible for any problems you may encounter in its use.

GetHelpStrings

This sample extracts the Finder Balloon Help strings from any files in any folder dropped onto the application.

How to Detect a CD

This folder contains sample code and describes how to detect whether a volume is a CD or not, discussing the various drawbacks of all the different approaches. It features C sample code demonstrating how

please turn to page 24

OpenDoc Human Interface FAQs

Keyboard Equivalents, Quit, Splash Screens, Buttons, and Embedded Scroll Bars

By Dave Curbow, Elizabeth Dykstra-Erickson, Kerry Ortega, and Geoff Schuller, OpenDoc Human Interface Team

We frequently receive OpenDoc questions that we need to answer in a public forum to help our partners' OpenDoc development efforts. We think you'd like to know about the OpenDoc topics these questions address.

Q: I received the Developer Release 3 CD and now I'm seriously writing some parts. Where do I find guidance on the human interface?
A: There are a couple of places to look for human interface information. First, Part 3 of the *OpenDoc Programmer's Guide* (on the DR3 CD) contains human interface guidelines for OpenDoc. We also post all of our FAQs and other human interface information to the CILabs Web page: http://www.cilabs.org.

Finally, you can subscribe to the two-way OpenDoc-HI list server and send us your questions and comments. To subscribe, send a message to Listproc@cilabs.org and include "subscribe OpenDoc-HI <your name, not e-mail address>" in the body of the message.

Q: What are the guidelines for the use of keyboard equivalents?
A: Keyboard equivalents, or Command-key combinations, execute from the rightmost menu positions to the leftmost. Suppose you assign a new keyboard equivalent, and another command already has that same Command-key assignment. When the user invokes a keyboard equivalent that is assigned to more than one command, the command that will be invoked is the one that appears farthest to the right in the menu structure. The user should not have to think about which command will be invoked; duplicates make the effect of the command unpredictable and confusing to users. Users may also find it very confusing if one keyboard equivalent invokes one command in one part of the document but invokes a different command elsewhere.

We've been careful to establish keyboard equivalents that represent the most effective use of a limited resource. To minimize the potential confusion, we recommend that you use only the reserved keyboard equivalents, defined in the *Human Interface Guidelines* (pages 128–29), plus the OpenDoc-specific reserved keyboard equivalents L (Info) and R (Redo). You should carefully consider the alternatives before adding a new keyboard equivalent to your editor.

Q: Can we consider restoring the Quit command so that users won't be confused by today's model?
A: There are two reasons why OpenDoc does not have a Quit command. First, Quit is an artifact of an application-centered model. Because we restructured the way applications (that is, editors) are started and stopped automatically, this command is no longer necessary.

Second, we know that users often forget to use the Quit command, which can lead to disorientation. That is, when users close the last document in an application, they're left with the menu bar for that application, but they often believe they're in another domain—usually the Finder. This causes a lot of frustration for users.

It's true that issuing a Quit command also means that all open windows of the application are closed. That's sometimes a very useful feature, particularly for experienced users. One option for closing all the windows of the current document is a Close All command—which is somewhat different from Quit, but is consistent with a document-centered model. However, the Document menu is already crowded. So instead of providing a Close All command, OpenDoc 1.0 allows you to perform the same action by using Option-click on the close box or pressing Command-Option-W. (One way that an OpenDoc document might have multiple windows open is if the user were viewing a contained part in its own window, using the "View in Window" menu item.)

We know that some developers are interested in using OpenDoc technology, but aren't using the OpenDoc document model. If you are clearly working with an application model, it is appropriate to retain the Quit command. If you're considering doing this, please feel free to contact us for advice. Watch this space for further information on various content-model–specific guidelines and appropriate variations on OpenDoc guidelines.

Q: When should I display my splash screen?
A: The human interface guidelines covers this topic to some degree. While we can't give you any legal advice, we can tell you what we think is the best human interface for splash screens.

Startup is a natural time to display your part editor's name and other information in a splash screen. However, OpenDoc may start up and shut down your part editor several times during the use of a document. Therefore, displaying your splash screen each time your part editor starts up can become overwhelming to users, especially when several part editors that are used in a single document try to display their splash screens.

We prefer that splash screens appear either at the end of the installation of your editor, or upon first use. After that, the splash screen information should be made available through the "About [editor name]" command in the Apple menu.

Here are some ways to make your splash screen pleasant and unobtrusive:

- Display your splash screen in a modeless dialog box.
- Make your splash screen as small as possible.
- Display your splash screen for as short a time as possible; 5 seconds is appropriate.
- Make your splash screen disappear automatically, without requiring user interaction.
- Avoid the use of buttons on your splash screen.

Q: How do I edit a button or similiar "hot" parts? Can I activate them?
A: In OpenDoc, you cannot activate a hot part. (A hot part—a button part, for example—does not activate when you click it or drop an icon on it. Instead, it takes such user events and performs some action as a result.) You can select it using whatever mechanisms the container provides, such as Shift-click, but you can't activate it. Thus, if the user needs to set properties, we recommend that the user edit the hot part's properties by selecting the part, invoking the Part Info command from the Edit menu, and clicking the Settings button. This will open the properties for this part and allow the user to

edit them. Remember that as the part developer, you are responsible for adding the Settings button to your part's Part Info dialog box.

Q: The sample code I've looked at seems to finesse the whole issue of scroll bars in embedded parts by allowing scroll bars only in the root frame. (I presume there can be only one root frame per part.) Is this cheating, or is it an OpenDoc human interface guideline? When is it appropriate to use scroll bars in the frame?

A: When your part is the root part in a document window, your part editor decides whether to display window scroll bars. In general, documents should have scroll bars when they can contain more information than is visible in the window. However, some documents are WYSIWYG, and some are not. If your part is WYSIWYG, then you should probably not display the scroll bars when it is embedded, since this can cause the layout of the document to change and create visual clutter.

Some parts have a legitimate need to use scroll bars to navigate through content that is not visible in the frame. For example, a part that presents a dynamic list may need a scroll bar. On the other hand, a simple text part probably doesn't need one, since the frame could grow as a user adds more text. When you determine that it is beneficial to implement scroll bars in your embedded parts, observe the following guidelines:

- Whenever the document window in which your part is displayed becomes active, enable scroll bars in all of your parts that have them—even if your part is not currently active. This allows the user immediate access to all such parts.
- When the user clicks in an enabled scroll bar, immediately begin to scroll. In addition, the part containing the scroll bar should become the active part except when the clicked scroll bar belongs to the root part. This exception allows the user to scroll through the document to bring obscured portions of an active embedded part into view without losing the active state.
- Provide a command that allows users to hide the scroll bars when they wish. Alternatively, you could provide an editor preference for this purpose.
- Don't hide the scroll bars when your part becomes inactive. Rather, show the inactive appearance of scroll bars.
- Scrolling keys (such as the page up and page down keys) scroll the root part. If the root part does not scroll, these keys apply to the active part.

See the *Macintosh Human Interface Guidelines* for more complete information on scroll bars. ♣

Copland Drivers—Time to Put the Pedal to the Metal

By Bob Megantz, TacTec

Copland is coming—that's a fact. Copland, Apple's new version of the Mac OS, will provide users and developers with new and better ways to solve their computing problems, ensuring that the Macintosh will continue to be the most productive and easiest-to-use personal computer platform. Apple Computer, Inc., has redesigned the Mac OS to provide better performance, greater flexibility, and increased stability. It has also integrated QuickTime, WorldScript, and other key technologies into the core operating system. For device and board manufacturers, Copland provides a hardware abstraction layer that separates hardware from applications and allows the system to handle new hardware easily.

Apple listened to what its customers wanted in a new operating system, checked out the competition, and then set some pretty lofty goals for Copland. Apple's primary goal for Copland is to make Mac OS computer users more productive than they could be on any other personal computer platform.

Another important goal for Copland is customer satisfaction—not just with the features and performance of the new operating system, but also with its ease of installation. Users must be able to install Copland and get their applications and hardware up and running quickly, without a lot of hassle. For this to occur, developers like you must provide drivers for your products at the same time Apple releases Copland to preserve customers' investments in existing hardware and applications and to drive (no pun intended) new product sales.

Apple engineers designed Copland to provide new opportunities for hardware manufacturers to develop and market compatible proprietary hardware. They did this by making Copland run on multiple platforms, be extensible, and be easier to maintain. To put it simply, Copland will provide a larger market for your products.

Apple knows that it must work together with developers to make Copland a success, and that drivers are an important component of the Copland introduction strategy. For this reason, Apple is putting a lot of effort into working with you now to ensure that all the elements needed are in place when Mac OS computers move to Copland.

Who Should Care About Copland Drivers?

Anyone who writes drivers for hardware devices will be affected by the new driver model. So will providers of core operating system services, such as those of you dealing with network protocols (all networking systems must make use of Open Transport) and file system volumes (Copland uses a new file system).

In addition, application developers who understand the features and power of the Copland driver architecture can develop superior products by taking advantage of those features. For example, under Copland, two different applications can operate SCSI devices simultaneously, while under the current Mac OS one application's SCSI operation must finish before the second application's operation can begin. Users of applications taking advantage of this feature will notice a more responsive environment and improved performance.

The rest of this article takes a closer look at why Copland drivers are different, what the new driver model looks like, and how you can leverage your existing drivers and develop new drivers to ensure a smooth transition for your products and, most important, your customers.

What's Different About Copland Drivers?

As you know, hardware devices need drivers to interface to the operating system. Because Copland is native to the PowerPC processor and utilizes the PCI bus, drivers designed for the current Mac OS will not work with Copland. That's the bad news. The good news is that Copland designers have come up with a driver model that makes it easier for you to write drivers and provides better performance and greater functionality than the current model. You can also design some drivers for current products so that they will work with Copland in the future, and you can structure others to minimize the pain associated with converting them later.

Copland drivers differ from System 7.x drivers in three fundamental ways:

- The Copland kernel has no 680x0 emulator in it, so Copland drivers can contain no 680x0 code.
- Copland drivers are part of an entirely new driver architecture.
- Copland drivers cannot contain Toolbox calls.

Since no emulation is allowed in the Copland kernel, Apple engineers have completely overhauled existing 680x0 driver mechanisms, which make use of 'DRVR' resources and the Device Manager. Instead, Apple is providing a set of tailored subsystems and supporting driver service libraries that provide the foundation for key features of Copland, such as preemptive multitasking (allowing several tasks to run simultaneously and preemptively) and memory protection (assigning protected memory space to each task).

