The Developer Business Report

September 1995

AppleDirections

Inside This Issue

Apple News

New Additions to the Power Macintosh Family Employ PCI Bus

New Systems Ship With Open Transport, Variety of Other New Software

At Macworld Boston, Apple Computer, Inc., added four competitively priced systems to the growing Power Macintosh family, the Power Macintosh 7200/75, the Power Macintosh 7200/90, the Power Macintosh 7500/100, and the Power Macintosh 8500/120 computers. The Power Macintosh 8500/120 employs the second-generation PowerPC RISC processor—the PowerPC 604—while the other systems are built around PowerPC 601 processors.

The new systems join the recently released Power Macintosh 9500 computer in employing Peripheral Component Interconnect (PCI), the industry standard expansion bus found in many Intel-based PCs, which Apple has adopted as the expansion bus for the Power Macintosh line. The PCI bus should provide performance up to three times faster than that provided by the fastest NuBus™ systems. Any PCI card that complies with the PCI 2.0 specification will work with the new computers, with the addition of the appropriate software drivers.

please turn to page 10

Strategy Mosaic

Seeing Through Windows—And Into the Future

By Gregg Williams, Apple Directions *staff*

Apple's Plan to Respond to the Competition and Promote Its Vision

August 24, 1995, will be a great day for the Macintosh community, and not primarily because of Microsoft's official announcement of Windows 95. The U.Q. (Unreality Quotient) of the computer industry will begin to drop as, day by day, more people begin using Windows 95 and find that—gasp!—they *still* have a Windows-based machine on their desktops.

August 24 will be a great day for Apple Computer, Inc., because—finally—it can begin the process of competing with a real product, not a cloud of perfumed promises.

But the industry is nervous, almost hysterical. "This is your last stand, Apple!" they shrill (for the *n*th time). "Bill Gates is at the door, and he's got a *really big stick!* Oh, whatever are you going to *do*?"

"Not to worry—we know what we're doing," Apple says (also for the *n*th time).

Has anything changed? Is anything going to be different this time? You're skeptical, and you have every right to be. As Apple's business partner, you deserve to know how we're going to defend the Apple platform (and, not

please turn to page 5

AppleDirections

Volume 3, Number 9

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, Alex Dosher, Cindy Gallie,
David Gleason, Kris Newby, Caroline Rose

Manager, Developer Press
Dennis Matthews

Manager, Apple Developer Periodicals
Mark Bloomquist

Production Manager
Diane Wilcox

Prepress/Film
Aptos Post

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

Guy Returns

Way back in the 1980s, one of the key people responsible for introducing the Macintosh computer wanted to be sure there was plenty of great software to run on his new baby. He borrowed a term to describe his efforts to encourage software development: The term was *evangelism.*

Of course I'm talking about Guy Kawasaki, someone I imagine is familiar to many of you as the original Apple evangelist. Guy left Apple after helping assure the success of the Macintosh computer to launch a couple of software companies of his own. He returned to Apple in July as an Apple Fellow, primarily to work with Senior Vice President Dave Nagel and assure that Apple is doing everything in its power to make developers successful.

I thought that you'd want to hear from Guy as soon as possible, so Gregg Williams and I talked to him a couple of weeks after his return. Here are some of the things he had to tell us.

Apple Directions (AD): What do you hope to accomplish by returning to Apple?
Guy Kawasaki: On a tactical level, my goal is to ensure that cool applications appear on the Macintosh computer before any other platform. On a higher level, my goal is to reinvigorate the fervor and zeal of developers, early adopters, and even Apple employees.

AD: What are the most exciting things about Apple's platforms today?
Guy: The overall integration of technologies such as OpenDoc, QuickDraw 3D, QuickTime VR, Thread Manager, Drag and Drop—basically, stuff that enables developers to create the apps they always wanted to—and the performance of the PowerPC chips.

AD: Don't you see developers abandoning Macintosh because of the large installed base of Windows machines?
Guy: Actually, this isn't happening at all—it's a story I planted so that Apple would ask me back [laughs]! I've been on the boards of several cross-platform companies, so I know that you don't just take your Macintosh volumes and multiply them by five to figure out how many Windows copies you can sell. Maybe it's one and a half to two times, because there's far greater competition in the Windows market.

AD: Do you think you can get Windows companies to do Macintosh software?
Guy: I'm going to give it my best shot, but on the surface, it looks difficult to convince a Windows developer why it should create a Macintosh version for the smaller Macintosh installed base.

More importantly, I think that the overall quality of Macintosh software is very high. Windows companies know they can't just ship a mediocre product and expect Macintosh owners to fall all over themselves to buy it just because it was available on Windows. Making software available on Windows first isn't exactly a selling point to a Macintosh owner, if you know what I mean.

AD: What would you be doing if you were a developer?
Guy: You forget that a mere six weeks ago, I was a developer. We saw a market for a dedicated e-mail product, so we seized it. This product could have been done three years ago, but it wasn't.

Our plan was to ship a Macintosh version to validate the market and then port the Macintosh version to Windows. We had two reasons for this order: First, we love Macintosh. Second, we figured that if we created a successful Macintosh product, we could always go to Windows. The converse is not true.

AD: What are the most significant changes in the software market since you started Apple evangelism?
Guy: There was a time when if a Macintosh product booted, it would sell. Things aren't that easy now. However, back then the people "in the know" predicted that no one would do Macintosh software because there was such a large MS-DOS installed base. Looks like they were wrong. I hope they'll be wrong again.

People have been predicting the death of Apple for a decade now. If I had a dollar for every time I was asked if Apple could survive a new version of Windows, I could buy a house with a garage bigger than Bill Gates's plantation.

AD: What do developers have to do to be successful in the market?
Guy: Create a great product—it's as simple as that. I have yet to see a great Macintosh product fail because of a lack of distribution. Of course, if you're a developer, who are you going to blame: yourself or "the channel"? Wimps need not apply.

AD: Why do Apple's relations with developers appear to be poor?
Guy: Arrogance. A few years ago when Windows wasn't as good a rip-off of the Mac OS, Apple had the attitude that it was a privilege and honor to write Macintosh software. That arrogance rubbed developers the wrong way. This arrogance is being purged from Apple through all possible orifices. In a sense, Windows 95 is one of the best things that happened to Apple because it will force us to get tougher and more competitive.

• • •

If you have feedback or questions for Guy on the subject of what Apple should be doing, you can participate in a two-way mailing list (where every message sent to the address gets seen by everybody else) by sending e-mail to listproc@abs.apple.com. To subscribe to the individual messages, put the following in the body of the e-mail message:
SUBSCRIBE SEMPER.FI <your real name>

To subscribe to a digest that contains all the messages of the past 24 hours, sent to you once a day, send a separate message to listproc@abs.apple.com with the following text in the body of the e-mail message:
SET SEMPER.FI MAIL DIGEST

Guy also suggests you check out the Web page of third-party hardware and software products at
http://www.info.apple.com/dev/thirdparty/third_party.html

He strongly encourages you to fill out the form located there that allows you to add your products so that everybody on the 'net can find out about them.

Paul Dreyfus
Editor

IndustryWatch: News & Perspective

Not-So-Subtle Messages

Radius Ships Mac OS Compatibles

In July, Radius shipped its first PowerPC processor–based Mac OS computers. The company shipped high-end Mac OS workstations and a second type of system based on the Power Macintosh 8100 computer with a price about 10 percent lower than Apple Computer's. Additionally, Radius said it would soon begin to ship a Mac OS–compatible system based on Apple's recently released Power Macintosh 9500 computer. Radius becomes the second company, after Power Computing, Inc., to ship Mac OS–compatible computers. Radius will target its existing desktop publishing, design, and video-editing customers with the new computers.

Implications/Opinions: Even if Radius computers don't make a huge, immediate impact on the market, they can be expected to give a boost to Apple's efforts to license the operating system and the Mac OS platform as a whole. According to the *Wall Street Journal,* industry experts expect Radius to sell approximately 125,000 Mac OS systems by year-end. Radius expects that within a year, revenue from its Mac OS-compatible computer sales will equal the company's current annual sales of $320 million. In addition, another Mac OS–compatible vendor, Power Computing, expects Mac OS–compatible system revenues of $100 million in its first year. If nothing else, sales of the Mac OS–compatible systems will give you thousands of new customers for your products, a number that you can be sure will grow.

The Elegant and Stylish Macintosh Computer

"Apple's New Performa Looks Like a TV Set But is a Dazzling PC." That's the headline of Walter Mossberg's widely read Personal Technology column in the July 13, 1995, *Wall Street Journal.* In the column, Mossberg says the design of Apple's recently launched Macintosh Performa 5200CD computer "sets a new standard" in all-in-one system design and proves that Apple "still builds a great computer." We can't help but agree with him, although we take exception with the points he makes later in the column about Windows 95 closing the gap between Windows and the Mac OS.

He saves the best for near the end of the article, saying,

> *The Mac is still more elegant and stylish [than Windows-based PCs], still more tightly integrated, with better links between software and hardware, because a single company makes both the computer and the operating system. Well-designed, aggressively priced models like the Performa 5200CD still make the Mac distinctive.*

October *Apple Directions* Online

The October issue of *Apple Directions* will be available on AppleLink, the Internet, and eWorld by September 15, at the following locations:

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

Internet: select Developer Services and Products at the location www.apple.com.

eWorld: in the Apple area of the Computer Center.

You can view the complete article on the Internet's World Wide Web at http://ptech.wsj.com. Also, you can read more about the Macintosh Performa 5200CD computer in "Apple Ships New PowerPC 603–Based Systems for Consumer Market" on page 1 of the August 1995 issue of *Apple Directions.*

Implications/Opinions: Mossberg hits the primary Macintosh advantage square on the nose: No matter how vastly the Windows operating system is improved, the computers it runs on will never be as tightly integrated as is the Macintosh computer with the Mac OS. This is an advantage Apple will be proving and improving—and touting—over and over again in the face of the Windows 95 juggernaut. You can do the same thing, both by writing software that follows the human interface guidelines and the Macintosh Toolbox APIs, so it behaves just as Macintosh users expect it to, and by demonstrating Macintosh integration to customers and would-be customers.

In the words of Michael Spindler to a group of reporters gathered in Cupertino, California, last month: "Our marketing strategy this fall will be to say 'Here's what a Mac does.' When customers look at a Macintosh system compared with one running Windows 95, we win every time."

Microsoft Indicates Life of Windows 95 May Be Limited

As Gregg Williams mentions in this month's Strategy Mosaic, Microsoft Chairman Bill Gates says in the June 12, 1995, issue of *Computer Reseller News* that his company ". . . will focus on [Windows] NT as our mainstream platform two to four years from now when 16 MB on the desktop is common."

Similarly, in a widely reported conference call with analysts July 17 that caused Microsoft stock to drop more than 6 percent the next day, Microsoft Chief Financial Officer Mike Brown predicted that Windows 95 sales will begin to wind down during Microsoft's fourth fiscal quarter (March to June 1996). He added that Microsoft will defer recognition of a large part of its Windows 95 revenue over the next 18 months, possibly to pay for support or upgrade products.

Implications/Opinions: Why would Microsoft talk about the advantages of Windows NT, and begin to talk about the limits of Windows 95 sales on the eve of its introduction? We think it makes sense for two reasons. First, Windows NT isn't limited to only the Intel hardware architecture; Windows NT allows Microsoft to extend its reach to RISC-based machines as well.

Second, you might remember back a couple of years ago when Microsoft told customers and the industry that, if they had problems with Windows 3.1, they should just wait for Windows 95. Customers bought, and the industry waited. Now, despite the problems that will be associated with at least the first release of Windows 95—its bugs, its lack of robust memory protection, that it provides preemptive multitasking if a system employs *only* 32-bit applications (of which there aren't many), to name a few—it wouldn't be out of character for Microsoft to deliver a similar "buy now, be satisfied later" message.

Special Report Gives Insight Into Home Market

The June 19 edition of the *Wall Street Journal* featured a 36-page special report on home technology called "The Revolution Comes Home." The report contains 16 separate articles on different aspects of home technology, most of them having something to do with computers.

Implications/Opinions: In the report's own words, "For years, anybody looking for a glimpse of the latest technology knew where to go: the office. . . . Not these days. Now home is where the action is." We couldn't agree more, as we've said in our Market Research Monthly column (see the August 1994 issue, page 20). Given just how crucial the home market has become, we think you can't get enough information about it if you're going to remain competitive. You can obtain copies of the report for $4 ($2 for each additional copy) to cover postage and handling by writing Technology, Dow Jones & Company, Inc., 200 Burnett Road, Chicopee, MA 01020-4615. Make checks payable to Dow Jones & Co.

A Day to Forget

In researching this month's Strategy Mosaic, "Seeing Through Windows—And Into the Future" (page 1), Technical Editor Gregg Williams discovered what else happened on August 24 (the date of the Windows 95 release) in years past.

Here are some of the events that occurred that day in history:

1985: Morrie Ryskind, co-author of the most famous Marx Brothers movies (including *Coconuts* and *Animal Crackers*), dies at age 89.

1980: Government investigators report that organized crime has gained extensive control of the mozzarella cheese and pizza industries.

1960: The U.S. Surgeon General approves the use of a live-virus, oral polio vaccine developed by Dr. Albert Sabin.

1954: Bela Lugosi, 73, marries Hope Lininger, 39, a movie cutting clerk, in Hollywood.

1944: *Science* magazine reports that penicillin mold inhibits the growth of tuberculosis germs.

1814: British troops capture Washington, D.C., and burn the White House during the War of 1812.

1572: Massacre of St. Bartholomew's Day begins in Paris.

Implications/Opinions/Wisecracks: We can't detect any legitimate connections between any of these occurrences and the Windows 95 release. But some of you jokesters out there may be wondering if, by picking the 181st anniversary of the storming of Washington, Microsoft isn't sending the Federal Trade Commission the subtlest of messages.

Giving Credit Where It's Due

When we wrote last month's item about the Information Superhighway anagrams, we didn't know who'd come up with them. Since then, we've learned that they were created by Mike Morton, a software developer in Hawaii, using anagram-making software he created called Ars Magna.

Apple Directions Technical Editor Gregg Williams ran the words "Apple Directions" through Ars Magna; it returned a plethora of anagrams, including *Inspired Polecat* and *Pale Predictions.*

If you'd like a free copy of Ars Magna, you can download it from one of these FTP sites:

ftp://ftp.hawaii.edu/mirrors/info-mac/text/ars-magna-ii-111.hqx
ftp://mirrors.aol.com/pub/mac/game/board/arsmagnatwo1.11.cpt.hqx ♣

Strategy Mosaic

Windows

continued from page 1

coincidentally, the platform your business is based on).

So I went to the Powers That Be—Michael Mace, Director of Worldwide Platform Marketing, and Allen Olivo, Director of Worldwide Advertising—to find out what Apple is going to do. The rest of this article summarizes what they told me.

What Is Windows 95?

Apple's view of Windows 95, one that we will be making public in various ways, is actually quite simple: "Windows 95 is a nice upgrade to Windows 3.1. We think it will sell quite well. But it's still not a Macintosh. Period."

Why Should Apple Respond to Windows 95?

There are two answers to this question. First, Apple will respond because the competitive marketplace requires it; if Apple didn't, much of the press and the public would take Microsoft's claims at face value, thus strengthening Microsoft's position—and Apple management has no intention of letting that happen.

Second, a response to Windows 95 is not only necessary but desirable, because it gives Apple an opportunity to expound on its leadership in and vision for the personal computer industry. (More on that later in this article.) In fact, Apple intends to make sure that people know that, though a computer running Windows 95 is a great improvement over its predecessor, it's still not as good as a computer running the Mac OS.

How Will Apple Respond?

During and directly after the Windows 95 introduction, Apple will run print ads that will explain the advantages of the Macintosh platform and educate the public on the limitations of Windows 95. The basis of comparison, Apple will contend, is not the Mac OS versus Windows 95. Rather, the proper comparison will be Macintosh computers running the Mac OS versus Intel-based PCs running Windows 95 (which I'll call the "Wintel 95 platform" in this article).

After the initial interest in Windows 95 dies down, Apple will concentrate on communicating the vision of our platform, but it will continue to make direct comparisons (to Microsoft and other competitors) as opportunities present themselves. In other words, while Windows 95 is news, Apple will make news off it. Once it's not, Apple will continue promoting its own vision in whatever way makes the most sense.

OK, So What *Is* Apple's Vision?

Apple is in the business of advancing the state of the art in personal computers. And if you don't want to take our word for it, here's what Tom Halfhill said in the December 1994 issue of *BYTE* magazine:

> *It would not be an exaggeration to describe the history of the computer industry for the past decade as a massive effort to keep up with Apple. In 1984, critics derided the Mac for its appliance-like simplicity, but it went on to pioneer or popularize almost every innovation in personal computing: the GUI, desktop publishing, built-in networking, plug and play, integrated multimedia, visual programming, hypertext, 24-bit color, the global Clipboard, undo, voice control, built-in color calibration, dynamic memory allocation, SCSI, and even 3 1/2-inch floppy drives.*

Take a look at the table "Macintosh innovations and Wintel 95" on page 6. I think it substantiates two important points:

- Apple has been the technology leader in the past decade—features that are just now appearing on the Wintel 95 platform have been on the Macintosh platform for years.
- In general, the Wintel 95 platform is simply an improved version of the current generation of personal computing. It still hasn't caught up with some features that the Macintosh platform has had for years, and in most cases its implementation of technologies is inferior to Apple's.

But that's only half the story. The table "Beginning the Next Ten Years" on page 8 shows that Apple continues to be the industry visionary; it lists technologies that Apple believes will be important in the next ten years. I believe that, just as the computer mouse, graphical user interface, and desktop metaphor are universally accepted in today's personal computers, the features in this table (and other features, yet to come from Apple) will be an accepted part of personal computing ten years from now.

Note that many of these technologies have been around for a few years. This is not a sign that they aren't going to make it—rather, it's a sign that they are continuing to gain acceptance, a process that doesn't happen overnight. (To make this point, I often quote myself in the form of Williams's First Law of Technology Adoption: When contemplating the adoption rate of new technologies, think in terms of *half-decades*.)

From this point of view, Apple's next-generation technologies are doubly important: Not only do they establish Apple's vision of tomorrow's computers, many of them also have several years' lead over anyone wishing to copy them. (QuickTime 2.0 versus Video for Windows is a good example of this.)

You may say, "Vision is all very well and good, but it won't save you if Microsoft takes over the industry in the next year or two." True, which leads us to the next question.

What Are Apple's Competitive Strengths Today?

In other words, "What does Apple have going for it that will allow it to survive against the immediate threat of Windows 95?"

