

AppleDirections

CONTENTS	
Amelio Announces Apple Strategy at WWDC	1
Strategy Mosaic: Why MacOS 8 is Important, Part 2	1
Editor's Note: Writing Assignment	2
IndustryWatch	3
Apple's New Organization	11
Apple Commits \$20 Million to Assist Developers' Marketing Efforts	12
Study Confirms Macintosh Superiority in Ease of Use and Customer Satisfaction	13
WWDC Wrap-Up	13
CD Highlights: System Software/SDK Edition, July 1996	16
Inside Apple Game Sprockets	17
Human Interface: Now It Can Be Told	18
OpenDoc Human Interface FAQs	25
Finding and Hiring Great Macintosh Programmers	28
It Shipped!	33
Developer University Schedule	35
Apple Internet Page	36
Next Month: Making the transition to OpenDoc	

APPLE NEWS

Amelio Announces Apple Strategy at WWDC

Delivering on his promise to announce his strategy for Apple Computer, Inc., in May, Dr. Gil Amelio unveiled his plan before 4,000-plus developers on May 13 during the opening session of Apple's Worldwide Developers Conference (WWDC). Later in the month, Apple announced the formation of a new corporate organization to implement the strategy; the organization is described in "Apple's New Organization," immediately following this story.

Stressing his commitment to developers by first announcing his plans at WWDC, Dr. Amelio described a corporate strategy that will position Apple in front of what he considers to be the industry's key "megatrends": convergence of computers, communications, and consumer electronics; explosive growth in Internet use; and the demand for media-rich digital content. He expressed Apple's intent to make the Internet central to the company's strategy.

Among the specifics of his address were the following points:

- Dr. Amelio expects the company to regain profitability within one year, but said he thinks complete transformation will take up to three years.
- Apple will simplify its product lines to lower research and development costs and make customers' buying decisions easier.
- Parts of Mac OS 8, the next major version of the Mac OS previously referred to as

please turn to page 11

STRATEGY MOSAIC

Why Mac OS 8 Is Important

By Gregg Williams, Apple Directions staff

Part 2: Simplified Programming Model and New Opportunities

In last month's Strategy Mosaic, I talked about two reasons why Mac OS 8 is important to you:

- Mac OS 8 provides a high level of backward compatibility with System 7 applications and hardware, making it possible for both you and your customers to go forward without losing your investment in current technology.
- A number of Mac OS 8 technologies provide the user benefits of increased performance, increased stability, and greater ease of use. These technologies provide you with an environment in which your application can perform better and give your customers a higher-quality experience.

In addition, I also gave an overview of the System 7 and Mac OS 8 program architectures, to give you an idea of what opportunities Mac OS 8 offers for writing new kinds of software.

This month, I'll finish this two-part article by discussing the two most important reasons for adopting Mac OS 8:

- Apple has streamlined and enhanced the Mac OS 8 programming model to simplify the job of writing software for the Mac OS platform and make you more productive in your programming.
- You can use a number of Mac OS 8 technologies to create compelling, leading-edge solutions that would be difficult or impossible to implement under System 7.

please turn to page 5

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/Graphic Designer

Lisa Ferdinandsen (LISAFERD)

Contributors

Peter Bickford, Dave Curbow, Alex Doshier, Elizabeth Dykstra-Erickson, Robert Megantz, Kerry Ortega

Manager, Developer Press

Dennis Matthews

Manager, Apple Developer Periodicals

Mark Bloomquist

Production Manager

Diane Wilcox

Prep and Print

Consolidated Publications, Inc., Sunnyvale, CA

© 1996 Apple Computer, Inc., 1 Infinite Loop, Cupertino, CA 95014, 408-996-1010. All rights reserved.