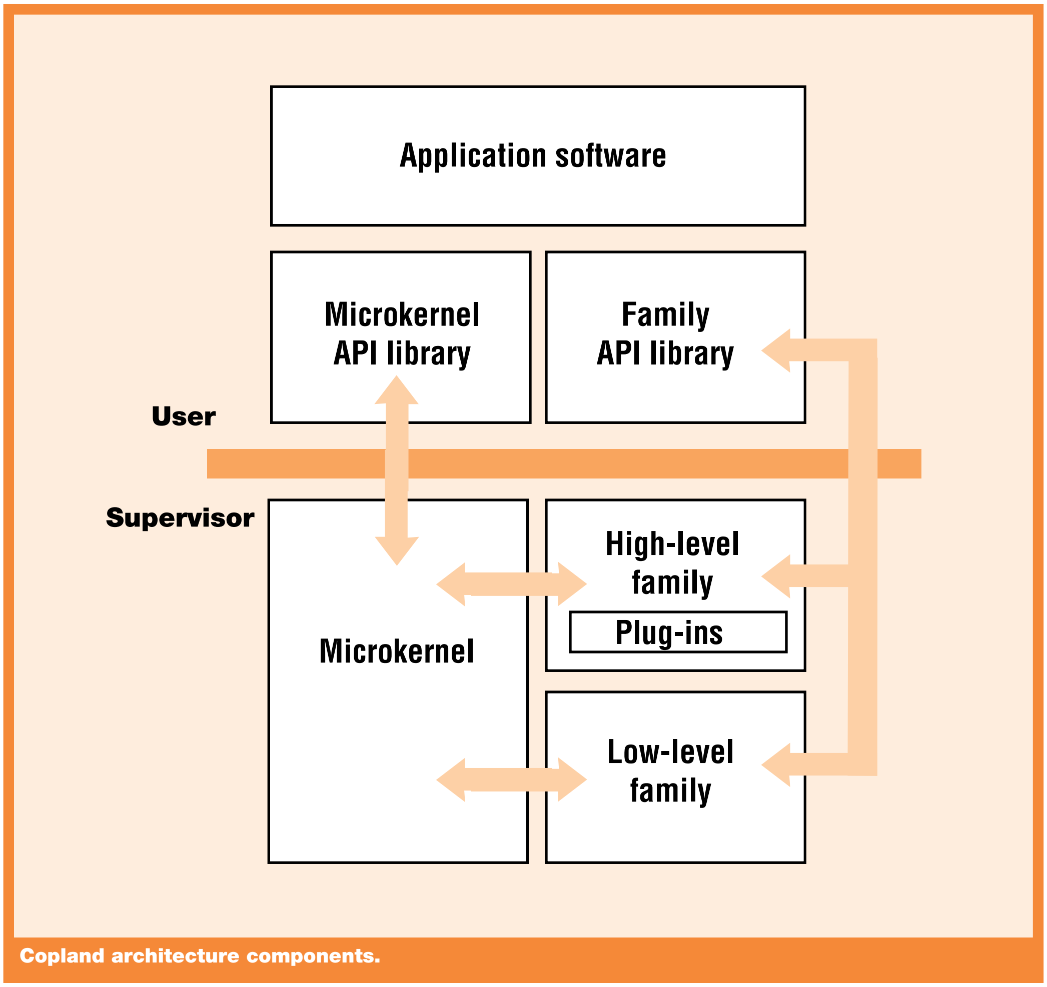
This approach offers some important advantages:

- You'll be happy to hear that, unlike the current Mac OS, Copland offers extensive system support specifically for drivers.
- Interrupt latency, defined as the time required to respond to an interrupt (such as one generated by pressing a key on the keyboard), is low and predictable, so a machine running Copland feels more responsive.
- Device drivers have been optimized, as explained later in this section.
- Best of all, you're going to find that it's much easier to develop drivers and obtain high levels of performance in Copland than in System 7.

Apple designed Copland's I/O architecture with two goals in mind—to provide blazing performance and rich functionality. Speed comes from the kernel being native (executing as PowerPC processor code, with no emulation) and from the fact that Apple has put all of the I/O functionality in the kernel. Copland also explicitly supports I/O functions by defining "families" of devices (described in detail later) and providing rich services for each family, depending on its needs. This approach has several performance-enhancing advantages, including more predictable operation and better interrupt capability.

As you know, the current Mac OS is very flexible, and many developers have worked around the system to get improved performance or special capabilities through patching traps. Under Copland, you won't need (or be able) to patch traps any more, as Copland's driver model does not include patching mechanisms. It does, however, provide performance and functionality that will allow you to supply both the speed and capabilities your customers will need without patching. Copland also supports currently existing and new I/O devices and offers new operating system capabilities, such as multithreaded I/O (which allows I/O processes to be cooperatively scheduled within a single application).

You'll need to be aware of a few things when you write your driver. Drivers exist in the supervisor address space, while the Toolbox and applications live in the user address space. For this reason, drivers can't call Toolbox code, and applications can't call driver code directly. In addition, applications can't com-



Copland architecture components.

municate directly with hardware (all communication must pass through the operating system) and can't contain any supervisor-mode code. Apple has carefully thought out these restrictions to provide the high levels of performance and services required, while making driver development much easier than in the past.

The Copland Driver Model

The Copland I/O architecture has three major components—families, plug-ins, and the activation model. The figure "Copland architecture components" on page 18, shows the place of several of these components in the Copland architecture.

A family is a collection of software pieces that provides a single set of services to the system (for example, the SCSI family and its SCSI interface modules). A family defines an interface for a group of related drivers. Families locate, load, and introduce plug-ins (discussed later) and provide family-specific service libraries. Families are considered high level or low level, depending on what they manage.

High-level families manage some type of I/O or category of service, independent of the hardware—for example, a device or bus. Open Transport is an example of a high-level family. Low-level families manage specific device or bus topologies and advertise their devices to the high-level families. You don't need plug-ins for low-level families, because arbitration is not needed. IDE (Integrated Drive Electronics), PCI (Peripheral Component Interconnect), PC Card (formerly called PCMCIA), and NuBus™ are among the Copland low-level families.

High-Level Families

Each high-level family has two parts that talk to each other, one residing in the application space and addressed by the API (application programming interface), and another that lives in the kernel. This second part operates in the supervisor mode and is always present in the kernel address space. Its responsibilities include plug-in registration, device discovery, driver loading, the service library, and the supervisor library API. Copland currently includes the following high-level families: Device Manager, ADB, pointing, SCSI, PRAM (parameter RAM), real-time clock, file systems, block storage, Open Transport, keyboard, display, and sound.

The family name generally describes the type of device or bus included in the family. For example, disk drives reside in the Block Storage family, all networking devices reside in the Open Transport family, graphics cards reside in the Display family, and so on. For compatibility with System 7, Apple engineers have retained the Device Manager family, even though they don't intend it to be used for new drivers. All Device Manager plug-ins will reside in other families.

You may have noticed a couple of missing families—in particular, a printer family. Printer drivers are different from other drivers in that they reside only in the application space and communicate with Open Transport, so they don't need a separate family. You can implement Copland printer drivers using the QuickDraw GX printing interface model. Existing QuickDraw drivers will work without changes under Copland, but at slower speeds and without the many improved performance features of QuickDraw GX.

Copland also does not provide direct memory access (DMA), coprocessor, or data-acquisition families, because Apple engineers have not found a good way to abstract these devices' services. They are currently evaluating the possibility of driver families for scanners, digital cameras, and other devices. Tape drives will continue to use SCSI 4.3, as Apple engineers have found no good reason to abstract tape drive services. Apple expects there will be other cases where a device or bus will not fit into an existing family; Copland includes the capability to develop new families when needed.

Plug-ins

Plug-ins are the second major component of the Copland I/O architecture. They are the part of the driver that you'll write. Their job is to export entry points defined by the family subsystem, implement the device-specific library, and, in some cases, interface to other family subsystems.

The format of a plug-in is defined by the family interface (not, at least in most cases, by the plug-in itself). The activation model is also provided by the family interface; the plug-in must use the same tasking model. Examples of plug-ins include Display family video drivers, Open Transport family driver and protocol modules, and SCSI family SCSI Interface Modules (SIMs), as well as the Device Manager Native Driver (NDRV).

The actual implementation of the plug-in is dictated by the type of device for which the driver is being developed and by the interconnect method, which also determines in which family or families the driver will reside. (Some devices belong to multiple families; for example, a CD-ROM drive might use services from both the Block-Storage and Sound families.) Once you determine your driver's family, you need to write only the hardware-specific portion of the code; the rest of the driver code is already available. This approach is quite different from existing driver models (where you must write code for the whole driver) and was implemented by Apple to make it easier for you to develop drivers.

Plug-ins use kernel support services, including memory allocation and control, timer functions, interrupt registration and dispatch, libraries of synchronization primitives, and the Name Registry. The Name Registry is a repository

How to Get Started With Copland Drivers

1. Become a member of a Macintosh developer program.
2. Attend a Copland driver briefing.
3. Join the Copland Driver Program.
4. Read the documentation listed in the "Copland Resources" box (page 20).

Once you join the Copland Driver Program and get up to speed, you should begin rewriting your existing drivers. If you are currently writing PCI drivers, you can get some early experience by writing a PCI Video or Open Transport family driver using the DDK; the resulting code will work under both System 7 and Copland.

You should also start planning new plug-ins. Schedule engineering time to write Copland drivers for any of your new products that will require drivers. Apple will provide a mechanism for distributing your drivers free-of-charge with the Copland installation. If you must charge for you driver, let us know and we'll work with you to find solution.

for system information. The high-level and low-level families share data through the Name Registry, which also advertises and locates family services and devices. The function of the Name Registry is similar to that of the Gestalt selector in System 7, but its methods are quite different.

The Activation Model

The third major component of the Copland I/O architecture is the activation model, which determines the tasking model used by the family. The activation model also defines which portions of a service request the plug-in performs (that's why all plug-ins in a family must use the same tasking model), and it dictates the context from which plug-in service requests are invoked.

Activation models can be thought of as only existing at the behest of the family. More specifically, issues addressed by the activation model include plug-in initialization and request processing (what happens at the task level, what can happen at the primary and secondary interrupt levels, and intrafamily communications by means of plug-ins).

To summarize:

- Families define the interfaces for groups of related devices, load and introduce plug-ins, and provide an abstraction layer for applications.
- Plug-ins, the part of the driver that you have to write, respond to their family's activation model.
- Applications respond to their families' APIs, and they don't really care about how the family and plug-in function.