Apple has condensed the advantages of the Macintosh platform into four main points, each of which will figure heavily into Apple's future advertising. I won't go heavily into each point—you've heard the arguments before—but you can be sure that future Apple ads *will* explain them in great detail. Here are the four points:

- First, Macintosh computers are more powerful than Intel-based computers. You've seen similar figures before, but here are the figures of the latest Power Macintosh/Pentium study, conducted by Competitive Assessment Services. This company did a composite test timing ten different applications (including such applications as Microsoft Word and Excel, Fractal Painter, FrameMaker, and Mathematica) doing 58 different tasks in the areas of productivity, graphics/publishing, and scientific applications.

The company found that, overall, the Power Macintosh 9500/120 computer was 52 percent faster than a similarly configured 120-MHz Pentium-based machine. And on graphics and publishing applications, the Power Macintosh 9500/120 edge was even greater—85 percent faster than a similarly configured 120-MHz Pentium-based machine.

Granted, this test was run with a Pentium computer running Windows 3.1, but Windows 95 is said not to run appreciably faster than Windows 3.1, so the percentages just given should not change much when the same tests are run on a Pentium computer and Windows 95.

A recent ad for the Intel P6 chip, under the heading "Where No Processor Has Gone Before," said the following:

> *Have you ever thought about what tomorrow's software will be like? Consider all you'll be able to do with the new applications like 3D imaging, voice recognition, and personal videoconferencing. . . . These software programs will be larger and will require increased performance from your microprocessor.*

Thanks for the insight, Intel—seriously. I agree with *most* of what you said, but the figures just given show that the Macintosh platform and the PowerPC processor have already "gone before" in delivering both the technologies and the processor power needed to make next-generation applications commonplace—in fact, Apple is doing it right now.

- Second, Macintosh computers are easier to use than Wintel 95 computers. I won't bore you with the details here—but you can be sure that Apple will show customers that, even though Windows 95 is a definite improvement over Windows 3.1, a Macintosh computer running the Mac OS will still be easier to use (and, equally important, more useful) than a Wintel 95 computer.
- Third, a Macintosh computer delivers more advanced multimedia and graphics than a Wintel 95 computer. Everybody knows

Macintosh Innovations and Wintel 95

Date introduced on Macintosh	Technologies present on Wintel 95 and Macintosh platforms	Mac technologies still not on Wintel 95 platform
1984 (actually 1983, on Lisa computer)	WIMP (windows/icons/mouse pointer) interface	
1984		3 1/2-inch floppy disk drive support
1984		Desktop metaphor
1984	Direct manipulation of files (no File Manager)	
1984	Long filenames	
1985		Built-in networking (LocalTalk)
1986		Built-in SCSI
1986	Plug-and-play hardware	
1987	Multitasking (MultiFinder)	
1987	Visual programming (HyperCard)	
1991	"Shortcuts" (aliases)	
1991	32-bit OS	
1991	Serverless file sharing	
1992		System-wide scripting (AppleScript)
1992		Unrestricted support for multiple languages in same document (WorldScript)
1992	Built-in color calibration (ColorSync)	
1992	Time-based data architecture (QuickTime)	
1993		Speech input (PlainTalk)
1993		Telephony support (Telephony Manager)
1994		RISC technology (PowerPC)
1994	Universal mailbox (PowerTalk)	
1994	Window-based virtual reality (QuickTime VR)	
1995		3D graphics (QuickDraw 3D)
1995		Videoconferencing (QuickTime Conferencing)

This table lists technologies popularized by Apple, by the date of introduction, and notes whether or not a technology is available on the "Wintel 95" platform. The purpose of this table is to show how far Apple is ahead of the competition in personal computer innovation. Note that this table does not address the quality of each platform's technologies. Apple believes that the higher quality of its technologies contributes to the overall superiority of the Mac OS platform.

what a nightmare it is to get multimedia CD-ROM titles to run on a Wintel machine, and Windows 95 will not remove all the problems. Apple will be sure to let customers know that—and remind them of the Macintosh platform's QuickDraw 3D, QuickTime VR, and superior video input/output and multimedia authoring tools, among other advantages.

- Fourth, a Macintosh computer offers compatibility—both with the Wintel world and with itself. The Macintosh platform offers built-in compatibility with DOS/Windows files and optional compatibility with DOS and Windows programs through either add-in hardware cards or third-party software (Insignia's SoftPC and SoftWindows™).

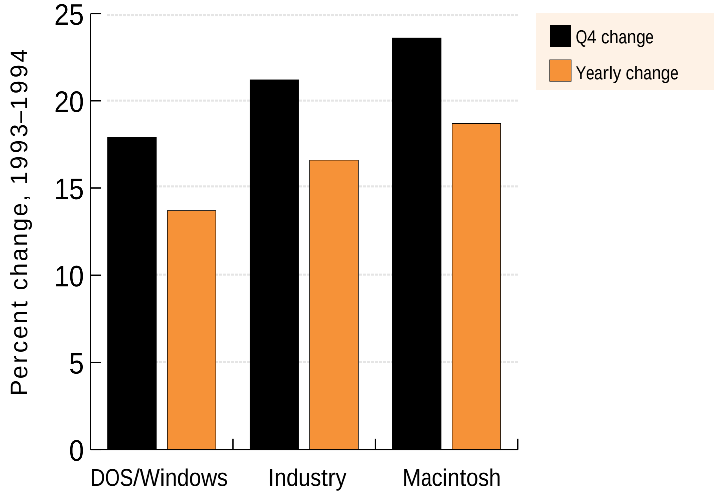
But let's not forget the superior compatibility that the Macintosh platform offers with itself—software applications and extensions that run without the extreme configuration problems that plague many new pieces of DOS or Windows software when they are added to an Intel-based computer. (And by the way, the June 12, 1995, issue of *InfoWorld* reported Microsoft as having "conceded" that "as many as 200 mainstream Windows 3.1 applications will not run on Windows 95.")

What Are Apple's Business Prospects?

It seems that in the computer industry, one sure-fire conversational gambit (somewhat akin to talking about the weather) is asking the question, "Do you think Apple is going to survive?" Apple's answer to that is, "Not only do we expect to survive, we expect to *thrive."*

Let's look at the facts. Macintosh sales are currently over 4 million units per year. Apple is the number one personal-computer vendor to the education and home markets worldwide, and it's number three in sales to business.

Sales of Macintosh Software Outpace Intel-Based Software in 1994



This figure is based on software sales figures from the Software Publishers Association for 1993 and 1994. The sales figures for "Industry" includes Macintosh, Intel, and other platforms.

Also, Apple has sold more RISC personal computers than all other vendors combined.

And if that's not enough, by the time you read this, Apple will have sold its 20-millionth Macintosh computer. (That's 20,000,000 Macintosh computers—a nice round figure!)

But perhaps the least-publicized indicator of Apple's health (though you read it first on page 3 of the June 1995 issue of *Apple Directions*) are sales figures from the Software Publishers Association that show that Macintosh software sales from 1993 to 1994 outpaced those of software for Intel-based personal computers.

How can that be? True, everybody talks about the phenomenal growth of the Windows platform (reported elsewhere as being 42 percent), but what's not obvious is that many of those sales are cannibalizing the DOS software market (which was reported as *decreasing* by 40 percent). It turns out that when you consider all the software (both DOS and Windows) being purchased by owners of Intel-based PCs, the growth of that market from 1993 to 1994 is less than the industry average, and much less than that of Macintosh-compatible software. The graph "Sales of Macintosh Software Outpace Intel-Based Software in 1994" shows the overall percentage changes; the article in the June 1995 issue of *Apple Directions* gives more detailed figures.

Granted, the market for Windows-compatible software is still larger than that of Macintosh-compatible software, but these figures clearly show that the Macintosh market is growing and that it is a good place for software developers to make money.

What Is Apple's Advertising Strategy?

Corporations are *big,* and they can't change direction overnight. I think Apple has done an excellent job of turning itself around as a company, but it's important to note that some of the key actions that began that process happened as far back as 1990 (lower-cost Macintosh computers) and 1991 (the alliance with IBM and Motorola to develop the PowerPC processor).

Why do I mention this? Because it's my belief that, just as Apple began turning its business around in the early 1990s, it's just about to begin the process of turning its advertising image around. So yes, I think that finally, finally, *finally,* Apple is going to begin the advertising that should get the Macintosh platform more of the credit it deserves with the public. It won't be as much as anyone wants, and it won't come as quickly as anyone wants—but before it can occur at all, it has to begin, and I think that's about to happen.

My high opinion of Apple's advertising strategy comes after having a talk with Allen Olivo, Director of Worldwide Advertising for Apple, and it's largely because of Olivo's thoughts that I believe that an important change is at hand.

Olivo believes very strongly that Apple must simplify its messages, then communicate them consistently worldwide. He believes in spending our less-than-infinite advertising dollars more wisely to maximize the effect on the market, and he plans to do hard-hitting ads that directly target competing products. Although, he said, we don't want to get into a brand-name fight like the one between AT&T and MCI, "If I have to name names, I'll name names."

Why do I think this time's different? Well, I've never heard Apple management say *that* before.

Olivo said that Apple's ads will emphasize three key messages:

- For the business market, Apple will promote its newest Power Macintosh computers as the most powerful available.
- For the consumer market, Apple will emphasize its leadership in multimedia. (Apple expects strong returns in its back-to-school and Christmas sales, and uncertainty over Windows 95 may help things out.)
- In general, Apple will emphasize the superiority of the Macintosh platform over the competition.

Capturing the Customer

Apple hasn't been very good at this in the past, and this is something Olivo wants to change. He says that Apple ads will focus on giving the public answers to the following key questions:

- Why should I buy Apple?
- Which label/model of Apple computer should I buy?
- Where can I buy it?

By getting these questions answered for the customer, Apple's advertising should clear the way for the only customer

Beginning the Next Ten Years

Date introduced	Technology	Significance
1992	AppleScript and Open Scripting Architecture	Network-wide, standardized application scripting
1993	AV architecture	Architecture for manipulating audio and video
1993	GeoPort	Open, cross-platform standard for connecting computers to telephony equipment
1993	PowerTalk	Easier, more powerful access to all incoming and outgoing electronic communication (e-mail, fax, phone, databases)
1994	Apple Guide	Online, task-based training, leading to active assistance in the future
1994	DOS/Windows compatibility in hardware	Compatibility with the DOS/Windows world, for those who need it
1994	Macintosh Drag and Drop	More intuitive manipulation of data between documents
1994	PowerPC processor–based computer	Using RISC technology to create the next generation of personal computers
1994	QuickDraw GX	Resolution-independent imaging, advanced typography
1994	QuickTime 2.0	Second-generation architecture for video and other time-based data
1994	QuickTime VR	A new way to interact with 3D, photographic environments (window-based virtual reality)
1995	Open Transport	Protocol-independent networking and communications
1995	OpenDoc	Cross-platform component software and compound-document architecture
1995	PCI	Industry-standard architecture for expansion cards
1995	QuickDraw 3D	3D graphics with standardized file format, human interface
1995	QuickTime Conferencing	Open, cross-platform standard for long-distance collaboration using video, audio, and shared documents
1995	System Object Model (SOM)	Cross-platform standard for software object reuse
1995	Telephony Manager 2.0	Second-generation API for writing implementation-independent telephony applications
1996	Common Hardware Reference Platform (CHRP)	Open standard for PowerPC processor–based computers that can run the Mac OS and other operating systems

These are some of the technologies that Apple has already invested in that Apple believes will drive the personal computer industry in the next ten years. More innovations will follow in the future.

response that matters at this stage: going out and making the purchase.

Getting these questions answered also helps combat one of the greatest barriers to an Apple sale in most computer stores: the overwhelming presence of Windows-compatible computers. It's an unarguable fact that when you go into most computer stores, the majority of shelf space is devoted to Windows-compatible computers, and few salespeople will by default steer customers to whatever Macintosh models are available.

Establishing a predisposition toward Macintosh through advertising is one good way Apple can combat the shelf and mindshare imbalance in most computer stores. If customers are committed to buying a Macintosh computer before they enter the store, it won't matter that the store has a larger selection of Windows-compatible computers—the only question they will ask the salesperson is, "Where are your Macintosh computers?"

First-time customers present a special challenge—they don't know the market, and they're afraid of making a bad choice. But they also offer a rich opportunity, because they represent a very large segment of personal computer sales today.

Apple's ads to first-time customers are aimed at convincing them that they can't go wrong buying a Macintosh computer. In particular, they will stress that, compared to the competition:

- You can do more with a Macintosh computer.
- It's easier to get things done on a Macintosh computer.
- Macintosh computers can grow to meet your future needs, so you don't have to worry about your computer becoming obsolete.

Apple Advertising Goals (Microsoft, Start Reading Here)

As I said earlier in this article, one of Apple's goals will be to promote Apple products and counter Windows 95 claims.

But Apple plans to do more than that. Rather than simply react to Windows 95, Apple plans to use acknowledgment of Windows 95 ("It's a nice upgrade to Windows 3.1") to discuss the vision of the Macintosh platform and how Apple is delivering next-generation computing while the competition is—imperfectly—trying to deliver today's computing experience. Most of this campaign will be print ads to Macintosh, PC, and general magazines. (Of course, Apple's holiday-season advertising won't be limited to print ads only; you'll definitely see television ads for Apple as well.)

Advertising that targets Windows 95 explicitly will diminish as Windows 95 becomes old news, although Apple will continue to make direct comparisons when it allows Apple to do some real damage to the competition. Though the style of the ads (ranging from polite to objective to "in your face") will change according to the situation, and the points made will be tailored to each market, Apple will consistently deliver a small number of messages and will do so worldwide.

A bit later (we're into 1996 by now), Apple's advertising campaign will attempt more. Apple management wants to rebuild and strengthen the equity of the Apple name. (Pundits say that in the future, brand names will influence buying decisions more than product features—and the Apple name (and logo) is one of the most recognized in the entire world.) In addition, Apple will use both television and print ads to promote the Macintosh platform by emphasizing its high-level benefits and making direct comparisons to the competition.

I should mention that Apple will be doing many things in addition to advertising to promote the Macintosh platform. One important initiative that should be started by the time you read this is a "Top 300 Influencer Road Show," in which key Apple managers will argue Apple's case, not just to the usual press and analysts, but to others who can influence Apple's success—developers, distributors, and others.

The View From 1996

Here's what I think the competitive landscape will look like in 1996:

Windows 95 will be successful, but it will be seen as only part of Microsoft's OS strategy. Already, the June 5, 1995, issue of *InfoWorld has* reported that Microsoft is giving some corporate information-systems managers the option to migrate to Windows NT instead of Windows 95. (The article also indicates that some IS managers have evaluated Windows 95 and found it wanting.)

Also, in the June 12, 1995, issue of *Computer Reseller News,* Bill Gates was quoted as saying, "We will focus on [Windows] NT as our mainstream platform two to four years from now, when 16 megabytes [of memory] on the desktop is common." The article went on to say that "Gates' pronouncement corroborates speculation that Windows NT would evolve as Microsoft's long-term strategic desktop platform."

Apple's sales will continue to grow, bolstered by a line of PCI-based, PowerPC processor–based desktop and notebook computers being sold at aggressive prices. Apple will be selling Power Macintosh computers at a rate of over 1 million units per quarter, and a steadily rising percentage of Macintosh customers will be enjoying the benefits and added capabilities of Power Macintosh software.

The industry will begin to recognize what has already happened—namely, that Apple has changed to meet the demands of the market. It will notice an Apple that is more able to articulate its advantages in the marketplace, willing to license its platform to other vendors, and aggressive in its promotion of open, cross-platform industry standards.

Apple's advertising campaign will begin to kick in, and the message to customers will be "Macintosh is better than the competition, and you should stay with us if you want the next generation of computers in your hands today." In particular, customers will know that the most important advantages of the Macintosh platform are

- more powerful computers that let you work faster and try new things
- greater ease of use, so you can focus on the work, not the computer
- advanced multimedia, which makes computers more useful and more fun
- compatibility, so you don't have to give up anything to use a Macintosh computer

Apple will be focused on delivering this message in its advertising, press releases, briefings, and executive speeches—and it will be doing so worldwide.

Finally, underneath these efforts to make customers want to buy into the Macintosh platform will be a final message: *Apple's role is to continue the leadership of personal computing in the next decade.*

And Apple's message to you is, "We're doing it—so stay with us." ♣

Apple News

Power Macintosh Computers

continued from page 1

Two of the new systems—the new Power Macintosh 8500/120 and 7500/100 computers—also employ a new architecture first used in the Power Macintosh 9500 computer, that allows them to be upgraded with faster PowerPC processors, as they become available. The systems' main processors and clock chips are placed on an expansion board, which can easily be swapped with expansion boards that incorporate new, faster PowerPC chips. Power Macintosh 7200 systems can also be upgraded to Power Macintosh 7500-level performance with a logic board upgrade, which is expected to be released at a later date.

All the systems ship with an upgraded version of the Mac OS that includes the first release of Open Transport networking software, while the Power Macintosh 7500 and 8500 include PlainTalk 1.4, QuickDraw 3D, and a QuickTime Conferencing application.

Details about each of the new systems and the software that ships with them follow.

Power Macintosh 7200/75 and 7200/90

Combining high performance with a low price, the Power Macintosh 7200 series is being targeted toward small and medium-sized business and home office customers. Power Macintosh 7200 computers use the PowerPC 601 RISC processor running at 75 MHz or 90 MHz. In a recent independent study by Competitive Assessment Services, the Power Macintosh 7200 systems outperformed Windows-based computers with equivalent clock-speed Pentium processors by 24 percent overall. The study compared a Power Macintosh 7200/90 system to a Dell XPS 590 PC and the Power Macintosh 7200/75 to a Dell Dimensions XPS P75 PC.

Power Macintosh 7200/75 and 7200/90 computers ship with 8 MB of memory, a built-in 1.4 MB Apple SuperDrive floppy disk drive, an internal 500 MB hard disk drive, and an internal quadruple-speed CD-ROM drive. They also include built-in Ethernet, DRAM expansion up to 256 MB, and built-in graphics acceleration and 16-bit stereo input and output. The Power Macintosh 7200/75 is priced starting at $1,699 (U.S. only), while the Power Macintosh 7200/90 is priced starting at $1,899 (U.S.). Customers outside the U.S. should consult their local Apple dealers for pricing.

Power Macintosh 7500/100

The Power Macintosh 7500/100 computer, aimed at customers in larger businesses, is more powerful and expandable than Power Macintosh 7200 systems. Utilizing the 100-MHz PowerPC 601 processor, customers can upgrade Power Macintosh 7500 systems to the PowerPC 604 processor by changing expansion boards. When running certain graphics and publishing applications, the Power Macintosh 7500/100 performed up to 71 percent faster than a Windows-based computer with a 100 MHz Intel Pentium processor, according to a CAS study, which compared the Power Macintosh 7500 to a Gateway P5-100XL PC.