Apple, the Apple logo, AppleLink, AppleScript, AppleSoft, AppleTalk, ColorSync, Firewire, HyperCard, Mac, MacApp, Macintosh, MacTCP, Newton, OpenDoc, Pippin, PowerBook, Power Macintosh, PowerTalk, QuickTake, QuickTime, and WorldScript are trademarks of Apple Computer, Inc., registered in the U.S. and other countries. Balloon Help, Cyberdog, Finder, QuickDraw, and Sound Manager are trademarks of Apple Computer, Inc. Adobe, Acrobat, Illustrator, Photoshop, PostScript, and Premiere are trademarks of Adobe Systems Incorporated or its subsidiaries and may be registered in certain jurisdictions. Java and other Java-based names are trademarks of Sun Microsystems, Inc., and refer to Sun's Java-based technologies. Netscape, Netscape Navigator, and Netscape Live3D are trademarks of Netscape Communications Corporation. NuBus is a trademark of Texas Instruments. PowerPC is a trademark of International Business Machines Corporation, used under license therefrom. RealAudio is a trademark of Progressive Networks, Inc. SOM is a licensed trademark of IBM Corporation. UNIX is a registered trademark of Novell, Inc. in the United States and other countries, licensed exclusively through X/Open Company, Ltd. All other trademarks are the property of their respective owners.

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



Why I Prefer Macintosh

It's probably been a while since anyone gave you a writing assignment, and I hope you won't be shocked to find that I'm giving you one this month. We're initiating a new column in *Apple Directions* called "Why I Prefer Macintosh," and it's going to be written by you.

Relax. Unlike a term paper, project status review, or even your kid's book report (which you aren't supposed to be working on anyway!), there's no due date for this assignment. You can take your time with it. But, for the sake of the Mac OS platform, please do it.

What I'm assigning you is this: In approximately 250 words (fewer, if you can—but in the words of Voltaire, "I would have written less if I'd had more time"), answer the following question:

Why do you prefer to develop on/for the Macintosh computer instead of a Windows-based PC?

Once you're done, e-mail your answers to me at dreyfusp@apple.com. The *Apple Directions* staff and I will edit your answers and publish them in our new column. (We obviously can't print flames, so please don't send them.)

Here's why we're doing this: One of *Apple Directions'* purposes is to spread the good word about developing on and for the Mac OS platform. Each month, we tell you about part of the Mac OS story: We describe the technical and business advantages of adopting a new technology in your product, tell you why and how Apple strategic directions benefit you, and let you know why—in general and specific terms—developing on and for the Macintosh computer is a solid business proposition.

We work for Apple Computer, Inc., so we can only go so far by ourselves in telling that story, which is why I'm asking for your help. You have the experience and the credibility to say why the Macintosh is a great development platform—and to say it convincingly.

Here's what one developer, Kim Hunter of Acrobytes Software in Mission Viejo, California, already wrote on the subject:

I can't enter your contest because although I wrote lots of Macintosh software, I have yet to write one line of Windows software. On the other hand, perhaps that is exactly what you're looking for!

I write Macintosh software without thinking. I use Symantec's C compiler, which is an

incredible dream compared with other compilers. Once in a while I have to thumb through Inside Macintosh, or Scott Knaster's books. That's how I wrote the original Seiko Smart Label Printer software; it's also how I helped Claris develop the FileMaker translator for Q&A. I wrote the Q&A translator software on a Macintosh computer, then verified it on a PC before shipping it to Claris.

When I set out to program on a PC, I found that the one I had was too slow and didn't have enough RAM or hard disk space to operate all the systems and the compilers. Also, Windows 95 wouldn't recognize my floppy disk, my mouse, or my video board. I finally gave up with upgrades and decided to get a new machine. I bought a system with a Pentium 133 MHz chip, but the mouse still didn't work, so I took it back. Tonight I'm installing the new beta version of Windows NT 4.0, then I'll install Windows 3.11, and, after that, Windows 95. The dealer had trouble with the new Intel Atlantis chip set motherboard. It was the first one they had installed, so they're anxious to know how I get along with it and if Windows NT 4.0 will recognize all the hardware. I was kind of surprised because I thought the PC was a "standard," sort of a "commodity" product.

So, any day now I may actually start writing Windows software.