Designing Copland Drivers

When you design a driver, your first job is to understand the run-time environment. Copland drivers operate under the Code Fragment Manager (CFM). They cannot perform many traditional driver functions by accessing Toolbox services. Each driver must utilize family-specific support libraries and activation models; a generic driver service library defines libraries and activation models not provided by another family.

You can load a driver from a device's expansion ROM (on a card, for example) or from a file in the Extension folder (on the hard drive). In addition, you can easily replace a driver if desired; for example, you can substitute a driver located on the hard drive for one on a card.

The Name Registry replaces two things from the current Mac OS: the Preference File, and the plug-in card's ID, which the card stores in nonvolatile memory. Each device has a Name Registry entry that contains configuration data (nonvolatile data stored in the system file or unique to a single system startup). Any code in the system has access to the Name Registry and its data.

Applications interact with Copland drivers using just two symbols, TheDriverDescription and DoDriverIO. TheDriverDescription describes the hardware, names the driver and specifies its configuration, and configures the driver replacement.

You call DoDriverIO for all driver operations; it operates by first issuing a call to prepare for an I/O operation, then calling PrepareMemoryforIO, and finally starting asynchronous I/O. Formal driver execution levels include the task (non-interrupt) level, which is defined by the activation model; priority-based secondary interrupt levels that operate at the convenience of the operating system; and the primary (hardware) interrupt level, which you invoke as seldom as possible to avoid slowing down the system.

Apple's Copland Driver Development Plan

Apple engineers are writing Copland drivers for Apple hardware, and Apple will provide source code from some of these drivers in future Driver Development Kits. This will help you jump-start your development efforts by studying and modifying Apple drivers for your products. The effort required to modify an Apple driver for your intended purposes will depend on the extent of the differences between your device and the devices in the same family for which Apple is writing drivers.

Schedule

Apple plans five phases in the Copland driver development effort:

- Phase 0: By the end of this year, Apple will provide driver design documentation in draft form both internally and to outside developers.
- Phase 1: Apple-designed Power Macintosh 6100, 7100, and 8100 prototype drivers will be available at the beginning of 1996, along with updated documentation.
- Phase 2: Apple will deliver Power Macintosh 6100, 7100, and 8100 final drivers and prototype drivers for other existing hardware.
- Phase 3: Apple will deliver final drivers for other existing hardware.
- Phase 4: Developers will deliver third-party drivers.

Apple will also provide you with a preliminary Device Driver Kit (DDK) by early next year. The DDK will include sample drivers and, for the first time, complete production drivers developed by the company for its products. In addition, Apple will include the test and qualification tools that are used internally so that you will have as complete a suite of test tools as possible to use with your drivers. Apple engineers plan to work with you to develop additional testing tools for you.

What You Can Do Now

There are several things you can do now to help ease the transition to Copland. Some involve relatively minor changes to the way in which your current drivers are being written, while others will involve more effort. Remember, though, that you and Apple need to work together to provide your customers with the system components they need to quickly and easily make the move from System 7 to Copland.

Current Driver Development Efforts

First, if you haven't done so already, make sure that your

Copland Resources

- *Designing PCI Cards & Drivers for the Power Macintosh**
- "PCI Device Driver Development," in *develop,* Issue 22*
- "PrepareMemoryforIO," in *develop,* Issue 24
- The PCI Driver Development Kit 2.0 contains documentation, tools, and sample code (including the documentation marked above with an asterisk). You can get these materials from the September 1995 Developer CD, using the path Dev.CD Sep 95 RL:What's New?:New System Software Extensions:PCI Driver Development Kit.
- Apple Developer Services Web site: http://www.info.apple.com/dev/
- *develop* magazine: http://www.info.apple.com/dev/develop.html
- Mac OS Web site: http://www.austin.apple.com/macos/
- Apple Web site: http://www.apple.com/
- "Why Mac?" Web site: http://www.apple.com/whymac/

current driver development work maximizes compatibility with Copland.

- If you're currently writing PCI drivers, you may be able to design them to work with little or no extra effort under Copland. You can find details on how to do this in *Designing PCI Cards and Drivers for Power Macintosh Computers* (see the "Copland Resources" box on page 20 for more details).
- Use the Enhanced Device Manager (a version of the Device Manager, modified to work with System 7.5.2, that allows the use of native drivers in PCI systems) or another family expert—Copland will support them. (A family expert is code that works with all the Name Registry entries associated with a given family. You can read about family experts in the *Designing PCI Cards and Drivers for Power Macintosh Computers* document.)
- Remove Toolbox dependencies. Any Toolbox calls will cause the driver to break under Copland.
- Isolate operating system dependencies. You can use code that processes events or handles the physical device in your Copland driver (although it will have to be translated from 680x0 language if it's not written in a high-level language), but you'll have to rewrite any code dealing with the way the device driver relates to the operating system.
- Learn to use the Code Fragment Manager, so you can produce software that the system knows how to load.
- Keep in touch with Apple. The Copland team members are interested in your feedback and can provide you with the latest information (see "The Copland Driver Program" below).

Plan for Copland Drivers

You should also start thinking now about how to manage the transition from System 7 to Copland. Apple is putting a lot of new resources into developer support, including dedicated teams to help you write the drivers you will need. Start by attending a Copland driver briefing; Apple is holding them all over the world. If you haven't already done so, join one of the Macintosh developer programs, so you'll get the latest developer information as soon as it becomes available.

Get involved now in the Copland Driver Program (described in the box below), and get on the Copland database so Apple can get your input and feedback on issues of mutual interest. Also, if you have questions or concerns about how to make the transition to Copland drivers, you can send e-mail to AppleLink address NEW.IO or Internet address new.io@applelink.apple.com.

Working together with Apple will benefit you, Apple, and most importantly, your customers. Joining the Copland driver program will help ensure that your customers will experience a relatively painless transition from System 7 to Copland, which will in turn reduce the support that you'll need to provide them to make that transition. Using the Apple drivers that will be supplied to program members will also reduce the workload on your programmers.

Copland *Is* The Future Mac OS

The introduction of Copland will provide major improvements in operating system performance for you and your customers. However, to achieve that performance, you've got to develop Copland drivers for your products, so that your customers can experience a satisfying transition to Copland. Apple has provided you with a driver architecture that makes it easy to write Copland drivers, but you've got to do it.

Remember, now's the time to start thinking about Copland drivers. By working and planning together, we can put all the necessary elements in place to ensure a smooth and successful Copland introduction. By doing so, your products will remain state-of-the-art, and Apple and its developers will continue to provide Macintosh users with the best and most effective personal computing platform available. ♣

Bob Megantz is a principal in TacTec, a technology licensing and business development consulting firm in San Jose, California.

The Copland Driver Program

Joining the Copland Driver Program is the best way to ensure that your needs and your customers' needs will be met in the fastest and most efficient manner. To get an application for the Copland Driver Program, send a request to AppleLink address NEW.IO or Internet address new.io@applelink.apple.com.

Copland Driver Program participants can take advantage of a number of developer support services, including an upcoming Developer University Driver Class scheduled for the end of this year. Apple will provide the Copland DDK early next year to program participants, and is planning special family/theme driver "kitchens," where developers are invited to spend a few days getting hands-on experience with developing drivers, with help from Apple software engineers.

Documentation is available from Apple, in both online and printed form; you'll find lists of available documents and their locations in the box "Copland Resources" (page 20). Driver briefings have been held at Macworld in Boston; MacFest in Seoul, London, Tokyo, Taipei, and Hong Kong; and at the Seybold Conference in San Francisco. The dates and locations of future briefings will be found on Apple's World Wide Web sites, which are also listed in the "Copland Resources" box.

In addition to the engineering resources Apple has dedicated to the Copland program, the company has retained an outside public-relations agency to help publicize your developments. Apple is assembling dedicated teams of driver evangelists and developer technical support personnel to support program members; making the transition to Copland is not only the largest project Apple has ever undertaken, it is the company's number-one priority, and you can expect to receive a lot of help.

Human Interface

Culture Clash

By Peter Bickford

A few years ago, I was teaching a class in Tokyo. At the time, the United States and Japan were having one of their regularly scheduled trade scuffles, this time over cars. As you might have noticed, Japanese cars are very popular in the United States, whereas U.S. automobiles are a tiny fraction of the Japanese market. U.S. auto makers traditionally explain this by saying that Japan is using trade rules to close their car market to imports. Japan, meanwhile, usually responds that the reason U.S. cars don't sell in Japan is that U.S. auto makers don't make the sort of cars Japanese buyers want.

Whatever the merits of the U.S. position, the two American cars I did see in Tokyo seemed to bear out the Japanese claims. They were stock American designs with the steering wheels located on the left side. However, in Japan, cars don't drive on the same side of the road as in the United States. If you were to look at the thousands of Toyotas, Hondas, and Nissans parked along Tokyo streets, you couldn't help notice that all the steering wheels were on the right.

Japanese companies exporting their cars into the United States wouldn't dream of selling a right-steer car into a left-steer market for fear of being laughed out of town. But until fairly recently, U.S. auto makers were arrogantly attempting to sell left-steer cars into a right-steer market. Rather than invest the time and expense to change the design, they decided that the Japanese could just get used to driving the American way. Not surprisingly, the Japanese customers declined to play along.

The history of marketing is loaded with accounts of similar disasters. The Coca Cola company, for instance, used their popular slogan "Coke adds life" in China, where it somehow translated into a promise to elderly Chinese that it would bring their ancestors back from the grave. A similar debacle occurred when Chevrolet marketed their Chevy Nova in Mexico. *No va* means "doesn't go" in Spanish. In all these cases, the problems occurred because the people selling the products either didn't know or didn't care about the cultures they were selling into.

Cross-Platform/Cross-Culture

Of course, the cross-culture problem is not limited to soft drinks and automobiles. Software developers face similar problems when they move from one platform to another. Even when the new users speak the same language, there's a good chance that the new platform's "culture" is very different—and if you don't pay attention to those differences, you're courting disaster.

The fact is, users want applications that feel like they were designed solely with them and their platform in mind. For whatever reasons, they've chosen the computer they like the best, and they expect you to write software that respects that preference. They don't want least-common-denominator interfaces or shoddy ports from some foreign platform. Worse, they have no sympathy whatsoever for how bloody difficult it is to develop cross-platform software. The vast majority will only use your program on one type of computer, so they couldn't care less about common code bases, or the abstract advantages of having identical interfaces across all different platforms. They're comparing your software with the best of its type available for their platform—not the best *cross-platform* software that happens to run on their platform.