The Power Macintosh 7500/100 features built-in Ethernet, video input, high-resolution graphics, and CD-quality stereo; it can be used for videoconferencing with the addition of a video camera. Power Macintosh 7500/100 systems ship with 16 MB of memory, a built-in 1.4 MB Apple SuperDrive floppy disk drive, an internal 500 MB or 1 GB hard disk drive, an internal quadruple-speed CD-ROM drive, and a PlainTalk microphone. With a 500 MB hard disk, the Power Macintosh 7500 starts at $2,699 (U.S.); with a 1 GB hard drive, the system starts at $2,999 (U.S). Customers outside the U.S. should consult their local Apple dealers for pricing.

Power Macintosh 8500/120

Targeted toward media authoring, in-house publishing, and technical markets, the Power Macintosh 8500 computer uses the PowerPC 604 processor running at 120 MHz. In the CAS performance study, the Power Macintosh 8500 system performed 44 percent faster overall than a Dell Optiplex XMT 5/120 Windows-based PC using a 120-MHz Intel Pentium processor, and up to 80 percent faster when running certain graphics and publishing applications. With three PCI slots, DRAM expansion up to 512 MB, and the ability to be upgraded to use faster PowerPC chips when they're available, the Power Macintosh 8500/120 computer combines high-end PowerPC technology with system flexibility and expandability.

The Power Macintosh 8500/120 computer also features built-in Ethernet and video input and output capable of near-broadcast-quality high-resolution video as well as CD-quality stereo sound. It also supports real-time capture of quarter-screen video input.

Power Macintosh 8500/120 systems come with 16 MB of memory, a built-in 1.4 MB Apple SuperDrive floppy disk drive, an internal 1 GB or 2 GB hard disk drive, an internal quadruple-speed CD-ROM drive, and a PlainTalk microphone. With a 1 GB hard disk, the Power Macintosh 8500 is priced starting at $3,999 (U.S.); with a 2 GB hard drive, the system is priced starting at $4,499. Customers outside the U.S. should consult their local Apple dealers for pricing.

Software Details

All of the new Power Macintosh systems will ship with Macintosh System 7.5.2. This release is optimized to take further advantage of the PowerPC processors and the new Power Macintosh architecture. Included in System 7.5.2 is Apple's Open Transport networking and communications architecture. Open Transport comes bundled with AppleTalk and TCP/IP protocols, allowing simplified access to the Internet and virtually all Macintosh and PC networks. (For more information on Open Transport, see "Unified Networking for Mac OS Computers" on page 18 of this issue.)

The new Power Macintosh systems run virtually all existing Macintosh applications as well as "native" applications that have been optimized for PowerPC technology. Currently, over 820 native applications are shipping. As with all Power Macintosh models, the new Power Macintosh systems read Macintosh, Windows, MS-DOS, OS/2, and ProDOS floppy disk formats. The new systems can also run DOS and Windows-based applications using Insignia Solutions' SoftWindows program.

All the new systems ship with Megaphone by Cypress Research, a telecommunications application that utilizes the GeoPort Telecom Adapter's 14.4 K baud modem to provide full-duplex speakerphone, answering machine, and fax send and receive capabilities.

The Power Macintosh 7500 and 8500 systems also include PlainTalk 1.4 text-to-speech software and the QuickTime Conferencing application, Apple Media Conference, which lets customers communicate with voice, video, and data without having to leave their desks. Apple Media Conference works out of the box with the PlainTalk microphone included with Power Macintosh 7500 and 8500 systems and a user-supplied NTSC analog output camera.

Additionally, Power Macintosh 7500 and 8500 computers ship with QuickDraw 3D, a 3D-graphics extension to the Mac OS. QuickDraw 3D software lets users create, manipulate, and incorporate 3D graphics into their documents and their presentations as easily as they do with text and 2D graphics today.

In With eWorld, Out With AppleLink

Life just got better: Your AppleLink service is being cut off!

How can that be true? Apple Computer, Inc., is making its long-awaited transition from AppleLink to eWorld, a transition that will affect the entire Apple developer community—positively, we hope. Apple expects all AppleLink services to be closed down by year-end 1995, when it expects most, if not all, AppleLink subscribers to be using eWorld.

You need not have an eWorld account to remain a member of Apple's developer programs or to stay in touch with Apple, but we think you'll want one. eWorld gives you all the services AppleLink has and adds some very useful features, such as Internet access and access to areas with special developer content. And eWorld is cheaper than AppleLink, wherever you are in the world.

Here's what eWorld will provide:

- *E-mail.* eWorld provides secure electronic mail to the entire Apple community, including the expanding base of eWorld subscribers. Although there are no guarantees, it's even likely that you'll be able to use the same address you've used all these years on AppleLink. While AppleLink remains in service, you'll also be able to send mail between eWorld and AppleLink accounts.
- *Internet access.* Through its Internet On-Ramp, eWorld provides you access to the World Wide Web, FTP sites, newsgroups, and Internet mailing lists.
- *Developer Corner in the Computer Center.* Here's where you can find the content you're used to finding in AppleLink's Developer Support area, such as the online version of *Apple Directions,* in addition to a great deal more. The eWorld Macintosh Development Forum replaces the Developer Talk discussion board. You'll also find program information, APDA, Developer University, and most of the content from the Developer CD Series.
- *AppleCentral, for developer program members only.* Here you'll find discussion groups about emerging Apple technologies, Apple software, and information vital to members of Apple developer programs.
- *Widely expanded content.* In addition to everything you're used to reading on AppleLink, you'll have access to a broad variety of news and information over eWorld through the Computer Center, Business & Finance Plaza, Newsstand (with news and sports), Arts and Leisure Pavilion, and the other locations in the eWorld virtual community.

You'll receive all these services for significantly less than you're used to paying for AppleLink. The savings will be 50 percent or more, depending on the country you reside in and network charges. There are no additional charges for prime-time access, high-speed access, or downloading kilocharacters.

What will you have to do to make the transition? Actually, very little. In the next few months, Apple will make eWorld 1.1 available to all members of Apple developer programs as well as all subscribers to the Apple Developer Mailing. If you're a program member, you'll receive access to AppleCentral, which contains private Apple developer-only content.

The software package will contain registration instructions and all of the details on how to install and use eWorld. If you have an AppleLink account, it will tell you how to close it down. If you already have an eWorld account, you don't have to do a thing; keep using it, and wait for further details about how you can get access to AppleCentral, if you're a member of Apple's developer programs.

So, as we said, AppleLink will soon be replaced by eWorld, which we think will make life better for you in many ways. Look for your eWorld software soon; in the meantime, if you have questions about the transition, send an AppleLink message to DEVSUPPORT or e-mail to devsupport@applelink.apple.com if you're in the United States or Canada, or contact your local Apple office from elsewhere in the world.

Apple Ships OpenDoc Developer Release 3

Included in this month's developer mailing is OpenDoc Developer Release 3 (DR3), which adds stabilization to the feature-complete version of the OpenDoc component software toolkit shipped in May 1995.

The latest version of the OpenDoc Developer Release CD includes a pre-release version of the OpenDoc Development Framework (ODF), an object-oriented framework developed in C++ and targeted for building cross-platform OpenDoc component editors. ODF simplifies the process of building an OpenDoc component editor by implementing much of a component editor's default behavior. Additionally, OpenDoc DR3 provides supporting material to enable the development of OpenDoc components with Metrowerks 6.0.

Also part of OpenDoc DR3 is a prerelease version of Novell's AppWare, which you can use to produce OpenDoc components. In addition, OpenDoc DR3 for Macintosh includes an improved version of the Container Application Library (CALib), which offers you an easy way to add OpenDoc container support to existing applications.

If you have yet to receive your OpenDoc DR3 CD, send an e-mail message to opendoc@apple.com or complete the request form available from Apple's new OpenDoc home page on the World Wide Web at http://www.info.apple.com/opendoc/.

New Newton Toolkit Offers Compiler, Reduced Prices

Apple Computer, Inc., recently announced the availability of Newton Toolkit version 1.5 and Newton Book Maker version 1.1. The development environment Newton Toolkit (NTK) 1.5, available for Mac OS–based computers, features a new NewtonScript compiler and profiler, which enable you to design and optimize Newton applications. NTK 1.5 provides a range of user interface improvements designed to make NTK simpler, more intuitive, and easier to use than previous versions. NTK 1.5 also features greater support for developing software localized for multiple markets worldwide.

NTK, shipping since August 1993, is a high-level object-oriented development system that combines rapid prototyping and interface building, a library of extensible components, and an interactive development model, enabling developers to see the results of their work in minutes instead of weeks. These features make it possible for you to bring Newton products to market much more quickly than you could on other PDA (personal digital assistant) platforms.

Newton Toolkit 1.5 Features

NTK 1.5 is designed to address some of the key areas of customer feedback Apple has received about NTK since it first shipped in August 1993. The most important changes are the new NewtonScript compiler and code profiler, which enable developers to create much higher-performance Newton applications.

The NewtonScript compiler will enable you to compile specific routines in your application to native ARM code, enabling them to run up to 80 times faster than byte-code interpreted code. However, you should not compile your entire Newton application, for the following reasons:

- Not all code routines will see the maximum performance improvement, or even an improvement at all. Routines that interact heavily with the Newton operating system, particularly the view system, will not benefit from being compiled.
- Jumping back and forth too often between compiled and interpreted NewtonScript will also decrease performance.
- Compiled routines take eight to ten times more space in memory compared with interpreted routines, and most Newton devices have a very limited memory capacity.

For these reasons, NTK 1.5 also includes a new profiler that will allow you to measure the performance of specific NewtonScript routines both before and after they have been compiled. Using the NTK profiler, you can determine which code routines to compile.

In addition, NTK 1.5 includes a number of human interface changes, which will allow NTK developers to be more productive. The changes include

- fewer icons on the proto palette
- cleaned-up preference and settings dialog boxes
- ability to open a browser with a double-click
- ability to open a slot editor with a single-click
- error messages that now automatically take you to the offending line of code
- multiple text files with user-controlled build order

Newton Book Maker Features

Newton Book Maker 1.1, now sold separately from the Newton Toolkit, is a tool that allows people to create electronic books, travel guides, reference manuals, and corporate documentation for Newton-based devices. People without any programming expertise can use Book Maker to create sophisticated and easy-to-navigate applications from electronic documents. Newton programmers can use Newton Toolkit objects and the NewtonScript language to add additional features to Book Maker applications. Newton Book Maker 1.1 features several minor enhancements and bug fixes.

Future Plans for NTK

The next version of Newton Toolkit will feature support for including C or C++ routines in a NewtonScript application. The Newton C++ tools will have similar performance constraints to compiled NewtonScript: Code size will be significantly larger in memory, and performance will depend on the level of interaction with the Newton operating system and other interpreted NewtonScript code. However, Apple feels that these tools will give developers a greater level of flexibility when porting existing code from other applications, or when writing routines that they know they will want to compile for performance reasons.

NTK engineers are also hard at work on a Windows-hosted version of Newton Toolkit. This product will match the functionality of NTK on the Macintosh, but will be hosted on a Windows-based computer and feature a full Windows look and feel. Apple is strongly committed to ensuring that developers on both the Mac OS and Windows platforms are able to develop Newton applications.

Further in the future, NTK engineers are also working on porting Newton Toolkit to run “native” on Power Macintosh computers, on improving the debugging support (including source-level debugging), and on providing support for project management and version control.

If you have ideas or feedback on the NTK and Book Maker tools, including specific feature requests, please send it to the following Internet address: tools@newton.apple.com.

Pricing and Availability

With Newton Toolkit 1.5, Apple is establishing a new pricing structure designed to make Newton tools accessible to more developers. NTK 1.5 is priced at $299, and Newton Book Maker 1.1 is $199. (All prices are for the United States only; international customers should contact their local APDA office.) NTK 1.5 Update is available for $99 to customers who already own NTK, and it includes updated versions of both NTK and Newton Book Maker. All three products are currently available from APDA (see page 36 for ordering information).

Apple to Offer Oracle Power Objects for Client/Server Development

Delivering on its commitment to cross-platform and client/server computing, Apple Computer, Inc., recently announced the addition of the new Oracle Power Objects product line to its own suite of tools for application developers. Oracle Power Objects, developed by Oracle Corporation and licensed for distribution by Apple, is an object-oriented client/server development tool.

Apple plans to make Oracle Power Objects available in both Mac OS and Windows versions upon its release this month. It will be sold wherever customers currently purchase Apple or Claris software, including retail stores, mail-order companies, and APDA. A version supporting IBM’s OS/2 operating system is expected from Oracle next year.

Visual, Drag-and-Drop Programming

Oracle Power Objects is a visual development tool, meaning that most of the development effort consists of dragging and dropping onscreen objects that represent application components, such as database tables. These objects can be easily created, modified, and reused, a key benefit of object-oriented programming. Users who have no in-depth knowledge of relational database technology or the SQL programming language can use Oracle Power Objects to rapidly build sophisticated client/server applications.

Furthermore, the applications created by Oracle Power Objects are compiled into a single executable module, which can be distributed to others without any kind of licensing fee. By eliminating the need to distribute a runtime module and additional support files (as with other visual-programming solutions), Oracle Power Objects improves application performance and makes the distribution and maintenance of solutions easier.

Superior to Visual Basic

Oracle Power Objects supports the OLE Custom Extension (OCX) architecture developed by Microsoft for building reusable application components—called *custom controls*—for use in Visual Basic applications. Oracle Power Objects customers on the Windows platform will be able to use Visual Basic custom controls. Since Oracle Power Objects is cross-platform, custom controls that do not use Windows-specific calls will also work on the Mac OS and OS/2 versions of Oracle Power Objects.

Oracle Power Objects includes the Oracle Basic scripting language, which is used to add procedural functionality to the application. Oracle Basic is compatible with the BASIC syntax found in Microsoft's Visual Basic, to the point that customers familiar with that product should be able to directly apply their expertise to Oracle Power Objects. And the applications they create will, unlike Visual Basic applications, support both the Mac OS and Windows platforms.

Another advantage of Oracle Power Objects over Microsoft Visual Basic is that the latter contains no built-in support for database access and manipulation, whereas Oracle Power Objects provides these functions through onscreen objects that are added to an application by drag-and-drop manipulation. In addition, Visual Basic contains no built-in database connectivity, whereas Oracle Power Objects includes support for several major database management systems (DBMS).

Configurations and Pricing

The Oracle Power Objects product line consists of two unique products, each available for both the Mac OS and Windows operating systems. The Oracle Power Objects Standalone Edition version 1.0 is designed for prototyping client/server applications without the need to connect to host-based data, or for delivering applications that run on a single computer with a local database. The product includes a forms designer; report writer; database session manager; query-by-form tool; Oracle Basic editor, debugger and compiler; and a complete stand-alone local DBMS that is upwardly compatible with the Oracle DBMS. The DBMS can be distributed with stand-alone applications, or it can be used to help prototype client/server applications that utilize server-based data when deployed. The stand-alone version also works with Personal Oracle 7, Oracle's single-user DBMS. Suggested retail price for this product (in the United States) is $395; all other customers should contact their local Apple dealer for pricing.

Oracle Power Objects Client/Server Edition version 1.0 is a complete system for delivering full-scale client/server applications. It includes all the features of the Standalone Edition detailed above, plus connectivity to Oracle, Sybase, and Microsoft SQL servers, allowing full-scale deployment of applications accessing host-based data servers. Suggested retail price for this product (in the United States) is $1,995; all other customers should contact their local Apple dealer for pricing.

Oracle Power Objects for Macintosh requires a Macintosh, Power Macintosh, or other Mac OS–based computer with a 68020 or faster processor, Macintosh system software version 7.5 or later, at least 8 MB of memory, and at least 10 MB of hard disk space. The Windows version requires an IBM or Compaq-compatible computer with a 386 or faster processor, Microsoft Windows version 3.1 or later, at least 8 MB of memory, and at least 10 MB of hard disk space.

Future Plans for Native Version, OpenDoc Support

The next version of Oracle Power Objects Client/Server Edition is expected to add support for Microsoft's Open Database Connectivity protocol (ODBC), which will expand the scope of supported databases to include most relational and legacy database management systems.

The next versions of both Oracle Power Objects Client/Server and Standalone Editions are slated to add support for IBM's OS/2 operating system. Additionally, two key Apple technologies are expected to be supported as well. With OpenDoc container support, developers and users will be able to use OpenDoc components by simply dragging them into their Oracle Power Objects applications. Also added in the same release will be native-mode support for Power Macintosh computers.

In addition, Oracle plans to encourage the development of Oracle Power Objects objects and OCX custom controls by establishing the Oracle Object Marketplace, a World Wide Web site where customers will be able to preview components before purchasing them.

Product Positioning

Oracle Power Objects is a product that was designed specifically to help customers without extensive programming experience create custom client/server applications; a future version of Oracle Power Objects will provide container support for OpenDoc parts.

In contrast, the recently announced technology code-named *Denali* (see page 10 of the July 1995 issue of *Apple Directions* for details) is based entirely on the OpenDoc component software architecture. It is meant to be a general-purpose application development environment that will be used by a wide range of developers.

Product Support

Apple plans to provide installation and configuration support for its Oracle Power Objects products for the first 30 days following purchase of the product. Additional support options will be available directly from Oracle Corporation.

Apple Internet Connection Kit Gives Macintosh Users Internet Access

Among the new products Apple Computer, Inc., announced at Macworld Boston was the Apple Internet Connection Kit, which gives Macintosh customers an easy way to gain access to the Internet. Apple now provides Macintosh customers with two simple ways of using the Internet: The less experienced can access the Internet through eWorld, Apple's online service, while more experienced users can choose the Apple Internet

Connection Kit for direct access to the Internet.

The kit combines Apple software with Internet applications from other companies. It includes the following software:

- Netscape Navigator, the popular Internet browser
- Claris's new Emailer Lite electronic mail software
- the Apple Internet Dialer, which lets Macintosh customers quickly register with a qualified Internet service provider for their Internet connection
- Fetch FTP software, which lets users download files using File Transfer Protocol (FTP) services
- Stuffit Expander decompression software from Aladdin
- NewsWatcher browser software, which gives users access to Usenet newsgroups
- PPP and SLIP to allow the use of MacTCP over a modem line
- NCSA Telnet terminal emulation software, which lets Macintosh users connect to UNIX-based Internet libraries and catalogs
- QuickTime VR Player, which allows users to view 3D multimedia presentations and QuickTime video clips downloaded from the Internet
- Adobe Acrobat Reader, which lets users view portable document format (PDF) files in their exact original format

To provide support while users set up their Internet connections and use the other software, the Apple Internet Connection Kit includes Apple Guide online help for all its components. Also, customers can combine other Internet tools with the kit and be assured of full compatibility.