Please join Kim in saying why—technically, financially, or both—you prefer the Mac OS as your development platform. The answers we publish will be read by the 15,000-plus developers around the world who receive *Apple Directions*; they'll also be posted on the Apple developer Web site, currently visited by 25,000 people each week. You'll potentially be playing an influential part in spreading the good news about how and why Macintosh development is thriving.

Finish your writing "assignment" as soon as you can, and e-mail it to me. Your words could help convince developers who haven't yet experienced the joys of Macintosh to join the cause.

Paul Dreyfus

Editor

dreyfus.p@apple.com

PowerPC 603 and PowerPC 604 Chips to Hit 300 MHz in 1997

The PowerPC Alliance recently committed to driving the performance of both the PowerPC 603 and 604 families to over 300 MHz in 1997. The 603e processor, acknowledged by the industry to deliver Pentium-like performance, got off to a slow start but now looks like it's ready to move ahead quickly. The 604 chip already performs about 30 percent faster than a Pentium Pro of identical clock speed, based on independent benchmarks such as those conducted by *BYTE* magazine. It will be especially formidable once it hits these high clock speeds.

Implications/Opinions: When it was first introduced, the PowerPC RISC-based microprocessor showed promise as the firm hardware foundation for the future, one that would deliver higher performance more efficiently than the CISC x86-based architecture. These performance increases show that the PowerPC Alliance is delivering on that promise. Apple, Motorola, IBM, and other Mac OS licensees will be moving aggressively to build systems based on the increasingly powerful PowerPC processors, giving you more reasons than ever to develop software that lets Apple's and your mutual customers take advantage of RISC performance. It's up to all of us to communicate the advantages of PowerPC processor-based hardware and software solutions over Wintel computers.

CHRP Won't Support OS/2

Sometime in the past few months, IBM made the very quiet decision that OS/2 will no longer be one of the operating systems supported by the Common Hardware Reference Platform, or CHRP. CHRP is the specification

for PowerPC processor-based personal computers that will run a variety of operating systems, including Mac OS, Windows NT, AIX, and Solaris.

Implications/Opinions: By dropping OS/2 from CHRP and licensing the Mac OS for its own CHRP-based systems, IBM is poised to make a substantial effort to market the Mac OS. This can only result in an increase in Mac OS market share, and more customers for your products.

Windows 95 Adoption Slower Than Expected

The April 1, 1996, issue of *Computer Reseller News* reported that only 25 percent of customers choose Windows 95 over Windows 3.1 in PCs that provide the option of installing either version of the operating system. This number is drastically lower than the 50 percent adoption rate that most analysts had expected by now. What's more, some software developers are experiencing slow sales of Windows 95 applications, according to the report; at least two—Touchstone Software Corporation and Corel Corporation—reported losses because of lower-than-expected sales of their Windows 95 products.

Windows 95 adoption appears to be especially slow in the business market, where many customers appear to be waiting for the next release of Windows NT. According to a poll conducted for *Computer Reseller News* by The Gallup Organization, 3 percent of corporate buyers for Fortune 1000-type companies said they currently use Windows 95; among smaller and midsize businesses, that percentage rises, but only to 13 percent.

The first beta release of Windows NT 4.0 took place in February, and final release is expected in the third quarter of calendar 1996. If the final version lives up to expectations, many business customers will likely bypass Windows 95 and move directly to Windows NT 4.0. As a result, forecasts for sales of Windows NT in 1996 have climbed. Dataquest predicted in March 1995 that 4 million copies of Windows NT would sell in 1996; Dataquest recently revised that forecast to 8 million copies.

Implications/Opinions: When Windows 95 was first released last August, we reported that Microsoft was already downplaying its significance, and that Bill Gates himself said Microsoft would focus on Windows NT. (See both Strategy Mosaic and IndustryWatch in the September 1995 issue of *Apple Directions*.) In many ways, the market is performing just as Microsoft—and many others—expected it would, after the myriad problems with the Windows 95 upgrade became apparent.

The shift in the Wintel business market to Windows NT represents another opportunity for the Mac OS platform to attract switchers. Yes, Windows NT delivers multitasking and stability—which are being built into the next version of the Mac OS, Mac OS 8—but it's harder to use than Windows 95, which means that it's even harder to use than the Mac OS. Also, it's not fully backward-compatible with other versions of Windows, and it does not include plug-and-play capabilities, presenting yet another upgrade challenge for Wintel customers.