If you want to win the cross-platform development game, it's not enough merely to translate your old code to run on a different processor and display the right sort of windows. You have to convince your new users that you understand the values that led them to choose their platform in the first place. For instance, a Macintosh user who values simplicity and elegance is not likely to warm to an interface originally designed for a 3270 terminal. No matter how functional the interface might be, it will seem strange and foreign to a Macintosh user.

Similarly, a giant in DOS word processing failed horribly the first time it tried to bring its flagship product over to the Macintosh. One of the factors in this failure was their insistence on offering a "code view" mode that showed all the various formatting codes (for example, "[Bold][Italic]Hello World![End Bold][End Italic]") in line with the text, a decidedly useful feature on character-based DOS screens. To Macintosh users, who had bitmapped screens and WYSIWYG displays from the start, including such a feature just indicated that the company didn't really know what the Macintosh was all about.

Understanding the Other Side

It used to be that when some lout in a bar wanted to pick a fight with you, he'd make a crude remark about your date. After a few perfunctory rounds of "Sez who?!" "Sez me!" fists would fly and glass would be shattered. In Silicon Valley, you can often get the same reaction with the old "Mac OS vs. Windows" gambit, or if that fails, "Pascal vs. C." Obviously, I've got my own opinions on these issues, but for the sake of cross-platform development, let's just leave it that people need to use the tools that work best for them, and skip the holy wars.

There's a time and a place for touting our own platform's advantages, but when we're doing cross-platform development, our goal should be to understand—not convert—the users of the new platform we're developing for. Imagine that the new platform is like a foreign land, and your goal is to be able to do business there smoothly. When some less-enlightened individuals go to another country, they demand (loudly, usually) that everything be done just like it was back home. Brighter people do their homework beforehand, read about the culture, and study its language and customs. Once they go there, they immerse themselves in the culture, asking questions and keeping their eyes open.

In the same way, you should attempt to immerse yourself in the culture of the computer platform you're developing for. Get a computer of the sort you'll be developing for and use it as your primary machine for a month or so before you do serious development. Join a user group and spend time talking to regular users of that platform. Study its system software and human interface guidelines. Find out what the platform's traditions and idiosyncrasies are. (For example, does the platform use different words or phrases for common actions? A good example of this is that Windows has an Exit menu command, while Mac OS computers use Quit.) Most of all, you should try to figure out the different values that made users of the platform choose it in the first place.

Several aspects differentiate the PC world from the Macintosh world. Here are a few from my own experience:

- *Configurability vs. simplicity.* PC users tend to value configurability and information on the low-level workings of the system. Traditionally, this knowledge is seen as part of being a "power user," and users feel constrained if they can't get access to these facets of the system. Most Macintosh users, on the other hand, place a higher value on simplicity and design elegance. They see fiddling with interrupt vectors, device drivers, and so on as a distraction from being able to get work done.
- *Interface standards.* Macintosh users are annoyed when a program gratuitously violates human interface standards that they've grown accustomed to. Not only does it make them feel unsure of how the program functions, but it makes the application seem less stable and reliable in their eyes. Conversely, the PC culture gives much more play to individual developers, letting them experiment with different interface styles and elements.
- *User control.* A good Macintosh program always makes users feel like they are in control of the situation; the program is basically waiting to respond to their actions. In many parts of the PC world, it's considered acceptable, even desirable, to lead users by the hand through a series of tasks. Lately, much emphasis has been on "wizards" that take the user, step by step, through a complex process with little ability to deviate from the preplanned sequence of events.
- *Aesthetics.* Each platform has its own set of design aesthetics. The Macintosh's emerging 3D looks, for instance, strike me as more lighthearted, less "heavy" than Windows. Windows-based applications tend to use dark grays, lots of chiseled surfaces, and relatively little white. Both platforms almost have their own color scheme—for instance, with the same shades of blue, green, and gray popping up time and again in Windows-based applications. The strange part is that over time we internalize these aesthetics and use them everywhere because they look good to us. It's not a bad idea to have a "native" designer develop looks for the interface on your new platform lest it be spotted as a port from 50 paces away.

Obviously, these observations don't hold true for every user or every piece of software. Still, each platform has its own basic culture and values—the more you can discover about them, the more successful your software will be at fitting in there.

Avoid the Red Flags

If you're from the United States and you go to England, you'll soon learn that despite the countries' allegedly common language, there are things you can say that will immediately peg you as a Yank: ordering a hamburger and fries instead of a beefburger with chips, or calling the sport *soccer* instead of *football.* And if you really want to make the lads down at the pub cringe, pronounce the final letter of the alphabet *zee* (they say *zed*).

Similarly, once you're lived in the world of the platform you're developing for, you'll discover that certain innocent-seeming interface quirks are like waving a red flag in front of a bull. Recently, I saw a Windows port to the Macintosh that put a little button marked "close" on the right-hand side of each window's title bar, adapting a feature of Windows 95. Weirder still, a company outside the United States recently developed an image-editing program for the Macintosh that slavishly copied the underlined-letter Command-key scheme from Windows, even though the program was being developed solely for the Macintosh. It was exactly this sort of red flag that earned the program the dread title of "un-Maclike" in the press. In three different reviews, the program earned an average of about one star out of five.

If your basic interface is sufficiently well constructed, moving it to another platform need not be so much a matter of revising it entirely as it is of simply avoiding such red flags. Before shipping, run your interface past several of the new platform's native users and ask them to point out any features that don't fit in. If they find any such problems, try to either develop a solution that works on all your target platforms, or add special-case code to handle each platform appropriately. Sure, it's extra work, but avoiding such blatant interface aberrations pays big dividends in review points and product sales. Don't let the desire for a common code base or an easy testing schedule leave you selling left-steering cars in Japan. By paying attention to your new platform's culture and taking the time to get the details right, it's possible to build a product that not only fits into the new market, but that your new customers can fall in love with.

Till next time,
Doc

Peter Bickford is a member of AppleSoft's Human Interface Design Center. You can reach him by AppleLink at THE.DOKTOR.

develop *Issue 24*

continued from page 15

- "Printing Images Faster With Data Compression"—PostScript™ Level 2 printers can accept JPEG-compressed image data directly, which can greatly improve printing speed. Here's what you need to know to take advantage of this ability.
- "The New Device Drivers: Memory Matters"—Using PrepareMemoryForIO to set up memory for data transfers to or from other devices is a complex but very important process. This walk-through points out traps and pitfalls along the way.

We've also got columns on making applications scriptable and getting the most out of QuickDraw 3D, as well as on PowerPC performance profiling and ToolServer (for running MPW commands in a stand-alone mode). KON and BAL return to write a Puzzle Page of their own, while Dave Johnson writes what may be his last Veteran Neophyte column for a while.

To enjoy Issue 24 in all its variety, take a look at it on this month's Developer CD, or in print if you've subscribed to *develop* through APDA. You'll also find *develop* articles on the World Wide Web at http://www.info.apple .com/dev/develop.html. And remember, we're always eager to get your feedback—and your article submissions—at AppleLink address DEVELOP or Internet address develop@applelink .apple.com.

Caroline Rose
Editor, develop

CD Highlights

continued from page 15

to use the Device Manager, SCSI, and calls to the CD-ROM driver to give hints about whether a volume is a CD-ROM or not.

iso9660

This folder contains example code for building simple ISO 9660 volumes. It demonstrates reading and writing to a device using ISO 9660 format.

Macintosh Technical Notes

Technical notes are a collection of short (and not-so-short) articles dealing with specific development topics.

This month features 12 new and revised technical notes in Acrobat format on a wide range of topics, from "Understanding PCI Bus Performance" to "Understanding Type 11 and No FPU Installed Errors."

Also included are author guidelines for Apple engineers or developers worldwide who want to write or contribute a tech note.The guidelines explain the new structure and approach to tech notes. Please send any feedback on the new structure to AppleLink DEVFEEDBACK or devfeedback@applelink.apple.com.

PCI Driver Development Kit 2.0

This folder contains information on designing PCI cards and drivers for the PCI-based Power Macintosh models, such as the Power Macintosh 7200, 7500, 8500, and 9500 computers.

This is version 2.0 of the Driver Development Kit (DDK)—an update to the PCI DDK that appeared on the September 1995 CD. The PCI DDK 2.0 contains new headers and libraries, a new tokenizer, updated samples, and updated documentation (notably an update document to *Designing PCI Cards and Drivers,* and a newer *PCI Binding* document).

PeopleSoft/Apple White Paper

Apple and PeopleSoft have launched a partnership that will deliver PeopleSoft applications that run on the Mac OS. This document is a technical white paper that describes the partnership, the product, and the direction in which PeopleSoft is heading.

Performance Tools

This folder contains several tools for testing the performance of code on PowerPC processors. 4PM is a tool that uses special registers in the PowerPC 604 processor to gather performance statistics; the Emulator control panel allows you to turn off the dynamic recompilation feature of the 680x0 emulator; and the POWER Emulator control panel allows you to turn off the emulation of IBM POWER instructions, letting you find them in your code.

QuickDraw 3D Reference (HTML)

This is a prototype of the *3D Graphics Programming With QuickDraw 3D* book in HTML format. We're including this on the Developer CD to generate some feedback on design, usefulness, and so forth, in a feasibility study of HTML as an online delivery mechanism for technical documentation. Please send your feedback regarding this HTML book to dpfeedback@applelink .apple.com.

SDK Documentation

This folder contains documents (and references to documents) from some of the various SDKs in the New System Software Extensions folder on the Tool Chest CD. They are collected here for convenience, and to make the Acrobat search index more complete.

WASTE

WASTE, the WorldScript-aware styled text engine, is a replacement for TextEdit. WASTE is a text engine that supports WorldScript and text larger than 32K. It has built-in undo support, and it also supports dragging and dropping of text, pictures, and other objects. It can also support tabs. WASTE comes in both C and Pascal versions.

Note: This is *not* an Apple product. It is provided on an "as is" basis. Apple is not responsible for any problems you may encounter in its use.

Coming Next Month

At press time, there is still no definite word about a new system software update; but if one does appear in the next month, you'll find it here. . . .

Alex Dosher
Developer CD Leader

Business

Special Market Report

Developer Opportunities in the Health Care Market

By Hadley Brandt and
Ken Seymens, Apple Computer

The $1 trillion U.S. health care industry is, to put it mildly, feeling a bit under the weather these days. With the federal government reducing Medicare funding, medical institutions are suffering from the same malaise that the finance and defense industries experienced ten years ago—the growing pains of learning to run an efficient business.