The Apple Internet Connection Kit requires a Macintosh, Power Macintosh, PowerBook, or other Mac OS–based computer; at least 8 MB of memory; Macintosh system software version 7.5 or later; a 9,600-bps (or faster) modem (14,400 recommended), or a connection to a local area network with an existing Internet connection; a minimum of 10 MB of available hard disk space (15 MB recommended); and a high-density floppy disk drive or CD-ROM drive. The suggested U.S. retail price for the Apple Internet Connection Kit is $59; pricing outside the United States will vary. The product is expected to be available in most locations by September..

Apple Is Committed to Gaining Market Share

Dan Eilers, senior vice president of Worldwide Marketing and Customer Solutions, sent the following letter to Apple employees July 13, 1995. We share it with you, unedited, to help clarify Apple's current market share strategy.

I had to be peeled off the ceiling when I read the front page article in the July 10 issue of *MacWEEK* by Jon Swartz and Robert Hess entitled "Apple: Market share no longer top priority" (*MacWEEK*, July 10, pages 1 and 92). As you all know, the headline is entirely misleading and just plain wrong and I want to take this opportunity to go on the record about our market share strategy to clear up any confusion this article may have caused.

Of course it is *true* that we want to sell more products, etc. to our installed base. *But* it is just as true that we want to capture more customers for Apple in our chosen segments (education, both higher ed and K–12, home, small business, entertainment, large business and government). That's why I've got market share graphs hanging on my wall—to keep close track of our market share movements by segment. I talk about our aggressive plans to gain share within these segments in everything I do, from employee meetings to speeches to presentations to interviews.

Our share position in each clearly delineated market segment will determine our future. We are aiming for leading share in each area in which we compete. That is the whole point of our strategy. Let me be very clear here. If we grow share in these fast growing core market segments, we will grow share overall and that remains our goal.

Additionally, we are looking to licensees to be a source of new customers for our platform. *But* clearly we are looking to entice *new* customers to Apple products and services.

Finally, while we do intend to be much more aggressive in the way we tell our story, and how we offer a better solution to customers, it is not our intention to "slam" the competition . . . but merely to assure that our points of value difference are more clear to our customers.

Dan Eilers
Senior Vice President
Worldwide Marketing
and Customer Solutions

Apple Enterprise Awards Go to Nine Developers at PC EXPO

Nine developers and their custom Mac OS applications were named winners of the first annual Apple Enterprise Awards during a ceremony held at PC EXPO in June. The Apple Enterprise Awards, sponsored by Apple Computer, Inc., and Computerworld, Inc., and co-sponsored by MacIS, recognizes custom, large-scale solutions implemented on the Mac OS platform. One winner from each of nine categories was presented with the Apple Enterprise Awards Trophy by Apple President and CEO Michael Spindler and Gary Beach, CEO of Computerworld.

A broad range of organizations submitted entries, including Fortune 500 companies, research and educational facilities, and a variety of federal, state, and local government agencies. Evaluation of those entries was conducted by eighteen leading analysts, scientists, journalists and educators. Here are the winners, listed by award:

- Client/Server—Department or Workgroup: Alex, Brown & Sons Incorporated—Alex, Brown Trading System (ABS Trading System).
- Client/Server—Education and Government: University College Dublin—WEST (Web Educational Support Tools) 1.0.
- Client/Server—Enterprise: Control Data Systems, Inc.—Control Data Advisor.
- Communication and Collaboration: Bergen Brunswig Corporation—AccuSource.
- Cross-Platform: The Naval Air Systems Command (Information Management Department)—Naval Aviation Wide Area Network.
- Mobile Computing: Centric Systems Corporation—The Mortgage Market.
- Publishing: Second Look Computing of the University of Iowa—Arachnid 1.0.
- Multimedia/Kiosk—Business: Andersen Windows, Inc.—The Andersen Window of Knowledge System.
- Multimedia/Kiosk—Education and Government: Penn State University—Penn State Alumni Association Information Kiosk 1.0. ♣

Technology

Inside This Section

develop Issue 23: Music, 3D, and More

Would you like to know how your application can easily play some music, build 3D geometries, or implement multiple-pane dialog boxes? You'll find all this and much more in Issue 23 of *develop,* Apple's award-winning technical journal.

- "Music the Easy Way: The QuickTime Music Architecture" explains how to add music to your application without having to learn the intricacies of MIDI or sound production.
- "The Basics of QuickDraw 3D Geometries" shows how the geometries in QuickDraw 3D fit in with the rest of the system and how to make good use of them.
- "Implementing Shared Internet Preferences With Internet Config" examines a shared preferences solution for Internet applications, showing how to use it in your applications and also how it works, using the Component Manager, as a robust shared library mechanism.
- "Multipane Dialogs" describes a method of implementing dialog boxes with multiple panes, using a scrolling list of icons to select panes.
- "Document Synchronization and Other Human Interface Issues" explains how to make window titles in your application match

please turn to page 24

CD Highlights

Reference Library Edition, September 1995

Featured this month: everything in Adobe™ Acrobat! We've finally got almost everything converted, indexed, and all together in a big happy pile for you. Thanks to Acrobat's compression, we can finally include all of the core technical documentation on the Reference Library Edition, rescuing the AOCE, QuickDraw GX, and human interface books from their lonely exile of past months.

Reference Library Edition

Also, since it's much easier to build documents in Acrobat than in Apple DocViewer, most everything has been converted and thus is included in the disc-wide index for your Verity relevancy-ranked searching pleasure. Be sure to upgrade to the latest version of Acrobat Exchange LE (available on this disc, in the Utilities folder) and check out the Read Me First! file for various Acrobat tips 'n' tricks. We've only just begun to figure out what we're doing with Acrobat, and a lot of polishing remains to be done; if you have ideas or suggestions for improved documents, bookmarks, or hyperlinks, send them to us at Internet address dpfeedback@applelink.apple.com or AppleLink address DPFEEDBACK.

So, along with six updated Technical Notes, a whole folder full of new Macintosh Technical Q&As, and even a new Development Kit or two, here's what's new and revised on this month's disc.

AEGizmos

AEGizmos consists of four libraries that provide alternate ways of building and reading Apple events and Apple event descriptors. They're faster, more memory-efficient, and, often, easier to use than the normal AE API. Version 1.4.1 is now compatible with CodeWarrior 6 and includes a nonshared library for PowerPC.

Asia Market Guide

This guide provides information about markets, distribution, and localization for the Asian region, as well as a directory of additional information sources.

Developer Notes

Included here, in addition to the regular archive, are developer notes for three new products: The Power Macintosh 7200, 7500, and 8500 computers, just announced at Macworld. (See news story on page 1.)

Developer Univ. Course Info

This package provides information on the types, cost, and location of training available to Apple's developers through Apple's Developer University. It enables you to determine which training will be appropriate to your needs and how to obtain it. It includes current course descriptions, class dates and locations, and where to get self-paced materials.

please turn to page 25

Human Interface

Easy to Use

By Peter Bickford

Doc Goes Back to Basics to Talk About the Core Principles of Building Usable Software…

I could have bought a round-trip ticket to London. Or a flatbed scanner. I could have bought new furniture for the living room, a really great television, or that mixing board I'd been wanting for my music studio. Instead, it was 11:30 at night and I was sitting upstairs at the Linotronic shop, and my head was slumped down on the desk while my job got run—again—on the big imagesetters. My day had begun at 6:00 A.M. when I learned that my expensive, full-color box job was ready for press check. Great news, except that the press operator had noticed that something was missing: the color blue. This might have made for an interesting artistic effect, except for one thing: The name of the product was in blue. No blue, no box.

What had happened? Good old fashioned "user error." Or rather, user *interface* error. In preparing the piece for print, I'd decided to check the colors one last time. However, my reselecting the exact blue I wanted from the color library had an unexpected side effect: It quietly and automatically moved a radio button from "Process color" to "Spot color." I didn't notice the change then, and it didn't show up on a quick proof print. But it sure as heck showed up on the press. So there I was after a long day of scrambling, obsessing about all the things I could have bought for the $700 this little glitch had cost me in extra press charges and wasted film.

A Failure to Communicate

The page-layout program I was using comes in a box that hails it as "easy to use." The computer I was using it on is also supposed to be easy to use. Come to think of it, virtually every computer I've ever used claimed to be easy to use, whether it ran DOS, Windows, or the Mac OS. Nevertheless, like most users, I've spent hundreds or thousands of hours learning the programs, working around their errors and idiosyncrasies, configuring and maintaining them, and doing all manner of chores that have nothing to do with getting my work done. To paraphrase Inigo Montoya from *The Princess Bride,* "This word *usability* you keep saying . . . I do not think it means what you think it means." Or, in the words of the sadistic warden from *Cool Hand Luke,* "What we've got heah is a Failure to Communicate."

No particular computer or operating system has a monopoly on usability. Usability has little to do with icons or flashy graphics. It starts with the goal of simply letting users get their work done without spending a lot of time worrying about *working the computer.* Ideally it then goes a step further to actually help users work better and faster than they could otherwise.

Making It Easy—The Short Course

Good, usable software relies on certain basic principles. These apply no matter whether you're working on a Sun SPARCstation or an Apple Power Macintosh computer. The reason is that most of these principles work on the human abilities and psychology rather than the conventions of one platform or another. The following is a list of ten basic principles that drove the design of Apple's Macintosh. Similar lists appear in the user interface development guidelines for Windows, OS/2, and other major user interfaces.

1. *Consistency.* Things that work one way in one part of the system should work the same way in other parts. This allows users to learn something once, then apply that knowledge again and again as they use the computer.

   Just as important as consistency within a program is consistency between programs. No matter how important your program is, it won't be the only one that your customers are using. If every other program uses a certain key combination to trigger the Save command, it's not a good idea to use that combination to mean Send in your mail program. Similar caveats apply to the use of such standard interface elements as menus and windows. Even if your program might be made slightly better by making these behave in a nonstandard manner, it's guaranteed to drive your users crazy. If they have any kind of choice, it will also drive them to a competitor's product.

2. *Aesthetic integrity.* There's an old aphorism that says when a man wears a bad suit, people notice the suit. When he wears a good suit, they notice the man. The same goes for interface design. A good design is understated and lets the user concentrate on the information being presented. Bad designs use loud graphics, overly gray, chiseled backgrounds, and other faddish ornaments that are just there because the designer thought they looked cool.

   Some designers even go so far as to change the look of standard interface elements such as buttons and scroll bars. This really confuses users, since they assume that the change must have been done for a reason, and they'll actually intuit their own rules for how these revised elements differ from the standard ones. It seldom occurs to users that the only reason you made them look different is that you got bored with the regular ones.

3. *Perceived stability.* Even if your program thinks it knows what's best for the user, keep in mind that they're the ones in

control, and that no change in their environment should happen without their knowledge and permission.

This is the one that cost me the trip to London when the computer changed not only the parameter I asked for (the particular shade of blue), but also things I didn't ask it to change (whether I wanted the color to be printed using a spot color or a four-color method).

4. *See-and-point, not remember-and-type.* Computers are good at precisely remembering things like codes, command names, and lists of data. People are generally terrible at it. Instead of making users remember and type this sort of data, the computer should always give them a list of valid possibilities and let them choose from it. Not only will the users' anxiety level drop, but the programmers are spared having to handle all the error conditions that arise when users guess wrong.

5. *Direct manipulation.* Good graphical interfaces allow their users to feel as if they are directly controlling a little world inside the computer. Instead of abstracting out their work to a set of command words, they can just grab the things they want to work on using the mouse and interact with them directly. Want to delete a document? Drag it to the trash can icon. Want to move a file from one directory to another? Just grab the file icon and move it to the folder you want it in.

   This sort of direct manipulation is the real strength of graphical interfaces. Instead of memorizing commands and parameters, users are able to learn the behaviors of a few simple interface objects like folders or buttons, then apply that knowledge in different situations.

6. *Metaphors from the real world.* Sometimes it's possible to clue users in on how something in your computer world works by equating it to something they already know about from the real world. This isn't so much a general principle as it is a really good trick for making an interface understandable to newcomers. Most graphical interfaces, for instance, represent hierarchical disk directories as folders. The idea is that since users know how to use folders in the real world (you can put things in them, give them names, put them inside other folders, and so on), you'll have an idea of how to use them when you see them on a computer screen.

   Other metaphors in common use include the various brushes and tools in paint programs, "in boxes" for mail systems, and even the all-purpose trash can. Naturally, there are some limits to the use of metaphors, and they can't be used at all in command-line interfaces, but when they can be used properly, they can be very effective.

7. *WYSIWYG—What You See Is What You Get.* Documents onscreen should match what they'll look like when they're printed. In the old days, word processors often used command characters for font changes, as in ^ BThis is a bold phrase ^ b, which became **This is a bold phrase** when printed. While most software is now WYSIWYG when it comes to fonts, we still have a long way to go in terms of color matching, separations, and other fine points.

8. *Feedback and dialog.* Good programs never keep the user guessing. They react immediately when you perform an action, such as clicking a button. And if something is going to take a long time, the computer keeps you informed about not only what it's doing, but how long it's expected to take. The quality of feedback your program provides and the user's blood pressure level are directly related. If the user clicks on a button and your program doesn't respond in some way within half a second, the user starts getting nervous. One study showed that when the computer failed to visibly respond to a button click, it took just 8.5 seconds for half of the participants to assume the machine was hung and press the restart switch.

9. *Forgiveness.* Humans make mistakes. Good programs allow for this by letting them undo their last action, or even revert to a previous version of the document. If users are about to perform some potentially damaging action from which there is no going back, the computer should inform them of the danger and ask whether they want to proceed.

10. *User control.* No matter what, the user must be the one in control at all times. Nothing destroys a user's peace of mind faster than having the computer appear to be taking over the action. Want to generate a few thousand letters and dozens of negative columns? All it took was the rumor that Windows 95 contained a runaway agent that would sniff out and report all the software on a user's system and report it back to Microsoft. Think about this before you design any sort of system, agent, or wizard that attempts to put anyone other than the user in control.

It's Up to You

The ten principles above are just a start on what it takes to get truly usable software. What really needs to happen is that companies and consumers must start paying attention to the costs of hard-to-use software.

Companies might want to consider, for instance, that a study in the September 21, 1993 issue of *SoftLetter* concluded that the average technical support call carries a total cost of $23.33. Just one such call per customer could easily wipe out the profit the company makes from the entire sale. Put off fixing usability problems now, and you can be sure of paying for them later.

We need to do our part as customers, as well. We must realize that the cost we pay for a computer or software program only begins with the price rung up at the register. The real cost can be tenfold in terms of time and energy (and, sometimes, in wasted Linotronic film). We need to demand more of the companies that produce these products, and hold them to task if they don't deliver. "Easy to use" has to stop being a buzz-phrase and start being a fundamental quality of any product we would consider buying.

Till next time,
Doc

Peter Bickford is a member of the Apple Computer, Inc., Human Interface Design Center. He can be reached by AppleLink at THE.DOKTOR.

Unified Networking for Mac OS Computers

Open Transport Makes Networking Easier for Developers, Users, and Network Managers

By David Gleason

There is no more vexing problem in the computer industry today than that of network complexity. Over the past several years, the proliferation of networking protocols and the rapid expansion of desktop systems in businesses have contributed to a networking reality that is increasingly difficult to manage.

Open Transport is Apple's solution to the problems of network complexity. It provides built-in protocols and basic networking services, as well as an extensible foundation for developing networking products. In this article, I explain why Apple Computer, Inc., has developed Open Transport, provide an overview of the Open Transport architecture and how it fits in with Apple's plans for the future of the Mac OS, and then take a look at the benefits it provides for users, network administrators, and developers. Finally, I'll itemize some developer opportunities that Open Transport makes possible.

If you develop networking applications, networking cards, or networking protocols, Open Transport should be of significant interest to you. Through a new set of application programming interfaces (APIs), Open Transport provides standards-based interfaces that you can write to, making it easier to provide networking solutions for Mac OS users and others:

- If you write network-based software products, Open Transport permits you to write to an industry standard API and develop applications that work regardless of the user's underlying network. This will allow you to sell to a larger audience and develop to only one API instead of several.
- If you develop networking cards, the Open Transport hardware-level API makes it much easier to create products that can be ported easily across platforms.
- If you create networking protocols, Open Transport lets you write to a standard networking API, so that your protocols will run on a variety of platforms using the same routines.

Open Transport is a complete rewrite of Mac OS networking, including entirely new versions of AppleTalk, MacTCP, and serial communications. It is an integral part of Apple's plans to bring the Mac OS platform to a new level of performance, ease of use, and robustness, and it promises to provide significant improvements in networking for users, network administrators, and developers alike.

Keep in mind that Open Transport will not be all things to all people. It has, however, been designed to provide broad backward compatibility for existing networking applications, while introducing new levels of performance and ease of use for networking services.

Recent articles in *Apple Directions* have explained the strategy behind Apple's forthcoming system software release, code-named *Copland.* (For details, see the article "Copland: Technology for Customers' Sakes" in the June 1995 issue of *Apple Directions*). Open Transport will become an integral part of Copland, and it is important for you to understand the role that Open Transport will play in the Mac OS in the future.

Open Transport Is Available Now

The first implementation of Open Transport, for the Power Macintosh 9500, appeared in June. The new Power Macintosh computers just introduced at Macworld (see the story on page 1) include Open Transport 1.0.6, and a version for other PowerPC processor and 680x0 Mac OS computers will be available in the future. Open Transport version 1.1 is expected to be announced during the fourth calendar quarter of 1995 and, soon thereafter, will ship with all new Macintosh computers.

Existing pre–Open Transport networking applications will continue to work for some time, but the "backward compatibility" modules will disappear in the Copland time frame, and networking applications will be required to adopt Open Transport.

There are two distinct advantages to adopting Open Transport now: first, the advantage of being an early adopter and reaching new markets; and second, the advantage of having already integrated Open Transport into your product when, later, you have to make the rest of the transition to Copland.

Why We Need Open Transport

You don't need to be a networking expert to know that we now live in what is euphemistically called a "multiprotocol world."

The essential networking services required by most users a few years ago, such as printing and file transfer, were provided on Macintosh computers by Apple's proprietary AppleTalk, the set of networking protocols that comes with every Macintosh computer and LaserWriter printer. Today, however, user needs have expanded to include the complexities of mixed computer environments, high-speed data transfers, multiple network addresses and configurations, and more. For example, with the exponential growth of the Internet, TCP/IP protocols are now considered essential not only for businesses and institutions but even for many home computer users.

Mixed environments of personal computers, minicomputers, workstations, and mainframes in large corporations have created a jumble of networking solutions that require extensive maintenance and expertise. Since each protocol and platform has its own technical requirements, programming interfaces, tools, and applications, this complexity of solutions leads to administrative headaches and rising computer costs.

To save time and overhead, computer administration departments are often forced to impose configuration limitations and purchasing restrictions on their users. Users—who are usually looking for a solution, not a particular protocol—wind up with fewer product choices and may be forced to adopt department-imposed solutions that do not work for them.