IBM to Deliver OpenDoc for Windows NT and Windows 95 This Year

IBM announced that it will begin shipping beta versions of OpenDoc for Windows NT and Windows 95 in June 1996; final versions will follow in the fourth quarter. The company has already shipped more than 600 alpha-level developer toolkits. IBM previously made OpenDoc for OS/2 Warp Version 3.0 generally

August *Apple Directions* Online

August's *Apple Directions* will be available by July 15 at the following locations:

- AppleLink path—Developer Support:Developer Services:Periodicals:Apple Directions
- Internet—<http://dev.info.apple.com/appledirections/adtoc.html>

available and is shipping a beta version of OpenDoc for AIX. General availability of OpenDoc for AIX is planned for the third quarter of this year. IBM's OpenDoc facility for distributing objects, the System Object Model (SOM™), is also available on IBM's MVS enterprise operating system and is in beta on the OS/400 system.

Implications/Opinions: By throwing its engineering weight behind OpenDoc on multiple platforms—including Mac OS and Windows—IBM is assuring that OpenDoc will become the only true cross-platform component architecture. Microsoft's competing OLE architecture fully supports only Windows. The notion that you could develop a single OpenDoc component and have it run on all popular personal computing platforms will soon be a reality, guaranteeing that you can leverage your OpenDoc development efforts into a single product that can be marketed to all the major platforms. We think this is a very solid reason to get going with OpenDoc software, if you haven't already.

VRML 2.0 Specification to Include Apple 3DMF Format

VRML, which stands for Virtual Reality Modeling Language, is the specification for 3D and other multimedia content on the World Wide Web. The VRML Architecture Group (VAG) has announced that it will base the next version of the specification—VRML 2.0—on the Moving Worlds proposal originally created by Silicon Graphics and supported by over 50 vendors, including Apple, Netscape, IBM, Sony, and Adobe. A modified version of the QuickDraw 3DMF (3D metafile) format was chosen as the binary file format for the Moving Worlds specification. In making its decision, the group rejected Active VRML, Microsoft's alternative proposal.

Moving Worlds will enhance VRML so that it provides object behaviors, audio, Java links, and other types of multiuser interaction and animation. The interactivity employed in Moving Worlds is based on Sun Microsystems' Java and JavaScript programming languages. VRML 2.0 will be an enabler for richer Internet-based multimedia plug-ins used for everything from games to aerodynamics modeling. Netscape has already announced that Moving Worlds will be the basis for its Live3D platform.

Implications/Opinions: VAG's decision represents a significant victory for Apple technology and keeps Apple front and center in determining the format of multimedia Web content. Despite losing a round in the battle to determine Web multimedia standards, though, Microsoft has not given up on Active VRML, the company's proprietary standard for interactive multimedia Web content. Some independent vendors have suggested that they may develop technologies that support both Moving Worlds and Active VRML. Microsoft has clearly not conceded: in response to VAG's decision, the company announced, "If a lot of people start building for [VRML 2.0], we'll support it."

It remains to be seen whether the market will choose an open standard supported by most of the industry's key players, or a proprietary standard pushed primarily by Microsoft. But we can tell you which way we'd like the market to go.

Asymmetric Digital Subscriber Line: High-Speed Network Connection of the Future?

Speaking of multimedia content on the 'net: To meet increased demand for high-speed Internet connections for viewing said content—and to try and head off competition from the cable TV industry—telephone companies have dusted off Asymmetric Digital Subscriber Line (ADSL) as an alternative to ISDN. ADSL is an old standard originally developed for video-on-demand applications, and it's now being repositioned as an Internet connection for home customers.

Designed to work on existing telephone service lines, ADSL doesn't interfere with conversation because it uses a higher frequency than the one used for voice, thus allowing for continued use of normal telephone service. ADSL's accelerated speeds—6 MB per second (Mbps) to the home and 640 KB per second (Kbps) from the home—are comparable to the 10-Mbps rate of common Ethernet. That's fast compared to the pedestrian 28.8-Kbps speed of today's standard modems and even ISDN's 128 Kbps.