Self examination hasn't been easy for this industry. How could an organization buy a million-dollar CAT scanner, yet live with a patient record tracking system that was state-of-the-art during the leech-bleeding era? Costs must be correlated with patient outcomes and operating costs must be reduced. And, as we all know, such a change requires high-dollar computer solutions.

For developers, systems integrators, and contract programmers interested in the health care market, this is a time of opportunity. Medical establishments are in the process of updating their predominantly proprietary, host-based computer systems. Some will try to patch together existing systems; others will throw out the old and start anew. But in both cases, attractively priced Mac OS– and Newton-based solutions have the potential to help these institutions reduce costs and streamline operations. In this article, we'll discuss this market's unique barriers to entry, and the most promising strategies for selling your Apple technology solutions into this market.

A Brief Medical History

To make an intelligent diagnosis of the opportunity in the health care information services market, it helps to take a brief look at this industry's history. Up until the mid-1980s, hospitals and clinics had little incentive to improve efficiency. Medicare—the U.S. government's health insurance program—was the industry's cash cow, bringing in 30 to 40 percent of health care revenues. This program's "cost-plus-profit-margin" method of reimbursement hid a myriad of business sins, such as excessive medical testing and expensive equipment purchases.

Enter the government economists, who realized that the Medicare program would run out of money if the system continued on its present course. So in 1984, the program moved to a reimbursement system where physicians were provided with fixed funds for specific medical procedures.

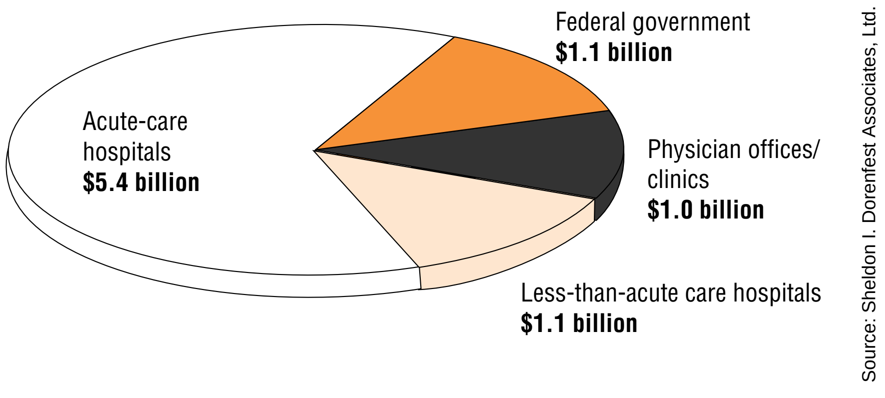
All of a sudden, previously profitable Medicare patients became a liability. Medical institutions found they had to shift Medicare overruns to private payers—employers and insurance companies. And this, along with rising equipment expenses, sent medical costs skyrocketing upward.

Computers were first adopted by medical institutions as a way to cut the costs of processing

Inside This Section

U.S. Health Care Information Systems Market, 1994

Total Market: $8.5 Billion



Source: Sheldon I. Dorenfest Associates, Ltd.

claims. These initial systems were primarily proprietary, host-based systems running large, hard-to-maintain custom software. Outside the billing departments, computers were often purchased by individual physician groups in an uncoordinated manner.

“The heath care industry is easily five to ten years behind the financial and insurance industries in the adoption of computers,” says Sally Yanchus, an equity research analyst at the New York City investment banking firm of Punk, Ziegel & Knoell. “Though it’s going to take the industry a while to catch up, the market opportunity is sufficiently great enough to provide many hardware and software companies with new business opportunities.”

Anatomy of the Market

Today’s health care information services (HCIS) market is enormous, as hospitals, clinics, physician groups, and managed care insurers (commonly called h*ealth maintenance organizations* or *HMOs*) strive to manage information and reduce costs through computerization.

As you can see in the pie chart on page 24, acute-care hospitals constitute the largest segment of the market—though it’s also probably the most difficult segment for Mac OS developers to crack. Hospital administrators, on the whole, are a conservative bunch; they’re more likely to buy “Big Blue” than Apple. Typically they buy solutions that have been installed in similar institutions, and with so many strong-willed constituents to please (that is, physicians and insurers), they’re rarely able to buy off-the-shelf software solutions.

Many large hospitals and large clinics own old host-centric computer technology that must be worked around or thrown out. IBM 3270 systems are commonplace, but on the hospital administration side, no system vendor has a dominant market position. These hosts or servers are often loaded with custom software that is expensive to maintain and expand. On the desktop/client side of health care institutions, it’s not unusual to see ASCII terminals on desks, and the market share of personal computers roughly mimics that of general business: 10.6 percent of acute-care hospitals use Macintosh computers, according to a 1994 survey of 2,097 acute-care facilities (source: Sheldon I. Dorenfest Associates, Ltd.).

If you’re new to the health care market, one way to sell your product into large hospitals and clinics is to team up with one or more of the top HCIS solutions vendors. In these partnerships, HCIS integrators benefit from your Mac OS expertise, and you benefit from referrals that you can use as springboards to other contracts. Another approach that’s worked for many Mac OS developers is to sell directly to hospital administrators, many of whom are looking to consolidate “best of breed” applications into their hospital information systems. Using tools such as Neuron Data or Omnis 7, we’ve seen some of these best of breed systems integrated into elegant, comprehensive clinical care systems.

Fighting the Paper Tiger

One of the best indicators of hospital and clinic needs is the Healthcare Information and Management Systems Society (HIMSS)/H-P Leadership Survey. As you can see from the survey’s results (in the box “Multimedia Use in Health Care”), the very computer technologies that medical information system (IS) people are looking to buy this year—multimedia-based solutions, educational software, and image-intensive remote diagnostics tools—are technologies in which the Macintosh excels.

According to this survey, the number-one priority of HCIS purchasers is to network together the fragmented computer fiefdoms within their institutions. This means that “glueware”—the network software, e-mail, and routers that help a system of multiplatform computers work together—will be in high demand in the coming year.

A second priority is to move from a paper-based record system to a computer-based patient record system, a goal that in many cases may take years to accomplish. One obvious advantage to electronic records is convenience and efficiency—treating and billing patients will no longer be slowed down by lost or unfiled records. The other notable advantage is that it will enable the health care industry to use this information to improve health care and reduce costs: It’s much easier and less expensive to access electronic-based patient data for use in outcome databases or cost accounting.

“Today hospitals, physicians, and insurers are starved for information, and the health care companies that can most quickly harness this information are going to be the survivors,” says Mark Hyde, president of Lifeguard, a managed care insurer based in Milpitas, California.

“There’s a gigantic opportunity for software companies that can create these information management tools.”

As a way to control costs, many managed care companies have been driving the creation of computer tools that correlate patient treatments with outcomes. Lifeguard, for example, uses a computer-aided physician precertification program to detect variations in medical practices and to screen requests for medical treatments. When variations are spotted, such as when Lifeguard discovered that four times as many hysterectomies were performed in California’s Central Valley than in Santa Clara Valley, the insurer can then inform member physicians, so they can reevaluate the effectiveness of their prescribed treatments.

From a clinical perspective, health care will be further improved when the next level of software intelligence is built into patient record handling systems. In the future, when health care providers pull up records, specialized software will alert them to patient allergies, conflicts in medications, and redundant tests. Physicians will be able to more quickly tap into vast repositories of medical information and treatment success rates. And with technologies such as OpenDoc,

Multimedia Use in Health Care

In the next year, health care information managers plan to purchase:

Multimedia-based solutions	**80%**
Health care education solutions	**42%**
Remote diagnostics (telemedicine)	**32%**

Source: 1995 HIMSS/H-P Leadership Survey the Apple-supported modular software architecture, institutions will be able to more easily drop this type of functionality into their complex information systems.

Newton Technology in Health Care

In order for electronic records to truly become a reality, caregivers need the ability to access and add information to a patient's record during each patient encounter. One challenge in achieving an all-electronic system is the amount of time that mobile health care professionals spend walking around, away from desktop computers. Because of this unavoidable situation, Apple's MessagePad and other Newton-based personal digital assistants are being used extensively to collect and access clinical and patient data.

The Apple MessagePad is the leader in the personal digital assistant (PDA) market, holding a 58 percent market share in 1994, according to BIS Strategic Decisions. And since 1994, the number of health care applications designed to run on the Apple MessagePad has more than doubled (source: Apple Computer, Inc.). Applications have been developed in areas such as home health care, managed health care, anesthesiology, mental health, dentistry, emergency medicine, respiratory therapy, nephrology, hand therapy, and occupational medicine. Types of applications include computerized medical records, encounter forms, prescription writing, critical pathways, basic life support, drug reference, and diet management. These applications are currently used in a variety of inpatient and outpatient settings. Price/performance, size, battery life, and the MessagePad's ergonomic design have all been influential in the health care profession's decision to use these PDAs.

In recent months, many major health care institutions have made major investments in Newton-compatible devices. For instance, Johns Hopkins Hospital recently completed a ten-month pilot project that uses Newton devices to improve physician and case manager documentation processing. Like many other hospitals, they chose this product because of its lab pocket size and the fact that it easily interfaces with either Mac OS or PC-compatible computers.

Yet another developer, Greenleaf Medical Systems in Palo Alto, California, has teamed up with the American Society of Hand Therapists (ASHT) and a private therapy clinic to create a Newton-based system that helps physicians and therapists treat ailments such as carpal tunnel syndrome and assess important outcomes information over time. Using Greenleaf Medical's ORCA (Outcomes Reporting and Clinical Analysis) software, hand specialists can quickly document patient conditions, evaluate patient information, and access the UE Net, ASHT's national "upper extremity" data base. This system has been installed in Walter Reed Army Hospital in Washington, D.C., and Loyola University Hospital in Chicago.

Home health care providers have also rallied around the Newton platform. This year a number of home health care applications for visiting nurses have been introduced, including ClinPen, developed by Green Mountain Software Corporation in Colchester, Vermont; VISTA, developed by StaMar in San Clemente, California; and Pegasus, developed by Point-of-Care Systems in Toronto, Canada. And, of the twelve mobile home health care vendors demonstrating solutions at the 1995 National Association for Home Care meeting, half used MessagePad PDAs.