Perhaps the problem could be solved by a single, unified, dominant networking protocol; but the fact is that the track record of the computing industry suggests that universal agreement on a single standard is unlikely. And this lack of a single standard often forces everyone to make unsatisfactory choices—users suffer from limited options for networks and applications, network managers must fight for the use of a single protocol that can be supported in a cost-effective manner, and developers must face difficult choices in choosing which protocols and APIs to develop for. If you focus on a single protocol on which to develop networking products, this shuts you out of other markets; on the other hand, multiprotocol solutions generally require you to create multiple versions of your applications and hardware products, which is complex, expensive, and time-consuming.

Apple believes that the solution lies in acknowledging that we now live in a multiprotocol world and that the best solution is one that allows the operating system to work with multiple protocols that already exist. This takes the burden of decision off the shoulders of the user, the network manager, and the application developer, and offers greater satisfaction to all three. And this solution is the rationale behind the development of Open Transport.

What's most important, however, is that developers can begin integrating the features of Open Transport right away, and can be working with the Open Transport APIs today. By getting started with Open Transport now, you as a developer can enjoy the benefits that are already available.

The Benefits of Open Transport

Open Transport is not a new networking protocol; rather, it is an architecture that leverages off existing industry standard protocols and makes it easier for Mac OS computers to "fit in" with networks as they exist today. But, in addition, Open Transport will provide users with networking that is faster, more powerful, and easier to use than today's AppleTalk and MacTCP—while still being compatible with networking applications that depend on those protocols.

Later in this article, I'll explain the benefits of Open Transport to developers, users, and network managers. But first, let's take a look at the Open Transport architecture.

The Open Transport Architecture

Open Transport provides three standards for developers: the X/Open Transport Interface (XTI) from the X/Open Company Ltd., the Data Link Provider Interface (DLPI) set of APIs, also from the X/Open Company and STREAMS from UNIX® System V. These interfaces permit commercial and in-house developers to write to standard APIs that are widely used throughout the computer industry.

The figure "The Open Transport Architecture" (page 20) shows how both existing and Open Transport–aware applications work with Open Transport. Existing applications make the same calls they always have and, without their knowledge, hook into an existing protocol (either AppleTalk or TCP/IP) through a backward compatibility layer that calls the appropriate XTI routines. Note how Open Transport provides standard APIs for different kinds of development:

- XTI for application developers
- STREAMS for networking protocol developers
- DLPI for networking card developers

With Open Transport, users have new implementations of both AppleTalk and TCP/IP. Later implementations will include additional protocols as well, including Novell's NCP/IPX. And keep in mind that if your code works with Open Transport versions 1.0 and 1.1, it will certainly work with later implementations of Open Transport.

Open Transport uses dynamic link-and-load memory management services, implemented through the Code Fragment Manager (CFM) for PowerPC processor–based Mac OS computers and the Apple Shared Library Manager (ASLM) for 680x0-based Mac OS computers. In Copland, these services will be provided by its microkernel. In all implementations, these features will permit networking services to be loaded and unloaded as needed and will minimize the amount of memory used by networking services when they are not active.

Open Transport's dynamic link-and-load memory management services also permit it to provide dynamic protocol configurations, which means users will be able to switch protocols (for example, change from a TCP/IP Ethernet connection to a TCP/IP dial-up connection) without restarting the computer.

Open Transport Provides a Superset of XTI

XTI gives developers an API for writing networking-aware applications. The XTI API is designed to be independent of the underlying data transport provider in any network, which means your application can use the same set of functions to send a packet of data, whether the underlying network is AppleTalk, the Internet, or some other network.

In other words, if your application calls the XTI routines, you don't need to be concerned with the user's network. And that means you can sell your application to a larger market—no longer will your application be disqualified from consideration because of its being tied to a specific networking protocol.

The XTI API is actually a superset of the industry standard XTI interface; that is, Apple has included additional calls for Mac OS developers within the XTI set of routines. These extra routines integrate support for C++ programming and provide access to AppleTalk transaction-based protocols such as ASP (AppleTalk Session Protocol) and ATP (AppleTalk Transaction Protocol), which help provide reliable datagram delivery in the AppleTalk environment.

The STREAMS API for Protocols

Developed for use by protocol-utilities and network-utilities developers (such as Apple, IBM, DEC, and Novell), the STREAMS architecture is a UNIX standard in which protocols are implemented as software modules that communicate using messages.

In the STREAMS environment, all modules have certain attributes. For example, they process messages asynchronously, they are single threaded, they share a single address space, and they cannot indefinitely hold up processing.

The STREAMS environment is important to protocol developers because it allows them to fit their protocols into the "middle" of the Open Transport architecture (see the figure "Open Transport Architecture" on this page), thus allowing their protocol to work with any program or hardware product that interfaces to STREAMS. And since many protocols already support STREAMS, this means potentially much larger markets than before for such products. (Two well-known protocol developers who already support the STREAMS environment are Novell, with IPX/SPX, and IBM, with its AIX implementation of UNIX.)

The DLPI API for Card Developers

For developers of networking cards, Open Transport conforms to the Data Link Provider Interface (DLPI) standards. The DLPI API is a commercial product from the X/Open Company that describes the content and ordering of the messages that are passed between software components and hardware drivers. The DLPI layer of Open Transport gives developers a standard interface to write to for developing networking card drivers.

Combined with the PCI (Peripheral Component Interconnect) bus standard, the DLPI interface helps to open a new world of networking product development opportunities based on broadly used industry standards. You can now build cards that run on a variety of platforms; for example, if you produce PCI-based networking cards for the Windows or UNIX platforms, the DLPI interface only requires you to write separate drivers for the Mac OS platform for your cards to work on any Mac OS computer that has a PCI bus. And when you write to the DLPI interface, you can write drivers using high-level languages, not assembly language—which makes your task easier.

Of course, your existing NuBus cards will continue to work with Open Transport's backward compatibility (described next), but they will not benefit from its added features. Apple recommends that you design your future expansion cards using the PCI bus.

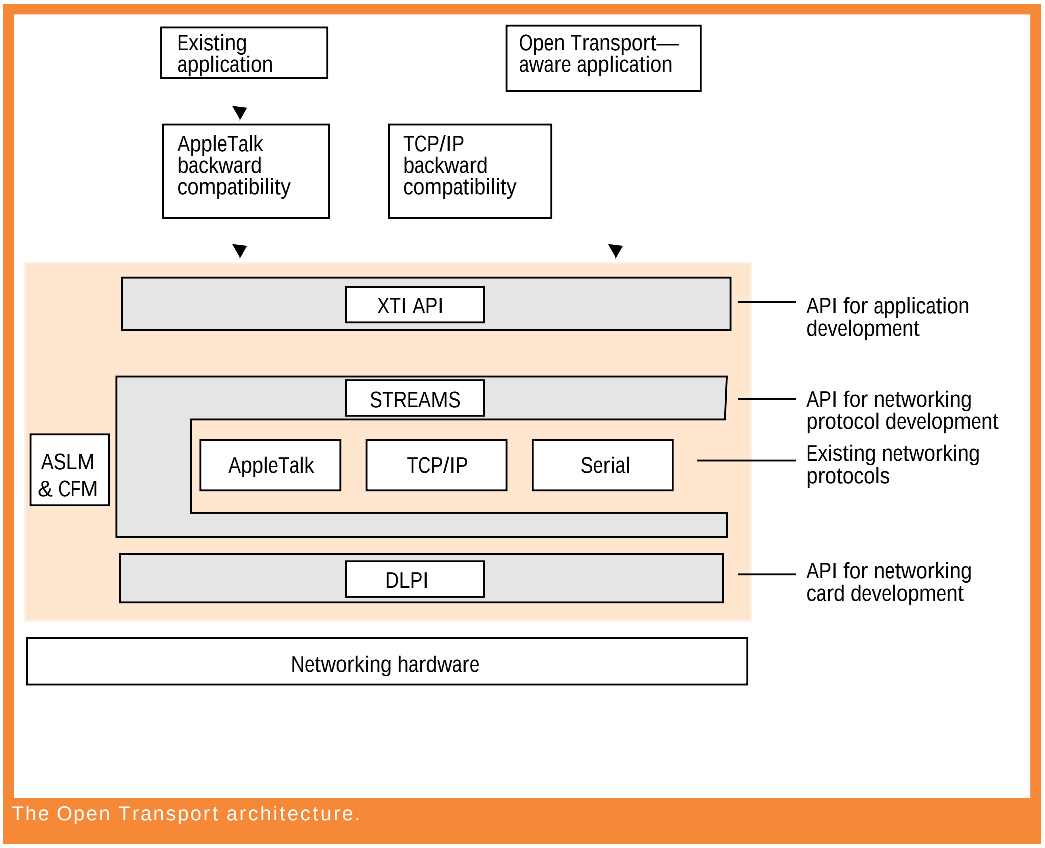
Backward Compatibility

While encouraging you to move to Open Transport as soon as possible, Apple realizes that you may not be able to do so immediately, and you need to support customers who are using your current products. In fact, a primary goal of the Open Transport 1.0 release was to provide this backward compatibility, to encourage the transition to Open Transport without "breaking" existing networking applications.

With Open Transport, backward compatibility is made possible by a set of libraries, based on the Apple Shared Library Manager (ASLM), that are included in the Open Transport "fat binary" libraries. These ASLM libraries are loaded with Open Transport and are dynamically linked with the applications that use them, permitting existing applications to function as before.

Open Transport also supports software that depends on the Communications Toolbox (which includes three pieces: the Connection, Terminal, and File Transfer Managers). Eventually, Open Transport/Serial (the Open Transport implementation of the serial protocol) will replace the Connection Manager. The Terminal and File Transfer Managers will continue to function; they will be PowerPC processor "native" and will run correctly in the Copland environment.

In summary, your existing applications will continue to work with Open Transport, but Open Transport will be the standard API


Existing application
Open Transport—aware application
AppleTalk backward compatibility
TCP/IP backward compatibility
XTI API
API for application development
STREAMS
API for networking protocol development
ASLM & CFM
AppleTalk
TCP/IP
Serial
Existing networking protocols
DLPI
API for networking card development
Networking hardware

The Open Transport architecture.

for networking in Copland. However, Open Transport provides a number of advantages—including increased performance and stability and a better human interface—that should encourage you to convert to Open Transport sooner rather than later.

Transport Independence

With current networking tools and systems, developers are forced to tie their applications to specific network infrastructure requirements. This creates a potential conflict between individual and organizational needs. If network managers dictate protocols to control support costs, users may not have access to the applications they need. If user dictate specific applications, they may increase support costs for the network manager by "dragging along" specific network infrastructure requirements. Developers are stuck in the middle and must guess which protocol will net them the largest installed base—and any choice they make will automatically cause many potential customers to no longer be interested in their product, simply because it is tied to a protocol other than the one they are using.

Transport independence is a concept that breaks this undesirable linkage. When implemented, it allows developers to write to a uniform set of APIs, users to focus on selecting the best applications, and network managers to make independent decisions about network infrastructure, all on an ongoing basis.

Open Transport is Apple's architecture for providing transport independence to Mac OS customers, while enlarging developers' potential markets for networking products. Open Transport supports transport independence by bringing the following three technologies together:

- a unified set of cross-platform, standards-based APIs for all networking and communications protocols (for example, the same application can send and receive data over either an AppleTalk network or the TCP/IP-based Internet using the same programming interfaces)
- a dynamic link-and-load architecture and set of protocols (protocols are loaded and unloaded on demand, conserving system resources, and making it possible to substitute say, TCP for ADSP at the application launch time)
- an addressing and naming support toolbox (for example, applications can open a communications end point by name, and Open Transport will automatically provide the appropriate name-to-address mapping services)

The use of Open Transport does not guarantee transport independence; for example, if an application uses a feature that is unique to a certain protocol, it will be tied to that protocol. However, Open Transport provides the foundation for a properly written application to be transport independent.

Levels of Open Transport Compatibility

Apple has determined three levels of Open Transport compatibility for software applications.

- A product that is *Open Transport–compatible* uses existing networking APIs but functions correctly with Open Transport installed.
- A product that is *Open Transport–ready* uses the new Open Transport APIs but is tied to a particular protocol stack; the Power Macintosh version must be native (that is, it must be PowerPC processor code, not 680x0 code).
- A product that is *Open Transport–enhanced* uses the new Open Transport APIs, is not tied to any particular protocol stack, and allows run-time configuration of the transport selection; the Power Macintosh version must be native.

You'll have to decide what level of Open Transport compatibility makes the most sense for your application, but clearly, Open Transport–enhanced applications will make the best use of Open Transport and sell to the widest audience.

Improved User Interface

Consistent with the benefits of transport transparency, Open Transport offers an easy-to-use interface and various network connection options. Open Transport makes a distinction between protocol software and configuration data. Since it stores network configuration information in a preferences file, these files can be stored for future use or distributed to end-users.

With Open Transport, network managers can provide their users with a variety of configuration setups in the form of preferences files; the users do not need to know routing addresses, host identification numbers, or even login names or passwords. This makes configuration easier for the user, while keeping control of network access in the hands of network managers. At the same time, individual users can save their own configurations to simplify the different kinds of connections they normally make.

Each individual user of Open Transport may select from three levels of access for each installed networking protocol:

- "basic/novice," which provides only the essential configurable choices
- "advanced/expert," which provides all configuration data and choices available for the given protocol
- "administrator," which is the same as advanced/expert, with the addition of a password-protected level of security, including an option to lock configuration items

For an example of the range of setup features, see the figure "AppleTalk novice setup" on page 22. In this example, note that there is an option for the user to define an AppleTalk protocol address.

The new implementation of AppleTalk supports an option to define and use manual protocol address administration, an alternative to the AppleTalk dynamic node addressing. This permits a network administrator to lock the user's address before distributing the preferences file.

In the Open Transport version of MacTCP, the user can configure an Internet setup with two simple steps: first, selecting a network connection (such as Ethernet), and then selecting the protocol (for example, the Dynamic Host Configuration Protocol, or DHCP). It would be hard to imagine how to make Internet access easier than this.

To assist the user in figuring out the details of protocols and configuration setup, Open Transport also provides Balloon Help and (with System 7.5) Apple Guide help. As a commercial or in-house developer, you can create customized guides that show users how to use your application most effectively—and reduce network support costs dramatically.

Developer Benefits of Open Transport

In general, all developers will gain two advantages from Open Transport. The first is a more stable networking infrastructure, higher performance, and better tools. When running within Copland, Open Transport will also provide a preemptive, multithreaded networking environment. The second is shorter development times for the human interface portion of your software; this will come about through the new Toolbox support that Open Transport will provide for network configuration and browsing.

In addition to these general benefits, different types of developers will see different advantages in Open Transport:

- *Application developers.* By creating applications that are not tied to one network architecture, you'll have a larger market for your work. By having to support only one networking API, you'll be able to bring products to market quicker and more easily. Your applications will run faster than was possible before, because both your application and Open Transport itself will be implemented in PowerPC processor code. And if you have an existing application based on DOS, Windows, or UNIX that does its networking through the XTI interface, Open Transport will speed the process of porting your application to the Mac OS platform.
- *Protocol developers.* The protocols you develop will run faster than before because you will implement them with PowerPC processor code. In addition, by tying into the STREAMS environment, you can implement your protocol with less effort than would otherwise be possible.
- *Networking card developers.* You'll benefit from the large, multi platform market for PCI cards. In addition, your driver code (being PowerPC processor code) will be very high-performance, and the DLPI API will make it easier for you to write your driver.

User Benefits of Open Transport

Open Transport makes it easier for users to connect to complex networks, and at the same time, it permits them to spend less time and effort figuring out the complex details of protocols, network architectures, and implementations. With Open Transport, users can work with the applications they choose and not worry about protocols, and they can access networks from remote sites without worrying about configuration settings. They will have built-in AppleTalk and TCP/IP protocols, and any Open Transport–compliant application or networking card will work.

In particular, users of PowerPC processor–based Mac OS computers will have a superior networking solution with Open Transport. PowerPC "native" applications that use Open Transport will run faster and more reliably than anything that is possible on Mac OS computers today. PCI-based Mac OS computers will benefit from increased bus speeds—permitting faster data transfer, for example, for high-speed Ethernet, FDDI (Fiber Distributed Data Interface), and ATM (Asynchronous Transfer Mode) connections.

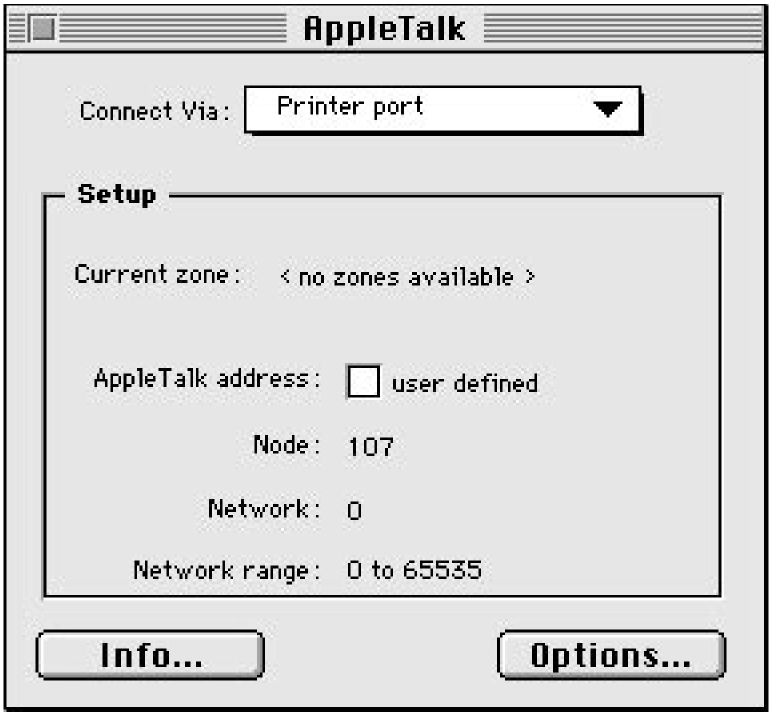
The Open Transport architecture also supports a feature called *multihoming.* With multihoming, users can have multiple networking cards active simultaneously, which allows them to do multiple networking activities simultaneously. While this feature is included in Open Transport versions 1.0 and 1.1, these releases do not provide a user interface for multihoming; developers can write their own (or wait for later versions of Open Transport that will include a user interface).

Network Manager Benefits of Open Transport

Finally, remember the plight of the network manager or administrator today, who must keep everyone else's computers running with an alphabet soup of protocols, data transfer mechanisms, applications, hardware platforms, and operating systems. With Open Transport, network administrators will be able to provide a variety of protocols to users; select various protocols dynamically, and access network routing information more easily; create preset configuration files to give to users; and build scalable networks that can be expanded without redesign and integrated with other networks that use the same APIs. These are features that need not be provided by applications built on Open Transport, but are a part of the Open Transport network architecture itself.