Implications/Opinions: Clearly, ADSL is a technology to watch. It is being reintroduced at an opportune time: Everyone wants to use the 'net, yet bandwidth limitations prevent

most home users from taking advantage of its multimedia offerings, such as those created with Shockwave and Moving Worlds. ADSL's high speed may finally open up Internet multimedia for home users. If you've built Web sites using QuickTime VR or any other Apple multimedia technologies, that's good news for you, since ADSL lets home users fully experience such multimedia Internet content, such as the QuickTime VR BMW site (<http://www.bmwusa.com/>).

Yet for all its promise, ADSL is not a magic cure for the Internet's bandwidth ailments. There are other gating items that affect Internet access speed, such as the bandwidth limitations of the service provider and content provider. Furthermore, the standard's asymmetric data flow—that is, different speeds to and from the home—is not optimal for all applications. It's true that most Internet users receive much more data than they send, which means that the ADSL incoming speed of 6 Mbps will meet most people's needs. However, videoconferencing works best with symmetric speeds, while multimedia publishing prefers asymmetric data speeds that are the reverse of ADSL's. Finally, the current U.S. \$1,000 price tag of an ADSL modem puts it beyond the reach of most households, despite the potentially competitive monthly fees that may be charged for service. U.S. West has indicated that it will likely price ADSL service at U.S. \$35 to \$40 per month (which compares well to current ISDN rates of \$25–50 per month). Yet, while there are more than 30 field trials in progress, no telephone company has yet to establish and approve standard ADSL rates.

The bottom line is that with potentially competing standards for high-speed home Internet connections, it won't be so long before the majority of consumer computer users will have access to the full array of online multimedia content. If the telephone companies decide, for example, that the business case for ADSL is good, little can stand in the way of a fast roll-out. Any new Internet products and content should be built with the multimedia future strongly in mind, unless they're to lose pace with the rest of the industry's offerings. ♣

Why Mac OS 8 Is Important

continued from page 1

Ease of Development Under Mac OS 8

The design of the original Macintosh computer was severely constrained by the minuscule amount of memory—128K—that it contained. This forced the computer's designers to make several decisions that got the maximum use out of existing memory, but at the expense of complicating the programming model. (For example, because the Memory Manager is not reentrant, various interrupt-level services—including vertical-blanking routines—cannot call any routines that move memory. This is one of many examples of how the current Mac OS programming model includes multiple types of software, each of which has special rules on how it must be written.)

As time passed, Apple engineers found ways to make Macintosh computers do new things, but these solutions were built on top of different parts of the Macintosh architecture. One example of this is that the current Mac OS uses at least three different mechanisms to implement shared libraries—the QuickTime Component Manager, the Apple Shared Library Manager, and the Code Fragment Manager, each of which is meant to be used in a different context. In this and many other situations, the result is that developers must understand multiple technologies to determine which to use. This has made the overall task of creating Mac OS-compatible software more complex than it needs to be.

Mac OS 8 represents Apple's first comprehensive attempt to streamline the Macintosh programming model. In doing so, *Mac OS 8 decreases the number of programming models you have to deal with and the number of special rules you have to remember to create Mac OS-compatible software.*

The end result is that the Mac OS 8 programming model is simpler and easier to remember than today's Mac OS. Granted, if you already know how to write classic Mac OS applications, you will have to learn how to make the best use of Mac OS 8. However, once you've done that, you will have less to remember than you do today; you will also find that Mac OS 8 does a lot of things for you that you formerly had to program yourself.

How Mac OS 8 Helps You Create Software

Mac OS 8 does several things to help you create software. For one, Mac OS 8 Toolbox managers have been simplified to provide a better programming model that shields you from direct manipulation of data structures. Not only does this allow you to think and program at a higher, more abstract level, but it also frees Apple engineers to change underlying system details without worrying about breaking the code you've written. For example, the Mac OS 8 File Manager is more powerful than that of System 7, but its API (application programming interface) is about half the size—which also makes it easier for you to understand and use.