"As hospitals move to reduce costs, home heath care will become increasingly important," says investment banker Yanchus. "We're already seeing growth in the use of hand-held, pen-based devices. These products enable nurses to visit patient homes, record patient health data, then wirelessly send data to a central database."

Newton developers can also take advantage of Apple's aggressive program aimed at helping systems integrators (SIs) and value-added resellers (VARs) create Newton-based health care solutions (summarized in the "Health Care Market Resources" box below).

Macintosh-Friendly Teaching Hospitals

Perhaps the easiest way for Mac OS developers to break into the institutional side of the market is through university-based hospitals. The Macintosh holds a 30 percent market share in higher education and, at last count, 126 medical schools have significant installations of Macintosh computers (source: Apple Computer, Inc.). By selling into these Macintosh-friendly medical schools, you'll be winning the mind share of thousands of new health care professionals who graduate from these teaching hospitals each year.

Besides using Macintosh computers for research papers, data acquisition, analysis, and simulations, medical school instructors are continually looking for interactive content that more effectively trains health care providers. Some of the most successful Mac OS–compatible medical education solutions have come from

Health Care Market Resources

Here are some Apple resources that may come in handy if you're marketing health care applications or services.

• *The Medical Solutions CD.* This sales tool provides Apple's field sales force with demonstration copies of third-party health care solutions. To inquire about placing your demo on this CD-ROM, contact Ken Seymens (mr. ken@applelink.apple.com) or Kris Brittain (brittain@applelink.apple.com).

• *Tradeshow participation.* Apple Computer participates in several heath care industry tradeshows, such as HIMSS (March 1996), the Symposium on Computer Applications in Medical Care (SCAMC) next October, and the American Radiology Society of North America (ARSNA) symposium. Apple often invites third-party software vendors to demo products in its booth.

• *Newton SI/VAR program.* This program provides Newton systems integrators (SIs) and value-added resellers (VARs) with customer leads, timely information from Apple, and technical support materials. The best way to stay informed of these and other opportunities is put your name on the Newton Medical List Server by sending an e-mail message to listproc@dsg.harvard.edu and including "subscribe newtmed <your name>" in the body of the message. Or check Apple's Newton World Wide Web page at the location http://med-amsa.bu.edu/newton.medical/newton.medical.html.

You can also post demos on eWorld at the location The World of Newton:Software Exchange:HealthCare.

Gold Standard. Today their product line, which covers topics from radiological anatomy to clinical pharmacology, can be found in almost every U.S. medical school. Another company that has found success in both the medical school and consumer software market is A.D.A.M. Software, Inc.. (For more information, see the box below.)

Patient Education

Today physicians spend an average of 6 to 10 minutes with each patient during office visits (source: Health Wave in Wilmington, North Carolina). This is clearly not enough time to provide in-depth education to patients, and those who want to learn more will have to turn to other resources, such as interactive software. In addition, health care insurers are finding that the lifetime costs of chronic diseases, such as diabetes and asthma, can be greatly decreased with better patient education. These trends, along with the prevalence of home computers, have industry experts optimistic about the outlook for health care education titles.

Software developers may also want to look into the opportunity of developing custom educational titles for pharmaceutical companies as a way to educate both patients and physicians on new drugs and treatments. (You can read about Bergen Brunswig's AccuSource™ CD ROM–based pharmacy catalog in the September 1995 *Apple Directions* article "Apple Guide—Intelligent Help for Your Business.") Multimedia programmers also may want to look into developing interactive tools that educate employees on competing medical plans and deliver enrollment information to insurers by e-mail.

Imaging the Future

Apple's strength in multimedia has helped it garner a tremendous following in the image-intensive specialties of surgery, radiology, pathology, and nuclear medicine. Today, many ultrasound, CAT, and magnetic resonance imaging (MRI) solutions use cost-effective, customizable Macintosh computers as an alternative to expensive medical equipment controllers. One developer who's made inroads into this market is Evergreen Technologies, Inc.: Their MedVision Viewer and MedVision Modality Pac enable a Macintosh to serve the same function as the $1 million film-viewing consoles that many hospitals use today. Also, Acuson has developed a Macintosh-based system that enables ultrasound data to be delivered over phone lines to specialists.

Remote medical care, commonly called *telemedicine,* is just beginning to take off. Surprisingly enough, one of the first institutions to adopt this technology has been the U.S. prison system, primarily because of state laws that require these institutions to provide inmates with quality health care. Another early adopter has been the U.S. military. Walter Reed Army Hospital in Washington, D.C., for example, uses Power Macintosh computers with built-in videoconferencing to provide medical care to remote UN hospitals in Somalia and Croatia.

Dr. Edward Gomez, the chief of vascular surgery and a telemedicine advocate at Walter Reed Hospital, talks about technologies that they'd like to see in the near future: "For telemedicine, we need larger, higher-quality images and quicker ways to transmit images. We also need better imaging input devices that are as easy to use as QuickTake cameras."

Apple's 3D imaging technology, QuickDraw 3D and QuickTime VR, also has the potential to revolutionize medicine. One developer working to create commercial medical applications of these technologies is Greenleaf Medical Systems of Palo Alto, California.

A.D.A.M. Software—Leveraging Their Body of Knowledge

One Mac OS developer who has met with success on both the institutional and consumer sides of the health care market is A.D.A.M. (Animated Dissection of Anatomy for Medicine) Software, Inc., based in Atlanta, Georgia. A.D.A.M. started out creating medical illustrations for textbooks and courtroom exhibits. In the early 1990s, the founders decided to leverage their medical illustration talent by developing a software product that would enable medical students and physicians to explore the human body using a computer. So, they gave their illustrators Apple computers, and the first A.D.A.M. software product was born.

In order to finance their new venture, the A.D.A.M. team started out slowly, first illustrating the foot and lower leg, then working their way up to the head. Using the SuperCard multimedia authoring tool, they illustrated body structures in layers, creating an animation that simulated the experience of dissecting a human body. At the same time, they added complexity to their product, creating a "pixel-level recognition" technology that enables users to click an anatomical structure and display more detailed information. Today, A.D.A.M. has illustrated 22,000 body parts and has created 17 different software products for education, professional, and consumer health care markets.

Dissecting A.D.A.M.'s product and marketing strategies, you can learn a lot about succeeding in this market. First, they focused on their core competency—medical illustration—and applied this expertise to create a medical teaching aid that provided an interactive experience that students couldn't get from a textbook. Second, they developed a proprietary technology—a dissection "engine"—which provided them with a competitive advantage in the marketplace. Finally, they leveraged their technology investment by creating specialized modules that addressed specific health care markets: medical professors and students, undergraduate health and life sciences departments, high school life science classes, and, finally, consumers.

Kathy Anderson, sales manager at A.D.A.M. Software, recently spoke about her company's channel marketing strategy: "Physicians buy through word of mouth, medical schools buy through higher education channels, high schools purchase through secondary education channels, and so on. Rather than trying to find one software publisher to sell to all of these customers (a fairly impossible undertaking), we formed partnerships with leading textbook publishers and/or software distributors in each subcategory. Today A.D.A.M.'s sales are doing extremely well in all categories, and we've just released our third consumer title, *Life's Greatest Mysteries.*"

[Editor's note: For more information on A.D.A.M.'s products, check out their Web page at http://www.adam.com.]

This firm is currently working on 3D software that works with its DataGlove hardware to help people with limited mobility manipulate objects remotely using a 3D computer-based interface.

"About 10 percent of the U.S. population—25 million people—have physical disabilities. In the future, virtual reality technology will enable these people to accomplish tasks that would otherwise be denied because of physical limitations," says Walter Greenleaf, president of Greenleaf Medical.

Looking further into the future, there are other Apple technologies that are yet untapped in the health care market. Here's a short list of some of the potential applications that we think developers could capitalize on, based on our interactions with health care experts.

- *Digital photography.* Mac OS imaging technologies, as well as the QuickTake camera, are still underutilized in the health care field. For example, professionals at a burn recovery center recently asked Apple sales if they knew of a software solution that uses digital photography to help physicians and patients monitor the healing of severe burns.
- *Text-to-speech and speech recognition.* Apple's innovative text-to-speech technology is a great building block for software solutions that help people with speech disabilities. And Apple's speech recognition technology, PlainTalk, enables the creation of limited-vocabulary speech recognition solutions. Though general-purpose speech recognition is still a few years off, there's a tremendous potential for applications that help health care professionals, such as pathologists and surgeons, who need hands-free computer-based solutions.
- *Internet access.* With Apple's new Internet servers, it's easier than ever to set up a Web page to educate consumers and professionals about your health care-related titles and solutions. Many commercial health care developers, such as A.D.A.M. Software, have set up Web pages to let consumers learn more about their products. In addition, there are many Internet bulletin boards that provide support to people with chronic diseases, such as arthritis, that you can access to publicize your health-related titles.

An Apple a Day . . .

The pressure to streamline and improve health care in the United States is creating opportunities for Apple developers, systems integrators, and contract programmers. Though the health care market is enormous, it's not a market for "the weak of heart"—health care institution bureaucracies and the prevalence of host-based legacy systems make it a difficult sell for small Mac OS developers.

Apple currently has a strong presence in medical schools, in image-intensive physician specialties, such as radiology and pathology, and in mobile health care (with Newton technology). Another promising market for Mac OS commercial developers is in consumer health care education. The market for these titles will increase rapidly as the installed base of Mac OS computers, Pippin CD-ROM players, and Apple set-top boxes increases. ♣

Hadley Brandt (brandt1@applelink.apple.com) is an Apple sales engineer, based in Atlanta, Georgia, who works with many of Apple's large health care customers. Ken Seymens, (mr. ken@applelink.apple.com) is Apple's heath care information systems market development manager.

Bootstrapping and Beyond: Financing Ideas for Software Developers

By Dean M. Gloster, Farella Braun & Martel

You have brilliant programmers, artists, and interface designers, and a great idea for a compelling software title. You lack, however, one major ingredient: money. You need money to pay those talented people and unavoidable business expenses. Many entrepreneurs choose to finance their ventures through "bootstrapping"—self-funding by performing work for hire, drawing from personal accounts, and borrowing from relatives. But there are less painful funding approaches that can, in many cases, better position your business for future growth and markets. This article provides you with a variety of financing ideas for both new projects and companies.

Question #1: Project or Company Financing?

The initial question that you have to ask yourself is whether you're looking for *project* financing or long-term *company* financing. There are two different funding universes for each of these needs. The first universe is made up of software publishers or individuals who finance specific titles or projects. The second is interested more in a developer's overall company than in a specific title. Members of this group, which include venture capitalists and strategic investors, want to hear about your plans for growing your company, so that they can reap huge gains when it is sold or goes public. (For an overview of these options, see the chart "Financing Alternatives for Developers" on page 30.)