Open Transport Availability

Open Transport 1.1, which is expected to become available in the fourth calendar quarter of 1995, will have versions for both 680x0-based and PowerPC processor–based Mac OS computers. The 680x0 version will require a 68030 processor or faster with at least 5 MB of memory (8 MB recommended), running System 7.1 or later. The PowerPC processor–based version will require a PowerPC 601 or faster processor



AppleTalk

Connect Via: Printer port

Setup

Current zone: < no zones available >

AppleTalk address: □ user defined

Node: 107

Network: 0

Network range: 0 to 65535

AppleTalk novice setup. This is an example of the simplest level of networking access that Open Transport provides.

with at least 8 MB of memory (12 MB recommended), running System 7.1 or later. Open Transport 1.1 will also provide support for PPP (Point-to-Point Protocol) and will allow remote networking through either AppleTalk or TCP/IP.

Apple will make Open Transport 1.1 available to customers in various ways. It will be made available at no charge to customers with volume license agreements or software maintenance agreements for either MacTCP or Mac OS System Software. Users will be able to buy Open Transport 1.1 as either a single-user or a 20-user shrink-wrapped retail product. In addition, you (as a developer) will be able to license Open Transport 1.1 to ship with your products.

Developers Already Committed to Open Transport

In recent months, Apple Computer has seeded Open Transport to a number of developers, and many have announced products that support Open Transport. Naturally, Apple intends to support Open Transport in its own networking products. Look for announcements in the coming months of Apple's support for Open Transport in these Apple software products:

- QuickTime Conferencing
- AppleTalk Remote Access
- AppleShare
- Apple File Sharing
- PowerTalk
- PowerShare

As for third-party support, when this article went to press, 34 vendors had announced support for Open Transport; 13 of those also announced support for the PCI bus architecture. This vendor list includes:

- 4-Sight
- AGE Logic
- Asante Technologies
- Dantz Development
- DCA
- Digital Equipment Corporation (DEC)
- Farallon Computing
- InterCon Systems Corporation
- Metrowerks
- Neon Software
- Northwestern University
- Novell
- Pole Position
- SoftArc Inc.
- StarNine Technologies
- Systematics Softworks GmbH
- Vicom Technology
- Walker Richer & Quinn
- The Wallongong Group
- Wall Data

Opportunities for Developers

If you create products in any of the following categories, you can benefit from moving to Open Transport as soon as possible.

- client/server applications
- mail applications
- scheduling programs
- network gateways
- terminal emulators
- Internet browsers
- network-based games
- LAN analyzers
- data-link management software

Cross-Platform Opportunities

When developing products for the Mac OS platform, think about how Open Transport offers incentives to expand to other markets.

Over time, Open Transport will help provide increased integration for the Mac OS platform with Novell Netware networks and networking protocols, including IPX/SPX. The Novell network market is large and growing, and Novell's cooperation with Apple may provide added incentive and easier integration that will help you to bring your products to that market.

Also, the Open Transport architecture creates incentives to develop networking cards that run on the PCI bus. These include

- lower cost, since PCI-based cards are usually less expensive to develop than NuBus cards
- drivers that can be recompiled for other platforms without recoding
- PCI cards that can work on many Windows-based and UNIX-based computers as well
- a larger market for your Mac OS product—or, if you are a Windows/UNIX developer, an opening in the Mac OS market for your product

For licensing and software information, see the box "Open Transport Availability and Pricing."

Remember, Open Transport is available today. Now is the time to start working with Open Transport and begin planning its integration into your products. ♣

David Gleason is the president of Verona Communications, a World Wide Web services provider, and is a former localization manager for Apple Computer Europe. He can be reached on AppleLink at DAVI-DOVICH, or at the Internet address david@verona.com. His home page URL is http://www.batnet.com/verona.

Open Transport Availability and Pricing

Licensing for Open Transport will be similar to the terms and conditions for licensing MacTCP. Developers who have already licensed MacTCP will be able to upgrade to Open Transport for a small fee (to cover the costs of the disks and shipping).

Developers who have not licensed MacTCP can obtain details by contacting Apple Software Licensing at 512-919-2645, at the AppleLink address SW.LICENSING, or at the Internet address sw.licensing@applelink.apple.com.

The Open Transport Software Development Kit (SDK) is in the current Mac OS SDK, which is available from APDA (see page 36 for ordering information). In the United States, the cost of four quarterly mailings of the Mac OS SDK is $299; international developers should contact the nearest international APDA office.

News

develop *Issue 23*

continued from page 15

the corresponding document names at all times.

That's not all: There are also columns that discuss the new generation of Power Macintosh computers, improved color management in ColorSync 2.0, customizing source control with SourceServer, and tips for organizing AppleScript dictionaries. Dave Johnson managed to work boomerangs into his Veteran Neophyte column, and KON and BAL (slackers that they are) once again found someone else to write their Puzzle Page for them.

Like every issue of *develop,* this one will entertain as well as edify you. So check out the articles and columns (now in Adobe Acrobat) and the code they describe on this month's Developer CD. Or browse the printed copy if you've subscribed to *develop* through APDA. We think you'll like what you see, but if you don't (or even if you do), be sure to let us know by sending an AppleLink message to DEVELOP.

Caroline Rose
Editor, develop

What's New on Mac OS SDK #4

The Mac OS Software Developer's Kit (SDK) gives you access to almost all the Macintosh system software extension SDKs that Apple publishes. Release #4, September 1995, of the Mac OS SDK includes over 30 individual SDKs on two CD-ROM discs. The Mac OS SDK is included in the September developer mailing (the same one including this issue of *Apple Directions)* for Apple Partners worldwide and Apple Associates Plus in the United States only. It's also available separately from APDA starting in mid-August.

Here's what's new on Mac OS SDK #4, listed by the appropriate folder name.

Apple Guide 1.2.8

This folder includes a few bug fixes and an electronic version of *Apple Guide Complete: Designing and Developing Onscreen Assistance.* This book is published by Addison-Wesley Publishing Company and is available in bookstores worldwide.

Apple Shared Library Manager 2.0

Here you'll find the final version of ASLM 2.0, which allows you to create and use dynamically linkable and loadable shared libraries on 680x0 and Power Macintosh systems.

AppleSearch Client

This folder includes AppleSearch Client for Windows 2.0.

ColorSync 2.0 Rev

The new ColorSync 2.0 Rev software is very similar to the version of ColorSync included on Mac OS SDK #3. The copyright string has been changed in the ColorSync Extension and the ColorSync System Profile files, and a new profile document called Apple Multiple Scan 15 has been added to the profiles folder. The CSDemo sample code has also been slightly revised. Also included is final electronic documentation covering advanced color imaging on the Macintosh.

Display Manager

This is the first release of the Display Manager SDK on the Mac OS SDK. The Display Manager allows a user to dynamically change the arrangement and display modes of the monitors attached to the Macintosh computer. For example, a user can move displays, add or remove displays, switch displays to higher or lower screen resolutions, and move the menu bar from one display to another—all without restarting the computer. The SDK provides information and examples of how to support the Display Manager in your applications.

Interfaces & Libraries

The set of interfaces included here is the version 2.1 standard set from MPW Pro #18. Also included are the Interface.o (680x0) and InterfaceLib.xcoff (PowerPC) standard Toolbox libraries. This folder exists on both CD-ROM discs that make up Mac OS SDK #4.

Macintosh Drag and Drop

Included here is a new version of the FinderDrag sample code.

MacODBC 2.1

This folder contains the new release of MacODBC, which is compatible with version 2.1 of the ODBC specification. Includes programming interfaces and libraries and sample code for an ODBC application, an ODBC driver library, an ODBC setup library, and an ODBC translate library.

Network Software Installer 1.5.1

Here you'll find the latest version of the installer for a variety of networking products, including LaserWriter Bridge, AppleTalk, Network Control Panel, EtherTalk, Apple Ethernet, TokenTalk, Token Ring, and A/ROSE.

PCI Driver Development Kit

The information in this folder explains how to develop drivers for PCI cards that can be used with the Power Macintosh 9500 computer (and future Power Macintosh models). Note that because of licensing restrictions, the Apple PCI/Open Firmware tokenizer (based on CFORTH93), required for code development, cannot be mass distributed. It's available from APDA (#R0680Z/A).

PlainTalk

New this time is version 1.4 of Apple's text-to-speech products—English Speech Recognition and Text-to-Speech, and Mexican Text-to-Speech. This version also includes a pre-release version of Apple's speech recognition toolbox. This version of the toolbox runs only on Power Macintosh systems.

QuickDraw 3D 1.0

This is the first appearance of QuickDraw 3D on the Mac OS SDK. QuickDraw 3D is a cross-platform 3D graphics library developed by Apple. The SDK includes a considerable amount of information for developers, including extensive sample code and documentation.

QuickDraw GX 1.1.2

This folder includes the installer for QuickDraw GX version 1.1.2. In addition, the GX interface files have been renamed, consolidated, and improved—they're now part of the standard set included in the Interfaces & Libraries folder. Lastly, some new sample code and revised samples of older code have been included.

QuickTake Digital Camera

This SDK for the QuickTake Digital Camera is included for the first time, including both Macintosh and Windows versions.

QuickTime Conferencing

The QuickTime Conferencing SDK is new to the Mac OS SDK. It describes a system software extension that your application software can use to set up video conferences, collaborate with others, exchange live video and sound, and share information over a network. The QuickTime Conferencing software requires a Macintosh Quadra 660AV or 840AV computer, Power Macintosh AV computer, or a Power Macintosh with an AV card upgrade.

Sound Manager

The Sound Manager system extension version 3.1 is included for the first time together with its companion Sound control panel version 8.0.5. This version supports two new compression formats (IMA and uLaw); it performs much faster on Power Macintosh systems. An entirely new set of programming information is also included.

Telephone Manager 2.1

The Telephone Manager folder has been completely overhauled and now includes version 2.1 software, new interface files, and new 680x0 and Power Macintosh libraries. In addition, the Stiletto program for testing the Telephone Manager API can now be built as a fat binary.

Thread Manager

The folder includes Thread Manager version 2.1.1d1+, which is designed for use by applications that use the CFM-68K (Code Fragment Manager for 680x0 Macintosh systems) run-time model.

The Mac OS SDK is shipped on a quarterly basis. A one-year subscription to the Mac OS Software Developer's Kit is available from APDA for $299.00 (U.S.). See page 36 for APDA ordering information. ♣

CD Highlights

continued from page 15

Display Manager Development Kit

This development kit has all the information you'll need to begin taking advantage of the Display Manger API. The Display Manager 1.0 API was first introduced for the Power Macintosh computer in System 7.1.2 and subsequently shipped in System 7.5 and System 7.5.1. The Display Manager 2.0 is included in post-7.5.1 systems and is also available as a system extension that is backward compatible to System 7.1. Licensing information for the Display Manager extension is included to make it as easy as possible for you to create and ship Display Manager–aware applications.

DisposeResource 2.0

DisposeResource is an extension that catches a common programming error: passing a resource handle to DisposeHandle. Version 2.0 correctly deals with ROM resource handles. This avoids an erroneous report during startup with Sound Manager 3.1 installed.

Drop•PS 1.1.3

Drop•PS provides the ability to download PostScript text files, as well as EPS (Encapsulated PostScript) files, to a network-connected PostScript™ printer. Files may be downloaded in the background, and entire folders or disks may be downloaded as well. This updated version fixes a bug that prevented Drop•PS from seeing printers in zones other than the default zone.

Note: This is not an Apple product, and is provided on an "as-is" basis. Apple is not responsible for any problems you may encounter in its use. Drop•PS is freeware, provided by Bare Bones Software.

DTS QuickTime Utilities.July-95

This is a collection of useful QuickTime functions and applications. Consult the documentation inside the Documentation folder for more details. See the document What's New, July-95 for information on updates and bug fixes.

Gestalt Selectors List 3.0

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

CD Highlights

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, and is provided on an "as-is" basis. Apple is not responsible for any problems you may encounter in its use.

Macintosh Technical Notes

Technical Notes are a collection of short (and not-so-short) articles dealing with specific development topics. New & revised Technical Notes for September 1995 include the following:

- IM—Files Errata
- NW29-File Sharing Ext 7.6.1
- OV16-Gestalt & _SysEnvirons
- QT07-TV Tuner Component
- QT08-Teletext Component
- TB42-Finder Icon Position

Macintosh Technical Q&As

Macintosh Technical Q&As are designed to provide you with solutions, workarounds, and code snippets to help your development efforts. The questions are those that Apple's Developer Support Center (DSC) has received on specific topics, along with answers from DSC engineers.

MacODBC 2.1

MacODBC is the Macintosh implementation of the industry-standard Open Database Connectivity protocols. This SDK is for developers who want to develop ODBC drivers or applications. It includes the ODBC Driver Manager (for distribution with drivers or ODBC applications), as well as an ODBC Tools disk with headers, sample drivers, and sample applications. Complete documentation is also included. MacODBC 2.1 provides "native" operation for applications and drivers on Power Macintosh computers. It supports version 2.1 of the ODBC specification.

OpenDoc Documentation

This folder contains the principal developer documentation for OpenDoc for the Mac OS. The subfolders contain the following documents:

- the *OpenDoc Class Reference,* in two formats: Adobe Acrobat, indexed for full text search, and QuickView, which gives you hypertext access to the OpenDoc API descriptions in the same manner as Macintosh Programmer's Toolbox Assistant
- the *OpenDoc Programmer's Guide,* in Adobe Acrobat format, indexed for full text search
- the *OpenDoc Cookbook,* in Adobe Acrobat format
- the *Drag and Drop Human Interface Guidelines,* in Adobe Acrobat format, which contains interface-behavior guidelines for Macintosh Drag and Drop (OpenDoc human interface guidelines, which were previously in this folder, are now part of the *OpenDoc Programmer's Guide*)

Note: All documentation is preliminary, and may contain errors. The information is incomplete in some cases and it's not guaranteed to be current with the latest version of OpenDoc.

PCI Driver Development Kit

This folder contains information on designing PCI cards and drivers for the PCI-based Power Macintosh computers, such as the Power Macintosh 9500. This information will also apply to future Macintosh models that incorporate the PCI bus.

PopupFuncs 2.6

PopUpFuncs is a productivity tool for developers. When you click on the PopUpFuncs control in a source code window's title bar, it creates a pop-up menu of every function contained in the source code file, allowing you to see instantly the contents of an unfamiliar file. If a name is selected from the menu, the file is instantly scrolled to the beginning of that function. PopUpFuncs works with CodeWarrior, MPW, THINK C/Symantec C++, SADE, BBEdit, and QUED/M, and parses C, C++, Pascal, Object Pascal, Assembler, Rez, and Fortran source files. See the file PopupFuncs Notes.c for details.

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

RAMDisk 1.2

This INIT/CDEV/DRVR combination demonstrates many of the techniques used to create a RAM disk and its accompanying control panel. It includes a sample driver, extension, and control panel written so it compiles in Metrowerks CodeWarrior 6.0, Symantec Think C 8.0, and MPW.

ShrinkWrap 1.4.2

ShrinkWrap is a utility for creating and mounting DiskCopy images. It will also mount many other formats of disk images and can utilize the StuffIt Engine for compression. See the file ShrinkWrap Read Me for more information.

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

Toolbox Assistant Update #3

This folder contains files that update the Toolbox Assistant QuickView databases. This update release contains the Toolbox information from the new volume Advanced Color Imaging Reference. The Routines A. . . Z and Mgr.qv file contains updated Routines and Managers pages.

Coming Next Month

Next month's disc will include Macintosh System Software, for hardware new and old.

Alex Dosher
Developer CD Leader

Business & Marketing

Market Research Monthly

Inside This Section

WWDC Developers Plan Aggressive Adoption of OpenDoc

Sometimes the best way to find out what's going on in the market is to find out what your competition is up to. This month, we give you insight into the market for component software by presenting data about how OpenDoc fits into the plans of some of your developer colleagues/competitors. The survey indicates that they have aggressive adoption and product release plans; if that's any indication, soon after the first OpenDoc customer release this fall there should be a large and growing OpenDoc component market.

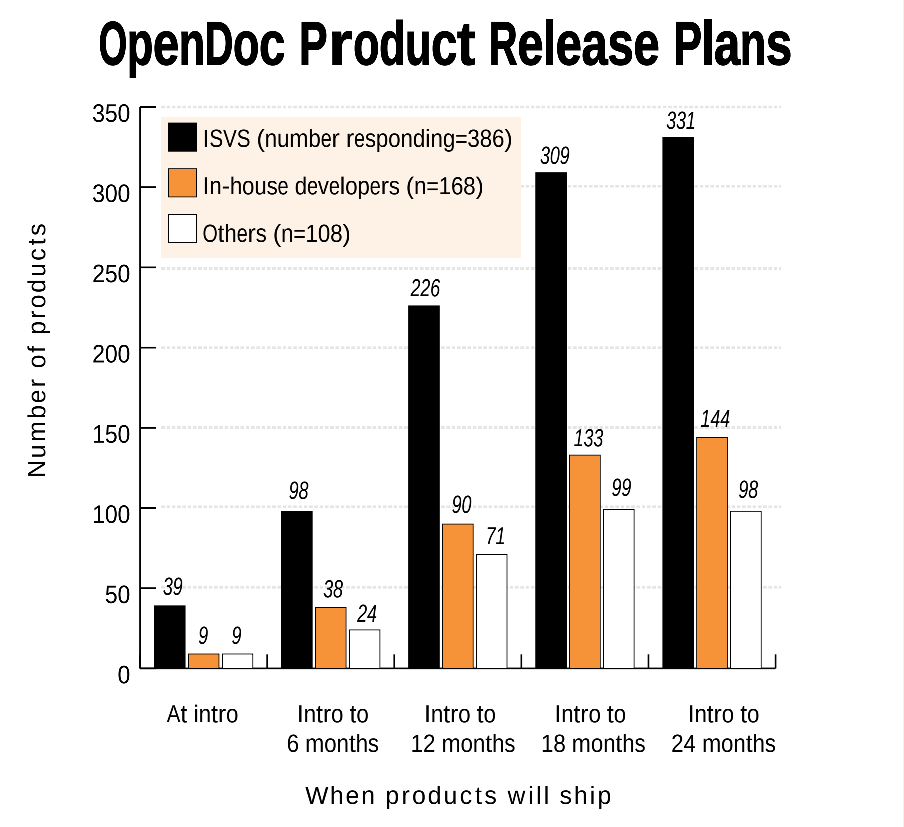
The data was gathered at the May Worldwide Developers Conference (WWDC), where Apple Computer, Inc., introduced Developer Release 2 of its architecture for developing component software. The survey was completed by over 25 percent of the 3,500-plus conference attendees (a remarkably high percentage, compared to other surveys we're aware of)—including independent software vendors (ISVs), in-house developers, and others (such as value-added resellers, consultants, and information systems developers).