A second way that Mac OS 8 helps you create software is that the Mac OS 8 Toolbox managers make it much easier to implement features that, today, you have to write, debug, and maintain yourself. (Examples of this include "sliders" and other controls, tear-off menus, dialog boxes, and scrolling panels.) This change allows you to spend less time on implementation details and more time on developing a superior product; in addition, it increases system stability by enabling all developers to use the same standard, tested code to implement a given feature.

A third way that Mac OS 8 helps you create software is by ensuring that certain technologies—among them, OpenDoc, Open Transport, QuickTime, QuickDraw GX, the QuickDraw GX line layout API, and WorldScript—are always present. In some cases under System 7, you could not be sure that a given technology would be present on every computer, so you had to write two separate code paths to accomplish the same task using two different technologies.

That changes with Mac OS 8. Since certain technologies are integrated into Mac OS 8, there are no multiple code paths to clutter your source code, and you don't have to implement the same functionality using two different technologies.

Simplifying Your Programming

Perhaps the most important way that Mac OS 8 helps you create software is by simplifying the number of programming models you have to know to create Mac OS-compatible software. Here are the major examples:

- *User-interactive software.* Under System 7, you have several ways to create user-interactive

software: applications, OpenDoc parts, desk accessories, and control panels, each with a different programming model that you have to remember how to use. In Mac OS 8, Apple has reduced these to two programming models—the Mac OS 8 application and OpenDoc parts—which you can use for everything from the smallest utility to the largest customer solution.

- *Shared libraries.* System 7 software can use the Apple Shared Library Manager (ASLM), the QuickTime Component Manager, or the Code Fragment Manager (CFM) to encapsulate code meant to be loaded dynamically by multiple programs. Apple designed Mac OS 8 to standardize on the Code Fragment Manager alone for shared libraries. (For backward compatibility reasons, Mac OS 8 continues to support the Component Manager, but only for multimedia-related uses.)

- *New driver model.* The new Mac OS 8 I/O architecture replaces two separate driver models, the older NuBus™ model and the newer PCI model, with a single, new I/O architecture. In many cases, you can write the driver for a PCI card in such a way that it will continue to work under Mac OS 8. (See *Designing PCI Cards and Drivers for Power Macintosh Computers* for details. This book is available from Addison-Wesley or through the Apple Developer Catalog. You can also find it on disc 2 of the April 1996 Mac OS SDK CD, using the pathname Dev.CD Apr 96 SDK2:Development Kits (Disc 2):PCI Driver Development Kit. • Documentation:Designing PCI Cards&Drivers f.)

- *Networking.* Under Mac OS 8, Open Transport replaces AppleTalk, MacTCP, and the System 7 Serial Driver as the preferred method for dealing with network protocols and doing serial communications. Open Transport gives you a single API for implementing any networking protocol that is available on Mac OS-compatible computers. (See the section on Open Transport at the end of this article for more details.)

- *Less restrictive programming of specialized routines.* Under System 7, several important routines—including VBL tasks, I/O completion routines, and Time Manager tasks—run either at hardware interrupt level (where routines can't call the Toolbox and can access only memory that is physically resident) or as deferred tasks. This changes under Mac OS 8, where most such routines run at the task (as opposed to the interrupt)

level. This makes it easier for you to write such routines, because you can write code without worrying about whether it will generate page faults. (Under Mac OS 8, such routines work correctly with Virtual Memory, and page faults can occur without disrupting them.)

- *Simpler human interface elements.* Mac OS 8 makes it easier for you to add text, icons, lists, PICT graphics, buttons, pop-up menus, and other human-interface elements to your software. It does this by encapsulating them in SOM (System Object Model) objects called HI objects. The object-oriented nature of HI objects allows you to customize them by creating new *HI objects* that inherit from existing ones.