Title and Project Financing Alternatives

The good news for developers seeking funding for a specific title is that there's a standard industry financing conduit that makes this process fairly simple: software publishers. Many software publishers provide small development teams with hundreds of thousands of dollars in advances to create compelling titles. The publisher then worries about product fulfillment, marketing, distribution, the risks of unsold inventory, and just about everything else that keeps entrepreneurs awake all night. Unfortunately, the price tag for this money (and transferred risk) is substantial: The developer's share of title revenues is relatively small, and the publisher's development advances are usually recouped out of revenues that a self-funded developer would have earned.

A major CD-ROM entertainment publisher who funds the entire cost of original, outside title development typically pays a developer a base royalty of only 10 to 15 percent of net title revenues, less certain specified costs. Royalties for first-time developers are typically lower than royalties for developers with a track record, and some publishers pay lower royalties, which look more like flat-fee work-for-hire agreements. It may, however, be useful to know that even when a publisher will not budge on the base royalty, a developer can often obtain a step-up royalty after a certain number of units are sold. Thus, a 12 percent royalty might increase to 15 percent after 50,000 units are sold, increase to 20 percent after 100,000 units, then increase again when sales top 200,000 units. Though you may hear of publishers who pay higher percentage royalties, do your research: It may be because they have less effective distribution, and the higher royalty is illusory.

Approaching a Software Publisher

For a contact list of over 100 different publishers of multimedia and related software products, check the FTP site, ftp://ftp.netcom.com/pub/va/vance/pubs. Many of these publishers consider outside submissions.

How do you get to "yes" with one of these publishers? First, do enough market research so that you understand whether the publisher is dealing with outside developers, and whether your project is appropriate for the publisher's product line and marketing strategy. Sometimes this is a little hard to know, since publishers can change strategies. But if a publisher publishes only science fiction adventure games, they won't be interested in your medical clinic accounting software, no matter how wonderful it is.

Then contact the appropriate publishers to obtain their product submission form, nondisclosure agreement, or similar product submission kit. Before considering your idea or project, most publishers require you to sign some form of submission agreement. Software publishers do this because they are legitimately concerned about being sued. After all, they review hundreds of submissions every year, and a title they create in the future may be superficially similar to a submission they received from a past developer submission.

Developers queasy about this prospect can choose to deal only with the companies that have reasonable submission agreements and a decent reputation in the developer community.

The Proposal

Once you have obtained the submission agreement and signed it, you typically send the software publisher a proposal. The proposal usually contains

- a brief description of the title
- a summary of key features and product advantages, explaining how these will help the publisher sell lots of copies
- a more extended discussion of each of the different modules in the product
- a description of the development team and their qualifications
- an analysis of competing products and the market for the product

A well-done proposal for an entertainment or education title also contains sample art or story boards. Often a developer will send in a partially functional prototype or a short videotape showing the operation of the title.

What Do Publishers Want?

Publishers prefer to see submissions from people who have previously completed a software title. In the publisher's realistic view, starting projects is easy, but finishing them (especially on time, on budget, and with the features operational as promised) is difficult. It's less risky if a developer has a successful track record (and has proprietary tools that make developing the second title easier). In addition, as Gail Williams, manager of the Corel Home product line, says, "We want to see new and interesting ideas, rather than second generation 'me too' titles." Jason Everett, in charge of product submissions at Brøderbund Software in Novato, California, agrees. This year he will review over 1,200 proposals and product submissions. He advises, "The further along you are with working code and a prototype, the more likely it is that your title will interest us." Howard Soroka, director of product development at Philips Media Games, which has now branched out from CD-I titles to Mac OS– and Windows-based CD-ROM discs, adds that with entertainment titles "you have to have excellent art."

What does it take for a developer to close a deal? "Persistence and passion for your title," says Rachel Bernstein of Frog City Software, a San Francisco–based startup that recently closed a funding deal with SSI for its first strategy game. Before signing, she had approached about a dozen publishers and entered into discussions with five.

Other Ways to Finance Titles or Projects

There are many reasons why you may want to go outside the normal software publishing route. Perhaps you're locked in the new developer's "catch-22": Because your resume lacks a prior hit, publishers are unwilling to fund title development unless they see

Financing Alternatives for Developers

Method	Pros	Cons
Bootstrapping	Retain company ownership	Personal assets and credit at risk, hard to sustain rapid growth
Venture capitalists	Provide business advice and substantial funding	Loss of company equity; demand for an exit strategy
Strategic investors	Distribution channel access, and symbiotic benefits—you supply creativity, they supply capital	Potential for control issues
Private investors	Available to fund true startup companies	Hard to find, may not be knowledgeable in your field
Debt financing	Enables rapid growth	Hard to obtain; often requires immediate debt service payments

a partially functional prototype—but until you get the requested publisher funding, you're not able to fund development of the prototype. Maybe you're developing a new type of product or a niche-market application that's difficult to place with publishers. Or maybe you want a bigger return than the standard, modest royalties funded by most software publishers. So what other financing alternatives are available?

Some developers have obtained funding for specific titles from "angel" financiers (for more information about these private investors, see page 32). Some borrow necessary funds from a private investor to produce the title (or at least the prototype), and, in lieu of interest, give the private investor/lender a share of the title's royalties. Where there is a substantial educational, literary, or public policy aspect to the title, some developers have obtained grant financing. An excellent guide to grant sources is the *Directory of Computer and High Technology Grants*, 2d Edition 1994, written by Richard Eckstein and published by Research Grant Guides (407-795-6129; ISBN #0945078072).

Finally, a common approach by developers is to bootstrap a title, funding hard costs off limited savings, credit cards, bartering for services, and the like, while essentially volunteering their own time on speculation. Then, when they get a finished product, or at least a substantial prototype, they're able to approach publishers about advances.

Company Financing Alternatives

Many developers have larger ambitions than to simply fund the development of a title or product; they are looking for ways to finance their company. During the startup phase of a *company,* a developer often pays expenses out of the founders' pockets and initial operating profits. Keeping overhead low is crucial. Some companies are creative in putting together barter deals, using interns, offering royalty participations instead of cash, and giving people on-screen credit in lieu of larger payments. For many developers, however, the bootstrapping model of living off work for hire and publishers' advances is far too limiting. It also means giving up too much potential future revenue. Rapidly growing developers require substantial capital to continue to grow and to position themselves for anticipated future markets. At some point, they seek outside funding, often from venture capitalists, strategic investors, and private investors.

An excellent guide to grant sources is the *Directory of Computer and High Technology Grants,* 2d Edition 1994, written by Richard Eckstein and published by Research Grant Guides.

The Funding Process and Business Plan

The search for funding starts with the creation of a business plan that analyzes the market for proposed products and states how the company will compete effectively in that market. It explains what differentiates its product from others in the market, and what is exciting about its approach. The critical thinking necessary for creating a business plan is important in establishing an overall vision and longer-term business focus.

For entrepreneurs looking for outside company funding, the business plan also has to be an effective sales tool that instantly grabs the attention of investors with its concise executive summary. Venture investors see *hundreds* of business plans a year, and they want a brief executive summary without excessive adjectives.

Business plans are remarkably similar. After a short executive summary, the plan should go on to describe the company's business, the team members, competitive advantages, such as proprietary technology, and the competition. The business plan includes details on how investments will be used and the obligatory "hockey stick" financial projection graphs that show how the infusion of new capital will lead to explosive revenue growth. How long should a business plan be? If it's used to attract investors, as short as possible, and certainly not over 20 pages.

What Investors *Really* Care About

Do you need a business plan to force you to do the serious thinking about your strategic focus? Yes! But is it what gets you company financing? Not really. An interested private investor, venture capitalist, or strategic investor will often say "send me a copy of your business plan" when they first hear from you. But the truth is, to effectively raise capital, you need a much sharper focus than even a well-written 20-page business plan. You need to reduce what is unique and interesting about your company into a short pitch that you could practically write on the back of a business card—or that you literally could tell someone in the time that it takes an elevator to go up five floors. CEO Steve Blank of Rocket Science Games calls this the "back of the business card or elevator test." Not bad advice from someone who raised almost $10 million for a startup company in a crowded market.

The most common complaint that venture capitalists and other investors have about most business plans is that they "lack focus." A related complaint is that many entrepreneurs seeking company funding have never clearly articulated how their company is different from all of the others in the industry (or in the slush pile of unread business plans piling up in someone's reception area).

Once you have your business plan and (more important) your specifically articulated, narrowly focused, compelling pitch, you are ready to approach the different categories of potential investors.

Venture Capitalists

Venture capitalists (VCs) are often the first target source of financing for software entrepreneurs because they are easy to find, and because their avowed purpose is to invest in emerging growth companies. Traditional venture capitalists serve on the board of directors of the companies in which they invest. Venture firms such as Kleiner, Perkins, Caulfield & Byers and New Enterprise Associates, who have in the neighborhood of $800 million to invest, don't often invest sums much smaller than $3 to $5 million. Other VC firms (particularly "seed funds" such as Draper Associates) target much smaller investments, in the $100,000 to $1 million range.

Venture capitalists demand a *substantial* equity stake in return for their investment, and they target companies where they see *huge* potential gains. Critical issues for venture capitalists are the idea, the technology, the existence of an exit strategy (sale or initial public offering), and the strength of a company's existing management.

Mark Gorenberg of Hummer, Winblad Venture Partners, an Emeryville, California, VC firm that invests solely in software companies, explains his firm's analysis of management: "We're not interested in 'boat anchors'—white-haired former executives who were just dragged into a meeting with us to demonstrate management experience. We like to see *appropriate* management skills for your kind of business. If you are a software developer, that may be experience in project management."

Venture capitalists are not, however, in the business of funding titles or products. They fund *companies* they perceive as having exciting products and the potential for dramatic growth over three to five years. Because venture capitalists are inundated with unsolicited business plans, it is critical to have an introduction, and to concentrate on firms that fund developers. A list of 80 venture capital funds that have made recent software investments is also at the FTP site ftp://ftp.netcom.com/pub/va/vance/vcs. A helpful online resource that includes advice on dealing with venture capitalists can be is found at the Accel Partners Web site, http://www.accel.com.