More than 86 percent of both ISVs and in-house developers who responded said they planned on developing an OpenDoc product. Of the ISVs surveyed, nearly 60 percent said they'd have a product on the market within a year of the December OpenDoc introduction, while just over half of in-house developers expect to complete their OpenDoc-based applications within the first year. Within two years of the first OpenDoc release, those surveyed expect to make available nearly 600 OpenDoc products. (See the graph "OpenDoc Product Release Plans" on this page.)

More than a third of each group—35 percent of ISVs and 38 percent of in-house developers—plan to develop a stand-alone OpenDoc component, while at least one-fourth of the developers—31 percent of ISVs and 26 percent of their in-house colleagues—expect to completely redo current applications and release them as a set of OpenDoc components.

Why do developers feel so bullish about OpenDoc? These three factors emerged from the survey as the main reasons driving developers to adopt OpenDoc:

- OpenDoc provides cross-platform support; software written to the OpenDoc



This graph shows the number of OpenDoc products that developers surveyed at the 1995 Worldwide Developers Conference expect to produce within two years of the first OpenDoc customer release.

application programming interface (API) can be used on a variety of platforms—currently Mac OS, Windows, OS/2, and AIX systems, and, eventually, others as well.

- The OpenDoc component architecture ensures that all OpenDoc software will have access to the features of other OpenDoc software, including future Mac OS technologies; developers and users will be able to integrate new technologies with very little effort.
- The OpenDoc component architecture enables the same code to be reused in multiple products; for example, if you write a graphing OpenDoc component, you can use that same component as part of a spreadsheet, database, or word-processing package.

The question "What would you like to tell the engineering team?" received hundreds of responses, most of them very positive. A few of the most notable were the following:

- "Ship it, ship it, ship it!"
- "CyberDog is very cool use of OpenDoc."
- "How do you make money in the parts world? What will the price points be?" Note that these two questions are still in need of answers; Apple is working on the details of the business side of OpenDoc development.)
- "The product looks great; it's greatly superior to OLE even with Microsoft's proposed changes to OLE."
- "Excellent demos. I really see the picture this year. Last year it did not make a great deal of sense."
- And our favorite: "I am now considering wearing suspenders." We assume this comment was made by someone paying homage to OpenDoc architect (and well-known wearer of suspenders) Kurt Piersol.

Keep in mind that the survey participants are at the vanguard of Mac OS development. They wouldn't have come to the WWDC if they weren't already highly motivated to adopt new Mac OS technologies and try to be among the first in the marketplace with leading-edge products. It's likely, though, that wherever the leaders go, everyone else will soon follow.

We'll put it this way: If just the few WWDC attendees who responded to the survey expect to release nearly 600 OpenDoc products by the fall of 1997, we can only imagine how many other OpenDoc components will be made available in the same period. Instead of waiting and following the pack, you'll be able to boost your competitive position by adopting OpenDoc today. ♣

Editor's note: If you'd like to look at some of the the first available OpenDoc components, check out the Component Warehouse at Apple's OpenDoc Web Page, located at the following site: http://www.info.apple.com/opendoc/

Business Feature

Launching a New Product— The Pop Rocket Story

By Kris Newby

Let's say you're an aspiring multimedia title producer. You decide to bet your entire personal fortune on your dream—to create the next big hit. So you quit your job, borrow from relatives, and run your credit cards up to the limit. Your motto: Achieve fame and fortune, or die trying.

Sound familiar? It's not only the dream of many a developer—it's the story of Pop Rocket, a San Francisco–based company, who after three-and-a-half years and more than $250,000 is about to ship their first Mac OS entertainment title, *Total Distortion.*

Developers can learn a great deal from Pop Rocket's story. In this article, Pop Rocket's three founders, Joe Sparks, Maura Sparks, and Kent Carmical, talk about the birth process of their new product, from design strategies to market launch. And with the wisdom of 20/20 hindsight, they offer advice to others with the drive and intestinal fortitude to turn an idea into a new product.

A Spark of Inspiration

Pop Rocket's president, lead programmer, sound engineer, graphic designer, and visionary is Joe Sparks. Joe started out as a musician, working in a band called *The Agents.* He taught himself programming, then in 1987 started Artificial Works, a multimedia production company that worked on projects for Apple Computer, Inc., NASA, Macromedia, and Paracomp, to name a few. After this he co-developed the successful adventure game, *Spaceship Warlock,* which was published by Reactor, Inc.

Sporting a pony tail, Elvis sideburns, and a "soul patch" (a micro-goatee), Joe Sparks looks more like someone you'd see behind a guitar than a Power Macintosh. Thoughtful and soft-spoken, Joe talks about the vision that motivated him to design and publish his own interactive media titles.

"With the diverse set of skills that Kent, Maura, and I brought to the table, I felt we could put together a unique music-intensive title for the CD-ROM market," he says. "The idea that first appealed to me was a rock-and-roll theme park with outrageous creatures and side shows. So I spent about a month creating a rough concept demo. After showing it around, I found that a lot of people responded well to a feature that let users create their own music videos. Based on that feedback, I expanded that concept into what ultimately became *Total Distortion*—what we call, for lack of a better description, a music video adventure game."

In *Total Distortion,* you, as a player, assume the role of a music video producer. The objective: to win fame and fortune by creating and selling MTV-like music videos. Riding in a rocket lavishly equipped with media-editing hardware, arcade games, books, and a deluxe kitchen, you travel to the dangerous Distortion Dimension to gather media for your music video. To get the best material, you have to overcome a series of deadly obstacles, including guitar warriors, puzzles, traps, and starvation. One of the most fun features is the media-editing station, which enables you to create your own music videos and share them with other users.

"What impresses people most about *Total Distortion* is the incredible number and variety of activities that you can undertake, from boxing with robots to listening to an hour and a half of radio shows," says Joe. "There's really something for everyone—whether you prefer action adventure or mind games."

Music to Their Ears

Kent Carmical, Pop Rocket's lead musician, songwriter, and sound designer, started out in the music industry, creating and engineering music for San Francisco Bay Area groups. He worked in recording studios for awhile, then began composing computer-based music for companies such as Atari and Apple Computer.

Kent got to know Joe Sparks while performing with the Agents. When Joe decided to create his own company, Kent jumped at the chance to join. "It seemed like a good opportunity to work with a good buddy, and do what I love—make music and live in a fun city."

Both men are believers in the importance of music in interactive media, and it shows—a beta version of *Total Distortion* won *NewMedia* magazine's 1994 Invision award for best audio/soundtrack.

"Music lets you communicate complex emotions and drama so much better than with just text or visual cues," says Joe. "And beyond the psychological value of a music track, it provides you with an economical way to provide users with feedback, saving you valuable programming and animation time." (For more on this topic, see the article "Working With Composers in Multimedia" on page 31.)

Technical Challenges

Total Distortion breaks new ground in the multimedia arena—most notably in music integration and user interactivity—and, as you might expect, the Pop Rocket team had their share of technical challenges to overcome.

Their first and easiest technical decision was which platform to develop the product on. "We chose to develop *Total Distortion* on the Macintosh first, then move it to Windows later, because all our favorite tools are on the Macintosh," says Joe.

The Pop Rocket team used Macromedia Director as their authoring environment, using its built-in object-oriented language, Lingo, for programming tasks. They modeled characters and objects in Macromedia's Swivel 3D and enhanced objects, textures, and backgrounds in Adobe Photoshop. Once objects were created, they were dropped into ElectricImage Animation for the rendering and animation part of the process. Director was used to assemble the audio, animation, and video tracks into the final title.

In the beginning of the project, Joe realized that getting good performance from his media-rich title would be one of his biggest technical challenges. So early on in the project, he made a strategic decision to use as many *sprites*—little graphic objects used to build animations—as possible.

Rather than creating animations à la Disney—where each still frame of an animation is stored as a unique image—frames were broken down into object-oriented sprites and assembled on demand in Director. The unmoving part of a frame was stored as one object, and moving parts—eyes, mouths, doors, and so on—were stored as different objects. As needed, each of these sprites were called into a frame. Though resulting animations may not be as natural as a Disney production, reuse of sprites significantly saves memory and improves access time.

For engineering and mixing digital and MIDI sound, Joe and Kent relied heavily on Opcode's Studio Vision, a much-loved tool that they've been using for about eight years, as well as Sound Design, Audio Media, and SoundEdit Pro. MIDI and digital audio were digitized in SoundEdit Pro. Inside SoundEdit Pro, the audio was reduced to an economical 11K footprint. While CD-quality audio provides higher quality audio (a necessity for reproducing movie soundtracks and human voices), the lower sample rate saved valuable memory without a noticeable quality difference.

The final title's performance was further enhanced by giving users the option to load 4 MB of frequently used code, small sounds, and interface pieces from the *Total Distortion* CD to the user's hard disk at startup. If users have plenty of hard disk space, they can load 43 MB of frequently used code, graphics, and music, which will result in greater speed in playing with the video sequencer. This improves performance, since hard disk drive access time is so much faster than that of a CD-ROM drive.

Creeping Featurism

One of the ailments that most creative developers face at one time or another is "creeping featurism"—indecisiveness in when to quit adding product features and start debugging. Clearly this factor contributed to Pop Rocket's lengthy three-and-a-half year development schedule, along with the fact that they were setting out to create a new genre of title, where they couldn't learn from their predecessors. Joe was left to work out the intricacies of an interface that would enable users to easily gather and edit media clips, create custom MTV-like music videos, and interact with video producers.

Another problem that often delays schedules is the pace at which development tools change. Programmers are often faced with the painful decision of staying with existing tools, or moving to a better version or technology that may save them time in the long run.

When Pop Rocket started their project more than three years ago, Macromedia Director was the best authoring tool on the market. There were no cross-platform authoring tools available—just Macromedia's promise that they would soon ship a Windows version of Director.

"We were surprised by the amount of code we had to change in order to move over to the new version of Director," admits Joe. "We had to learn a new scripting language and make architectural shifts that delayed our schedule a couple of months. One of the most tedious changes we had to make was to overhaul all our file and path names to a Windows-readable format—eight characters with a dot and suffix."

Since that painful transition, Joe has been impressed with enhancements that Macromedia has made to Director. And his team has honed the process of testing their Mac OS code on a Windows-based system: Using Farallon's Timbuktu product to send files over AppleTalk, they can now run frequent tests for Windows compatibility.

The Money Trap

Pop Rocket optimistically started their venture with a $50,000 budget and an estimated nine-to-ten-month schedule, the same amount of time it took to create *Spaceship Warlock*. But both the budget and schedule exceeded expectations as the size and complexity of the project increased. (*Spaceship Warlock* was only 124 MB in size versus *Total Distortion*'s 600 MB.) The development team got used to working hard and living cheaply, but after a year, the cash was gone, and the credit cards were loaded. At that point they visited distributors with a demo in hand, and soon signed an affiliate label deal with Electronic Arts. This arrangement provided them with a cash advance in exchange for exclusive North American distribution rights.

While the advance held them for awhile, their schedule crept out further, and a deal with a Japanese distributor, HatNet, saved them yet again from ruin.

As their venture passed its second year, Pop Rocket hit financial rock bottom. They decided their only recourse was to seek outside investment money, so with an alpha version of *Total Distortion* on a hard drive, they began talking to investors. Word-of-mouth enthusiasm for their product spread quickly, and before they knew it, Mark Gorenburg at Hummer Winblad Venture Partners approached them with a deal that ultimately ensured a healthy product launch.

At that point the industry PR machine took over. A beta version of their product won several major industry awards. *Business-Week* and the *Wall Street Journal* ran feature stories on their company. Celebrities such as Francis Ford Coppola and Robert DeNiro were calling them to "do lunch." And all looked rosy until they committed the ultimate faux pas—they missed shipping in time for the 1994 holiday season.

And while this error might have been fatal to lesser products (or might have resulted in banishment to a ten-pack bundling deal), *Total Distortion* seems to have emerged unscathed. Case in point: *Multimedia World* just gave the product a rare five-star rating, saying, "*Total Distortion* stands up to the test of time because its premise and artistic vision are as original and as appealing as when the title was first announced."

Guerrilla Marketing Tactics

Maura Sparks manages the business side of Pop Rocket, calling upon her experience as a partner in Artificial Works and a Paracomp marketer. Like the other Pop Rocket founders, she's a jack-of-all-trades, acting as official videographer, vocalist, and computer artist. She's also playing a key role in formulating their marketing strategy, along with Andrea Luskin, vice president of sales, and Kevin Krejci, marketing manager.

Maura and her team face two marketing challenges as *Total Distortion's* September ship date approaches—reviving PR efforts after hovering in the "vaporware" zone for about a year and educating consumers about why they need a "music video adventure game."

While larger multimedia publishers are introducing products with million-dollar marketing budgets, Maura realizes that her team will have to use some creative (that is, inexpensive) marketing tactics to get the attention of consumers.

"One thing that really surprised me was that you need about $100,000 in co-op dollars [matching funds for in-store promotions] to get good retail shelf position and advertising exposure," says Maura. "The big retailers also want to know that you have a well-rounded marketing strategy and a follow-through plan beyond the introduction. We will rely on a lot of unconventional and alternative marketing tactics to compete."

Another interesting issue centers around packaging: Even though they could probably ship both the Windows and Mac OS versions of their product on a single *hybrid* CD (since many of the same files are shared), their distributors have convinced them to ship their disc in two different boxes, one marked as a Mac OS product, the other as a Windows product. They recommend this approach because all too often retailers accidentally place all the Mac OS versions of a product on the Windows shelf, or vice versa.

Pop Rocket, with their limited marketing budget, aims to amplify their marketing efforts through distribution partnerships. They've signed an affiliate label agreement with Electronic Arts for North American distribution and formed localization/distribution alliances with HatNet, NEC and BMG in Japan and Mindscape in Europe. To maximize the impact of word-of-mouth marketing and reviews, they're rolling out their PR campaign in stages, hitting San Francisco and New York City first, then Los Angeles and Boston.

Some of the other marketing tactics they're employing to reach their target audience include

- a Pop Rocket Internet page that promotes a contest for the best "home-grown" music videos and encourages users to swap their own creations
- a low-cost demo CD sold through retail stores in advance of the product's release
- a music CD of their product's soundtrack, which will generate additional revenues and radio publicity
- a live performance of their soundtrack in Japan, including big-screen appearances by their title's animated characters
- product giveaways through MTV and college radio stations
- in-store appearances and promotions

On the marketing side, Maura is optimistic. Advance sales are well beyond predictions, and they've already begun receiving fan mail from the beta copies in distribution.

20/20 Hindsight

When I interviewed the Pop Rocket team, they were eight days away from freezing their code. The three and a half years of hard labor were just about over, and with 20/20 hindsight, they offered advice to others interested in creating a new product.

An older and wiser Joe Sparks muses, "If I were to do this all again, I'd rein in the scope of the project sooner and try to do a few things really well. Having to raise money as you go is extremely nerve-wracking, so I'd recommend creating a great concept demo up front so you can get adequate funding early on."

On the human side of a new venture, Kent Carmical adds: "Start-ups aren't for the faint of heart. They're all-consuming, and it's hard to find an easy balance between your personal and work life. Make sure your family knows what's ahead—you're going to be working long hours for quite awhile."

As the first chapter of Pop Rocket's story comes to a close, *Total Distortion*'s promotional slogan, "Achieve fame and fortune or die trying," takes on a personal meaning to its founders. Now they face a new set of challenges—managing the day-to-day operations of their product roll-out and figuring out what their next title will be.

So, will their product be a hit? I think I'll save that question for the reviewers, because right now I've got *Total Distortion* running on my other system, and this satanic heavy-metal guitar warrior is about to blow me away just because I created this really bad music video. ♣

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

Developer Outlook

Working With Composers in Multimedia

By Bob Safir, InterOctave

Turn on your favorite movie or computer game, turn *off* the soundtrack, and observe the effect. You'll probably notice that the story or game loses much of its emotional impact. Music communicates. In multimedia, it enhances a user's experience, setting a mood and drawing users into your title. Musical sounds also provide users with valuable feedback, enabling game or content developers to simplify a title's user interface.

In the last few years, software producers have begun to realize the value of music—they've learned that good music is good business. A catchy theme song can lure shoppers to your retail demo. Distinctive music can be used to enhance product recognition through television or radio spots. And some soundtracks can even be sold as audio CDs to generate additional revenues and radio publicity. (See "Launching a New Product—The Pop Rocket Story" on page 28 to see how one developer is making the most of its music assets.)

These business reasons—along with the widespread availability of music-ready systems such as the Apple Macintosh computer—are motivating more multimedia developers to hire professional music composers to create soundtracks. As a composer and the co-chairman of the MIDI Manufacturers Association's Interactive Audio Special Interest Group (IA SIG), I represent the musician's point of view on hiring composers. In this article, I offer advice on how multimedia producers can get the best possible soundtrack for their money, with ideas on creative financing, evaluating composers, and providing creative direction.

How Much Is This Gonna Cost?

During a recent discussion on contractual issues facing multimedia composers—buy-outs, royalties, work-for-hire, and so on—a colleague of mine said, "Multimedia is just like the music business, but you don't get paid." I responded, "How is that different from the music business?"

Composers often wonder why most projects don't have budgets for music. They can often hear the words before they're formed on the lips of multimedia producers or project managers: "We'd really like something great—unfortunately there's very little money." (You can substitute the words *budget, time, room, resources, funding,* and so on, for the word *money* in the preceding sentence.)

To some extent, it seems to be a buyer's market in which there are many more talented (as well as untalented) composers needing work than there is work available. In multimedia production, I suspect that not enough importance is placed on music to begin with, so not enough is allocated to it. This is unfortunate, since music is the thing that delivers emotional impact to a title. Try running that scene again with no audio. It's not quite the same, is it?

One of the difficult aspects of hiring a composer these days is that there are no rules, traditions, or guidelines on compensation. As we all know, there is no such thing as a "Standard (fill in the blank) Contract"—especially a standard *multimedia* contract.

And producers and composers come from dramatically different worlds when it comes to content ownership. Multimedia producers live in a world where content-delivery technologies change on a yearly basis, and they're used to grabbing all the future content rights they can. In their emerging market, unit sales are hit-or-miss propositions, so producers shy away from offering royalties.

Composers, on the other hand, take their cues from the music business. They're used to "owning" their content, receiving unit-based royalties, and granting producers limited use rights. Many composers cringe at the term *buy-out* or *work-for-hire,* not necessarily because of the financial aspects, but because of the legal ones. Although work-for-hire agreements are typical in television production, the composer still retains the performance rights, which often yield thousands of dollars in ASCAP or BMI royalties. (ASCAP and BMI are performance rights clearinghouses that manage the payment of royalties to composers when their music is played.) In addition, composers always retain "writer's rights" in the music business. In contrast, the wording in many multimedia contracts states, in essence, that a composer is not the author of the material (because the production or software company is "granted" those rights).