You can and should use HI objects anywhere within your application to create its human interface. Mac OS 8 includes HI objects that implement windows, lists, editable text areas, menus, ten different kinds of controls (including buttons, checkboxes, scroll bars, and disclosure triangles), dialog boxes, scrolling panels, grouped radio buttons, visual separators, and static panels for icons, pictures, and text.

Certain of these HI objects implement controls, lists, menus, and windows. These HI objects simplify your programming model by eliminating the need for System 7 CDEFs, LDEFs, MDEFs, and WDEFs (the definition procedures in System 7 that draw controls, lists, menus, and windows). Instead of using a System 7 definition procedure, you get the behavior you want in Mac OS 8 by subclassing the appropriate HI object and overriding its response to a user event. In addition, you can install event handlers that temporarily or permanently override the default response of a process (task) or window to a user event.

- *Simpler, more powerful drawing, typography, and printing.* In Mac OS 8, QuickDraw GX is always present and your application suffers no memory-usage penalty for the parts of QuickDraw GX it does not use. Because of these two facts, adopting the QuickDraw GX technologies that make sense for you will simplify your programming effort. In addition, by using QuickDraw GX, you will find it much easier to adapt your applications to international markets. (This article covers QuickDraw GX in more depth in the section “The Mac OS 8 Graphics Architecture” on page 8.)

Better I/O Architecture

If you write device drivers for Mac OS—compatible peripherals, Mac OS 8 will significantly simplify the process of doing so. System 7

device drivers will not work with Mac OS 8, so Apple is already working to help developers revise their drivers to work with Mac OS 8. (However, PCI-card drivers written to the specifications outlined in *Designing PCI Cards and Drivers for Power Macintosh Computers* will work with Mac OS 8.)

Apple wants to include as many drivers as possible with the first commercial release of Mac OS 8 to ensure that customers' third-party hardware investments will continue to work after they have installed Mac OS 8. The company is currently working with hundreds of developers to ensure that over 1,500 drivers will be ready for the initial commercial release of Mac OS 8.

As I stated in last month's column, Apple decided to “break” the existing I/O architecture to increase the stability and performance of device I/O and to make it easier for you to create device drivers by simplifying their programming model.

Under System 7, all device drivers were written to the same basic API, which was not a good fit to any one type of device. Mac OS 8 improves upon the System 7 situation with the addition of the driver family, which provides common services to devices that have common characteristics. Among the driver families included in Mac OS 8 are SCSI, display, sound, pointing, block-storage, and Open Transport devices.

The Mac OS 8 I/O architecture makes writing device drivers easier in several ways:

- Under Mac OS 8, drivers are implemented as Code Fragment Manager fragments (shared libraries) called *plug-ins*, which are always associated with a specific I/O family. Plug-ins must be written to that family's programming interface and implementation guidelines. The I/O family usually includes libraries of commonly used routines; these routines will often simplify your development efforts.

- Because Mac OS 8 device drivers are implemented as Code Fragment Manager fragments, they can be written in a high-level language without assembly-language headers. This gives you more flexibility and enables you to be more productive while coding.

- Mac OS 8 includes a database of system information called the *name registry*; the device portion of the name registry describes the configuration and connectivity of all the hardware in the system. Under System 7, you had to follow different rules to get configuration information on different types of

devices—NuBus, PCI, and PC card devices, for example. Mac OS 8 simplifies your driver programming effort by giving you one place to look for such information.

- Apple has been able to design the programming interfaces for plug-ins so that they will not change in future releases of the Mac OS. The stability of the plug-in programming interface will help ensure the portability of Mac OS 8 drivers in future releases of the Mac OS.

For more on Mac OS 8 drivers, see “Copland Drivers—Time to Put the Pedal to the Metal,” on page 17 of the December 1995 issue of *Apple Directions* (<http://dev.info.apple.com/appledirections/dec95/copland.html>). See also the important erratum on page 36 of the January 1996 issue (<http://dev.info.apple.com/appledirections/jan96/errata.html>).

Better Extensibility

Before Mac OS 8, you were often forced to use various patches to extend the computer in certain ways. Not only was this tricky (yet another programming model you had to master), but the result was potentially fragile when combined with other extensions that were loaded on the same computer.