Strategic Investors

A number of established companies have targeted multimedia and related software industries for acquisitions, joint ventures, or strategic investments. Entertainment, media, print, and existing software companies seek to create synergy with their existing distribution channels or seek to grow rapidly through investments in complimentary businesses. Sometimes these strategic investors are looking for investments for reasons other than pure profit: They may want to gain expertise, obtain access to technology, prevent a company from dealing with competitors, or become an instant distribution channel.

Examples of strategic investment partnership include Random House's investment in Living Books and Legend Entertainment; investments by Hearst New Media & Technology Group and Compaq in Books that Work; and the ABC/Capital Cities Creative Wonders joint venture with Electronic Arts.

For a contact list of over 100 different publishers of multimedia and related software products, check the FTP site ftp://ftp.netcom.com/pub/va/vance/pubs.

Private Investors

For companies that aren't far enough along to attract venture capitalists and strategic investors, there is still the alternative of private investors. A private investor ("angel" financier) can be an outstanding source of capital—if you can find one. The Apple-sponsored book *Multimedia Demystified* (published by Random House Reference) defines an angel as someone "who shares the project vision and wants to bring it into existence." Well, maybe. Typically, the angel shares money he or she has, in exchange for a cut of your profits. It is difficult to locate angel financiers unless they have a preexisting connection with the company principals or a substantial interest in—or sophistication about—the industry in which they are investing.

Although the venture capital community is far more visible, historically private investors provide far more capital to startup ventures. One way to obtain private investor funding is to approach executives and highly compensated technical people who are leaving much larger companies with golden parachutes, vested stock options, and the like. They often have the substantial capital, along with the relevant business or technical expertise, that it takes to build a successful startup company.

Debt Financing Alternatives

A sometimes overlooked funding source is debt, which allows entrepreneurs to grow rapidly, without sacrificing a substantial equity stake. Many small startup companies are funded (at least initially) on the credit cards of the founders. Unfortunately, debt usually requires immediate interest payments, creating potential cash-flow difficulties. Also, for borrowers who do not have a long operating history, traditional bank financing is impossible, because these institutions require a cash-flow track record and assets with a substantial liquidation value.

For companies that have a significant operating history but still do not qualify for traditional bank loans, other lenders, such as some commercial finance companies, offer slightly less conservative working capital loans, though typically at much more expensive terms. In addition, many computer vendors have arrangements with leasing companies to finance the lease of necessary equipment. And one company, Menlo Capital, specializes in providing debt financing to companies with venture capital funding. Several banks have special programs for women-owned and minority-owned companies, and there are special municipal programs targeted at job creation that make loans to small businesses.

Small Business Administration (SBA) loans are also an excellent financing source for amounts between $25,000 and $1 million. Because they are partially guaranteed by the SBA, lenders are more willing to make these loans, even when more risk is involved. Typically, SBA loans must be personally guaranteed, and with certain SBA loans other assets (such as a second deed of trust on the business owner's house) must be pledged as collateral. Currently, Citibank is aggressively marketing SBA loans to multimedia developers and other high-tech companies.

Beyond Bootstrapping

Many "great idea" software titles or companies have died on the vine because of cash-flow shortfalls. By looking beyond bootstrap financing, you may be able to more effectively expand your distribution channel or develop that next hit product. It just takes a little research and salesmanship to find a venture capitalist or investor that can help you turn your ideas into profit. ♣

Dean Gloster (glosterd@hooked.net), a partner in the multimedia group at Farella Braun & Martel (415-954-4472) in San Francisco, specializes in providing legal advice to software entrepreneurs and established multimedia companies.

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

November 27–December 1	Cupertino, CA
March 4–8	Cupertino, CA

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

November 13–17	Cupertino, CA
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

December 11–15	Cupertino, CA
January 15–19	Cupertino, CA
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

December 12–14	Cupertino, CA
January 16–18	Cupertino, CA
February 20–22	Cupertino, CA
March 19–21	Cupertino, CA

Newton Programming: Essentials 5 days/$1,500

December 11–15	Cupertino, CA
January 15–19	Cupertino, CA
February 12–16	Cupertino, CA
March 11–15	Cupertino, CA

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

November 27–December 1	Cupertino, CA
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

December 4–6	Cupertino, CA
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

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

Scripting with AppleScript/2 days/$600

December 11–12	Portsmouth, NH
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://www.info.apple.com/dev

- **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://www.info.apple.com/dev/). 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 Sites

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

http://dev.info.apple.com/
This site contains the Apple Developer Services and Products page, and is probably the most important World Wide Web page for you. Not only does it contain the online version of *Apple Directions,* and *develop,* the Apple Technical Journal, but this page also links you to a variety of other sites that give you access to the gamut of Apple's online developer support services.

http://www.apple.com/
This site contains the Apple Computer home page, with links that will eventually let you get to just about all the other Internet sites maintained by Apple, even the ones listed separately here.

http://www.info.apple.com/macos/
This is the Mac OS Web site. 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.

http://www.amp.apple.com
This is the site of the Apple Multimedia Program (AMP) home page. 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, as well as the links provided to other Internet sources about multimedia. It also includes the AMP Member Showcase, a searchable database of multimedia developers.

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.

http://www.info.apple.com/gomobile/
This site contains complete information about PowerBook computers and the full line of Apple mobile computing solutions.

http://coretools.apple.com/opendoc
This is the site of Apple's OpenDoc home page (note new address), featuring Developer Depot, where you can find the latest OpenDoc release, documentation, and tools, and Developer Showcase, from which you can download and sample actual OpenDoc parts!

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.

http://dev.info.apple.com/evangelism/powertalk/
Apple's PowerTalk home page, with 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 for no charge.

http://www.info.apple.com/qd3d/
Apple's QuickDraw 3D home page contains everything you need to know about QuickDraw 3D, including QuickDraw 3D applications you can "test drive."

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.

http://quicktime.apple.com
This site contains the QuickTime Continuum page with news and technical and marketing information about QuickTime.

http://qtvr.quicktime.apple.com
This is the location of the QuickTime VR page.

http://www.info.apple.com/gx/gx.html
This site contains the QuickDraw GX home page.

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

http://www.mae.apple.com
The Macintosh Application Environment (MAE) home page.

http://www.apple.com/documents/otherappleservers.html
This is the site of the Apple Internet Servers page. 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.

http://www.guideworks.com/
This is the location of the guideWorks home page; it contains so much

information about Apple Guide that you can think of it as the Apple Guide home-away-from-home page.

http://www.astro.nwu.edu/lentz/mac/programming/tools.html
This site is a terrific source for Apple and non-Apple Macintosh programming tools.

http://home.mcom.com/home/internet-search.html
This site contains the Internet Search page, which gives you access to InfoSeek, Lycos, and WebCrawler, three excellent Web search engines. If you use Netscape, you can reach this location just by clicking the Net Search button.

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.

http://www.freepress.com/myee/ultimate_mac.html
This site contains the ULTIMATE Macintosh page, including 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

These are locations that 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, along with your name (!). Send your suggestions to the Internet address a.directions@applelink.apple.com.

Special Notice: New Apple Directions List Server!

We've just started Apple Directions Express, which provides a regular digest of up-to-the-minute business news and information from Apple, in the most timely fashion possible. It includes pointers to Internet locations and other sources, for more detailed information. For more about Apple Directions Express, see this month's Editor's Note on page 2. You can subscribe by sending e-mail to adirections@thing1.info.apple.com. In the body of your message, type the string "subscribe <your real name>".

MPW List Server

Apple has also started a list server to get technical information to MPW users. To subscribe to it, send e-mail to listproc@abs.apple.com. In the body of the message, type the string "subscribe mpw-dev <your real name>". Leave the subject line blank or put a nonsense word in it.

http://www.euro.apple.com/
This Apple site 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.

http://www.info.apple.com/pacific/
This is the Apple Pacific home page, with information about Apple offices and developer support in the Pacific region, including Japan, Australia, Canada, and Latin America.

http://dev.info.apple.com/technotes/
This is the Web site for the Apple technical notes series.

http://pippin.apple.com
The Pippin Web page contains the latest information about Apple's PowerPC processor–based, low-cost CD playback device.

http://www.info.apple.com/newton
http://dev.info.apple.com/newton
These are Apple's Newton Web pages; the first is called the World of Newton, and it houses a variety of information about the Newton platform, primarily for customers. The second is the Newton developer site.

http://www.eworld.com/
Go to this location to find eWorld content and services on the Web.

http://www.digitool.com/
If you're looking for information on Macintosh Common Lisp (MCL), this is the place to go. Digitool's Web page contains information on MCL 3.0 and other MCL products, as well.

http://hyperarchive.lcs.mit.edu/HyperArchive/Abstracts/snd/util/HyperArchive.html
Occasional *Apple Directions* contributor David Gleason recommends this site, which is maintained by some of the good folks at the Massachusetts Institute of Technology. It contains some 200 Macintosh applications and utilities that can be downloaded for the price of connection charges.

http://www.ape.com/webstar/
This is an interesting site database of all the Macintosh computer-based Web sites its owner can find. So far, it contains more than 900 entries, sorted by country, type of Macintosh server, network speed, and other criteria. It also lists outstanding Macintosh sites and some losers that refuse to move to a Macintosh server solution.

http://www.rmii.com/~rbruce
This is a non-Apple Web site called the Ultimate Newton Page; we learned of it from its owner, Rob Bruce.

http://pilot.njin.net/~msproul/index.html
This site also comes recommended to us by its owner, Mark Sproul. It contains useful technical information for PCI and Open Transport developers.

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.

http://www.cs.brandeis.edu/~xray/mac.html
Nathan's Everything Macintosh page is another 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.

http://www.cilabs.org/
The location of the CI Labs home page, which contains a great deal of OpenDoc content.

http://rever.nmsu.edu/elharo/faq/vendor.html
The Macintosh Vendor Directory, a directory of companies that make and sell products for the Macintosh computer.

http://www.metrowerks.com/
This is the Metrowerks Web site, with information about its CodeWarrior PowerPC development tool.

http://www.nisus-soft.com/~nisus/
The location of the Nisus Software home page, which we list 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!

ftp://ftp.sri.ucl.ac.be/pub/
A French reader alerted us to this FTP site, where you can find French versions of Macintosh Internet software, including Eudora, Fetch, Finger, FTPd, Gopher Surfer, NCSA Mosaic, NCSA Telnet, TurboGopher, and many other applications. An affiliated World Wide Web site (http://www.sri.ucl.ac.be/SRI/jpk/logIntMacFr.html) describes—*en Français*—what's available at the FTP site.

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. ♣

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. 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