Ideas for Creative Financing

My advice to multimedia composers and producers is to explore creative ways to finance great music, and try to meet each other somewhere in between. While there are many fee structures for licensing music from music libraries, no one thinks of using the same or similar models for licensing *original* music. And don't rule out unit-based royalties, because there's one key benefit to getting composers invested in the success of your product—free PR. As with music CDs, movies, and books, royalties are "carrots" that motivate artists to talk up your product at conferences, on radio shows, to the industry press, and so on.

Here are some "creative financing" ideas for working with composers in the United States.

- *Flat fee with a sliding royalty scale.* If, like many multimedia producers, you're strapped for cash, consider offering a composer a low up-front fee in exchange for royalties—a percentage based on future unit sales. That way, if a soundtrack helps your title become a bestseller, everyone wins.
- *Work-for-hire fee, granting the composer writer's rights.* As a producer, don't automatically retain all creative rights—honestly evaluate which ones you really need. (Is there *really* a chance that someone will want to license your title's music for a Broadway show?) There is no reason why a production company can't be granted an unlimited, *exclusive,* worldwide right to use the music in its production with remuneration identical to a work-for-hire agreement. The advantage to composers is that they retain the writer's copyright (not *publishing* rights), and they might be able to reuse the material for a completely different, noncompetitive application should it ever arise. (For example, if you leave your composer the option of including some of your title's music on an audio CD at a later date, you may

be able to get your soundtrack created for a more reasonable cost.) Also, you should always let your composer retain performance rights with either BMI or ASCAP.

- *Up-front fee with a cost-recovery bonus.* Another way to reduce up-front costs is to pay your composer a flat up-front fee, then provide a bonus or royalty incentive to the composer once the title earns back its development costs.

While there's usually a negative connotation to the term *creative financing,* any type of agreement that makes sense can be written and can be a win/win situation for the parties involved.

The Eleventh-Hour Syndrome

One typical complaint heard from interactive multimedia composers (would non-interactive composers be *passive* composers?) is that they're called into a project too late in the process. It's not uncommon for a composer to be called in the last four to six weeks of a major feature film project or the last few hours of a multimedia production. The problem is not simply that the composer feels rushed—that's a surmountable problem to which composers have adjusted. The real problem is that the project is not getting the creative benefits of having early composer involvement.

The creative input composers can have on a project is directly proportional to their involvement in the creative process as a whole. Their ability to add depth, significance, viability, and overall relevance to a score is enhanced by a deeper understanding of the other elements in the picture. Understanding is increased by familiarity with story line, character development, and as much visual information as can be provided. (This is not to say that the composer needs a complete script—although that wouldn't hurt, either.)

One of the most helpful items is the storyboard. (Here, the saying becomes "A picture is worth several thousand notes.") Just as a storyboard provides every other participant in the project with valuable information, a storyboard provides the composer with cues that might not be obvious without visual information.

For example, suppose the composer is assigned the task of writing music for a villain. He spends a few days and hands it over for approval. It's villainous all right, with a slender, shaved-head, fast-on-his-feet type of character, right out of *Natural Born Killers.* The problem is that you, as the director, pictured an overweight, dark, mustached, slow-moving Mafioso-type villain.

You can accomplish some effective communication by providing detailed character descriptions. But a picture—better yet, a series of pictures—will put everyone on the same page. The composer will not only zero in on the villain, but will be able to foreshadow what's lurking around the corner. The difference is spectacular.

A composer who is supplied with only a single frame of an animation is a hundred times better off than a composer working "in the dark." That's what inspiration is all about. The more material supplied to the composer—notes from brainstorming sessions and creative planning meetings, character studies, rough outlines, first drafts, and so on—the more inspired the music will be. And, naturally, this implies that the composer is involved early on.

In the end, the music, interactive or not, will still have to move us.

The "New Paradigm" Syndrome

Of late, much has been said about "whole new paradigms" in music scoring for multimedia titles—more specifically, one "new model" involves generating interactive sound based on user actions. This theory goes on to state that music must now be thought of on a molecular or atomic level: individual musical phrases broken down into individual notes reduced to their even smaller sounds—maybe down to the sample level (a sound that's approximately 1/44,000 of a second long). These sounds could be randomly accessed in a nonlinear fashion, dependent upon user action and reaction at any given point in time. The old "linear" way of looking at things, so the theory goes, is dead and gone.

While there are some composers who have a grasp of what this implies, they're people who are gifted at the craft of scoring music in the *traditional* way as well. For example, Thomas Dolby, who is working on a tool known as the Audio Virtual Reality engine (AVRe) that is specific to this task, immediately comes to mind. Dolby has been on the "bleeding edge" for some time, but he's also a masterful composer who understands the need for emotional content in a music score.

There is still plenty to learn from the "old" way of doing things. Creating soundtracks strictly with computer-generated music—in other words, with algorithmic compositional tools that simply respond to commands such as "make it darker, lighter, exciting, nervous, random, Italian, mushy, tighter, bluer, fatter, turquoise,"—doesn't work. The music—and the listener—suffer from a lack of human artistry.

In many interactive games, there isn't enough synergy between visuals and the music. Before we look for new paradigms, which may or may not be beneficial, why not use some of the techniques that have made film and television music successful? First and foremost, you need a composer who understands how to derive emotional content from the simplest of scenes—even a DOS prompt, if necessary—and who has the talent to deliver the goods. If we were to diagram the problem as it currently exists in interactive music composition, it might look something like this:

- phase A: binky-boinky sounds that barely resemble music
- phase B: music with emotional content and relevance to the visual image
- phase C: technology that permits interactive musical responsiveness

The current problem is that many people think they can go directly from phase A to phase C, completely skipping phase B. And with this continuum, there has to be a thorough understanding of phase B, before phase C can be effectively developed.

If we begin to dissect existing media, the first thing we'll find is that titles consist of *scenes,* whether they are interactive or not. A scene has a mood or theme. It may last a few minutes or a few seconds, but it still needs an underlying score that conveys an emotion at a given point in time. For multimedia music to be successful, we must not throw out the baby with the bath water by skipping phase B. We shouldn't let the attractiveness of automated, computer-generated scoring lure us into a sense of false security. In the end, the music, interactive or not, will still have to move us.

A Note to Producers on Creative Direction

As stated in the discussion of supplying storyboards or other visuals, there are many things that the participants in a multimedia production can do to give creative direction to the composer. As a producer/director, you may not be familiar with the creative or technical terms that a composer might use, so use your own vocabulary. Cite examples, such as "Make it Eric Clapton–like, but with an ethereal background, like the score to the movie *Rush.*" Or "I'd like it to be high-intensity but very orchestral, like the score to *Speed.*" The trick here is to give some examples but not make them demands. When you're too definitive, you remove much of the creativity and originality that's more likely to occur if you are providing guidelines. Even better, say "Make it *Rush*-like with electric guitar, but go to town with it—give it your own spin. I'd also like to hear your ideas after you've spent a little time with it."

Another useful tip: Don't give any creative direction if you don't have any. It's better to be honest and say you're not sure what you want, than to give direction that could yield poor results. If you trust the instincts of the composer you're working with (and you should), you can ask her to come up with an idea or two for review.

Language plays a part in this as well. Many disciplines have their own vernacular, and musicians certainly have theirs. While the producer might tell the composer, "Give me something ethnic for the main title screen," the composer might say, "OK, this scenario takes place in the tropical rain forests. I'll give you something with Latin colorings, but I want the beat to be contemporary—rap grooves with a touch of industrial. Moody pads under an addictive melodic hook. Slap some Jan Hammer guitarish stuff over some screaming Nine Inch Nail–like slams. You know, funky, but not too. Modern, but not *post.*"

A producer who spoke this way might get some strange looks. For musicians, it's a way of life. So, without totally emulating musician-speak, try going for some color in describing the music.

Hollywood Influence—The Good and Bad News

Pull out the soapbox, please, because this is where I get to sound off to anyone who's hung in with me this far. I've got good news and bad news. The good news is that the Hollywood influence in interactive music is finally here. The bad news: the Hollywood influence in interactive music is finally here.

Now, I'm certainly not the first one to come up with this "Hollywood meets Silicon Valley" stuff—it's been discussed for a long time now. If you want some particularly good insight into this, read Ernest Adams's article in the *1994 Computer Game Developers' Conference Proceedings,* "Celluloid to Silicon: A Sermon for Newcomers from Hollywood." But I'm here to talk about the music aspect of the business, and I can't help but notice a few interesting things.

The good stuff? More experienced, more talented, more creative composers and musicians are getting involved in interactive entertainment. Record companies have formed interactive divisions.

Resources for the Interactive Music Industry

Apple Computer, Inc. has been working to make it easier for artists and music publishers to incorporate music into multimedia titles. Here's a list of tools, technologies, and support sources to help you with your music efforts.

- *Interactive music development tools.* Apple offers a variety of tools and technologies to help developers create and test enhanced audio CDs for computers based on the Mac OS and Windows systems. Tools include the QuickTime Music Toolkit and multisession driver software. In addition, Apple and Opcode Systems, Inc., recently announced Opcode's Open Music System (OMS), which will enable multimedia and music developers to seamlessly route QuickTime music tracks to external MIDI devices such as keyboards, synthesizers, effects processors, and drum machines. Apple expects OMS support to be included in a future version of QuickTime sometime before the end of 1995.
- *The Interactive Music Track (IMT).* Apple has launched a worldwide support program for musicians and music publishers creating multimedia content for the burgeoning interactive music market. The Interactive Music Track extends the well-established, 1,500-member Apple Multimedia Program (AMP) to musicians, providing multimedia and interactive music information, contacts, networking opportunities, and discounts on software and hardware. Members also receive Apple's QuickTime interactive music tools at no additional charge. There's no additional cost to be identified as a member of the IMT at the time you join AMP—just send a note to musicmaker@applelink.apple.com to be entered into the international IMT database. For more information or an application kit, call the Developer Hotline at 408-974-4897.
- *Increased online access to QuickTime resources.* QuickTime Online is a World Wide Web server that interactive music developers, consumers interested in music, and amateur movie makers all over the world can access. The server offers QuickTime 2.0 for both the Mac OS and Microsoft Windows, the latest interactive music information, music videos, and developer tools for downloading. The World Wide Web address for QuickTime Online is http://quicktime.apple.com.
- *The Enhanced CD Fact Book.* This little gem of a resource, written by Josh Warner, provides you with an overview of enhanced CD formats, advice on presenting project ideas to record labels, and technical tips on creating interactive music titles. This publication will be distributed to all Interactive Music Track AMP developers. If you need an extra copy, contact the Developer Hotline at 408-974-4897.
- *The QuickTime branding program.* Apple recently launched a branding program around QuickTime, including an advertising campaign featuring three advertisements targeted at interactive music enthusiasts and professional musicians.
- *The Interactive Audio SIG of the MMA.* To get information about joining the Interactive Audio Special Interest Group of the MIDI Manufacturers Association (a worldwide organization), send an e-mail message to MMA@earthlink.net.

Startup companies are producing music-based multimedia titles. Companies of all sizes are experimenting with enhanced CD formats—putting multimedia features into what would normally be standard audio CDs. Music, although still the ugly stepchild of multimedia production, is getting more attention and appreciation.

The bad stuff? A few, but certainly not all, of the more established composers don't have the vaguest idea of what it takes to compose for interactive media. (It's quite possible that Paul McCartney should *not* score your next project. Would he know how to squeeze the most out of an AIFF file?) Record companies have formed interactive divisions because *everyone else has.* And they are all too often baffled by what consumers want and what they will buy. Many producers are already talking about getting a name talent rather than one that's appropriate for the medium.

In addition, there are about a dozen different enhanced CD formats, making the Beta vs. VHS standards war look like child's play. And last, but not least (and possibly *most*), is the fact the music business is part of the multimedia business, and vice versa, bringing out the demons in both industries.

When I left Hollywood for Silicon Valley seven years ago, I thought I was leaving behind one of my favorite sayings: "*Music business* is an oxymoron." Implicit in this statement are the clichés we've all come to know and hate: "Yeah babe, beautiful. Your stuff is great. We'll work together. I'll have my people call your people, and they'll do lunch next week."

The problem is the way that some producers treat "talent." Many interactive multimedia companies have adopted a Hollywood practice of having a large number of composers bid on a project by creating a demo "on spec" (without pay), and without communicating details or the bid's final outcome.

Here's a slightly exaggerated scenario to help you understand the composer's point of view on this practice. Let's say a producer calls and says, "Yes, I heard your demo. You're obviously a professional with a long, successful track record. However, I have 18 composers who want this gig too. So, do me a demo of an opening title theme for *Walter Mitty Meets Godzilla,* and make it perfect. Read my mind, because I don't really know what I'm looking for. Then, submit your master-quality demo by, say, 2 P.M. today, and wait by the phone. I'll let you know tomorrow whether you got the gig or not."

Of course, the producer never calls back, you miss the gig, and you start to think you're the biggest no-talent, no-hope, sad sack in the whole universe. Soon after, you have identical experiences with nine other multimedia companies, who also love your initial demo and make you do new ones on spec. Finally, you're forced to start selling aluminum siding to make ends meet, which is what your mother said you should have done years ago anyway.

Finding and Evaluating Composers

Let me suggest some alternatives for finding and evaluating composers.

- *Maintain a file of composer demo tapes.* Use these tapes to evaluate composers, then narrow down your prospects to two or three before requesting them to create a sample track for your specific project. If possible, have them send a video of their work, so you can see how well they compose to visuals.
- *Reimburse bidding composers for their time.* Attract the best talent to your projects by offering a nominal reimbursement to the hand-picked composers who create demos for your project.
- *Use your word-of-mouth network.* One of the best ways to find good composers is through personal references from other multimedia producers.
- *Post upcoming projects online and at multimedia conferences.* Most major online services have music-related bulletin boards. And conferences such as the Computer Game Developers' Conference, InterMedia, and The Apple New Media Forum are attended by many talented composers with briefcases full of demo tapes.
- *Look for composers' ads in music and entertainment magazines.* Review music-related publications such as *Keyboard Magazine* and *The Hollywood Reporter,* since many interactive media composers run ads in them.

The struggling multimedia industry has neither the right nor the balance sheet to start talking like big-time Hollywood producers. Return phone calls. Be thoughtful and courteous. And help this industry adopt the best and not the worst of the Hollywood and Silicon Valley worlds.

The Sound of Silence

Observe a moment of silence. Turn down the volume on the TV set, the stereo, *the computer.* Imagine life without the music . . . without audio. Use this moment to measure the importance of sound as it relates to everything else. What is it like . . . this sound of silence? Use this moment to reflect on the emotional and business benefits of music. And call your favorite composer early and often. ♣

Bob Safir is the president of InterOctave (interoctav@aol.com) in San Jose, California. He's written soundtracks for Piper, *a title created by Splash Studios and distributed by Brøderbund; and for* Elroy's Toy, *an animation by Third Planet Productions, to name a few. He's also a Macintosh user, co-chairman of the MMA Interactive Audio SIG, and the MusicMedia columnist for* Keyboard Magazine. *(Portions of this article are reprinted from the copyrighted* Computer Game Developers' Conference Proceedings *with permission.)*

Listings

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 the relevant Internet location. 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 Web sites maintained by Apple Computer.

http://www.info.apple.com/dev/
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.

As do most of the Apple-sponsored Web pages, this page contains a link back to the Apple Computer home page (described next) as well as a link to the Apple Internet Servers page (described later in this feature), both of which we think you'll want to check out.

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. If you visit the Apple Developer Services and Products page first, you can get to the Apple Computer Home Page by clicking the Apple logo on the left of the navigation bar near the bottom of the Developer Services page. If you're a developer, this page is less specifically oriented to your needs, but it's here that you can find out about everything that's going on at Apple, including a complete listing of Apple products, competitive information, news and information about the latest Apple products, and even an occasional special announcement from Apple executives.

http://www.info.apple.com/dev/thirdparty/third_party.html
This is the location of Apple's Web page listing third-party hardware and software products. Check out what other people are doing, and fill out the form at the site to have your own products listed, too.

http://www.austin.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 software updates, examples, demos, documentation, tools, and other items to help you develop new Mac OS products.

http://www.info.apple.com/opendoc/
This is the site of Apple's OpenDoc home page, 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://www.info.apple.com/qd3d/
Apple's QuickDraw 3D Home Page contains everything you need to know about QuickDraw 3D (QD3D), including QD3D applications you can "test drive," listings of developers who support QD3D with links to their Web pages, and thorough developer information.

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. The first contains the Power Macintosh Support page, with customer support information, including a list of frequently asked customer support questions (and their answers) and a listing of new Power Macintosh software from Apple. The second contains Apple's Power Macintosh home page, which includes the Power Macintosh White Paper, the current list of shipping PowerPC "native" products, Power Macintosh news, and links to non-Apple sites with information about the PowerPC processor and PowerPC processor–based systems.

http://quicktime.apple.com
This site contains the QuickTime Continuum page with news and technical and marketing information about QuickTime. This page focuses on hot new QuickTime developments, which means that much of its content is currently dedicated to QuickTime VR—including some cool QuickTime VR scenes, such as Chris Isaak performing at Bimbo's nightclub in San Francisco.

http://qtvr.quicktime.apple.com
This is the location of the QuickTime VR page, which includes not only information about how QuickTime VR works and how you can get started programming with it, but also QuickTime VR movie samples, which you can view by downloading the QuickTime VR player from the page.

http://www.info.apple.com/gx/gx.html
This site contains the QuickDraw GX 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—even entertaining.

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. Prepared by Michael Yee, a current Sun (and former Apple) engineer with way too much time on his hands, the ULTIMATE Macintosh page has 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. While you're at it, you might want to check out Michael Yee's home page (there's a link to it at the top of the ULTIMATE Macintosh page), where you'll find listings of other interesting places to visit on the 'net.

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

Nathan's Everything Macintosh page is another treasure trove of Macintosh information, maintained by Nathan Raymond, who's apparently sacrificing his own sanity so that others may live in peace on the Internet. The Everything Macintosh page 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. Once you've seen the Everything Macintosh page, you may also want to visit Nathan's home page at World Wide Web location http://www.cs.brandeis.edu:80/~xray/, which includes the so-called Comprehensive Image Map of Everything, a graphical index of the World Wide Web complete with hypertext links to hundreds of other locations.

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 Metrowerks's Web site, with information about its CodeWarrior PowerPC development tool.

http://www.hotwired.com/

Not much about Apple here, but we thought you'd have fun at the *Wired* magazine Web site. Before you get in, you have to go through a fairly simple (no-UNIX-required) registration prodedure.

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! ♣

Know of a particularly useful site? Whether it's a Web page, an FTP site, or a newsgroup, let us know about it and we'll consider adding it to this feature next month. Send your suggestions to a.directions@apple.applelink.com.

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