Mac OS 8 addresses the topic of extensibility in two ways. First, many system services have significant extensibility built into them, so you usually don't have to patch system software to get the features you need. (This planned-for extensibility is also far more stable than System 7 patch-based extensions are.)

Apple also provides a second level of extensibility through a Patch Manager that allows you to patch the system. The Patch Manager allows you to name, order, enable, and disable patches, and it allows Mac OS 8 to determine where conflicts might arise between patches. However, the Patch Manager is an absolute last resort, and Apple recommends that you not use it unless it is the only way to get your task accomplished.

New Development Opportunities

So far, I've talked mostly about how, by creating software that runs in the Mac OS 8 environment, you can help deliver several key benefits to your customers—namely, higher performance, a more stable environment, and software that is easier to use. Now it's time to take a look at Mac OS 8 technologies that allow you to add features to your products

that would have been difficult or impossible to add before Mac OS 8.

Adding Preemptively Scheduled Tasks

In Mac OS 8, every application participates in preemptive multitasking. However, it's possible to increase your application's performance by adding multiple preemptively scheduled tasks to your application. Most of you will add preemptively scheduled tasks *within* a cooperative program. But, in situations where it makes sense to do so, you can also add server programs—preemptively scheduled tasks working in their own protected address spaces—that run in conjunction with a cooperative program. (For details, see the first part of this article in last month's issue of *Apple Directions*.)

To many, the phrase "increased performance" is simply another way of saying "runs faster." But increased performance is more than faster execution. True, an optimal partitioning of your program into separate tasks can make it possible for some code to do useful work when, under System 7, your application would be idling—for example, while waiting for user input or a slow mass-storage device. This can certainly make your application run faster.

However, by putting critical code in a task that is assigned a high priority, you can make your application seem more responsive to your users' needs. For example, by having separate tasks for the real-time playback of video and the application's human interface, you can create an application that—in the user's eyes—delivers both smooth video playback and on-screen controls that are immediately responsive. In this way, you can improve your application's performance by increasing the *quality* of the interaction between your customers and your application.

To sum up, you can use multiple preemptively scheduled tasks to increase your application's overall performance, and this increase can take two forms: increased speed and higher quality.

Assisting the User

Two of the goals for Mac OS 8 are to make Mac OS-compatible computers simultaneously more powerful and easier to use. You can participate in fulfilling that goal—and, in doing so, create state-of-the-art applications that your customers will want to buy—by adding user-assistance features to your applications.

Mac OS 8 includes some exciting new technologies that will make it possible for your

application to take a more active role in helping your customers get their work done. The following paragraphs describe these technologies. (For more details on these and other user-experience technologies, see "Looking Forward to the Copland User Experience," in

Mac OS 8 decreases the number of programming models you have to deal with and the number of special rules you have to remember

the April 1996 issue of *Apple Directions*, also on the Web at <http://dev.info.apple.com/appledirections/apr96/stratmos.html>).

- *Tips*. This is a new feature that allows users to call up a Tips window, which includes a list of tips likely to be relevant to what the user is currently doing. When users click on a tip, they get a screen of information telling them how to use their Mac OS-compatible computer more efficiently. (For example, a tip might tell the user about an easy way to start up from an external hard disk or close all open folders.)

You can enhance your application by adding application-specific tips to the Tips window. With additional programming on your part, you can also make it possible for your application to monitor a user's behavior and (if the user desires it) automatically display a relevant tip—a feature that will impress many users.

- *Delegating work*. Mac OS 8 gives you the building blocks to do *delegation*, which enables the computer to do work on the user's behalf, even when the user isn't present. One example of delegated work might be to have the computer do a hard-disk backup automatically every night and record the results of the backup to a log file. You should be thinking about which of your application's tasks you might want to make into delegated tasks.

- *Experts*. In contrast to Apple Guide (which users invoke to learn how to do a well-defined task), *experts* are used to achieve a desired goal without teaching the user how to perform the work needed to reach the goal. (Often, an expert will ask the user high-level questions about what the user wants to accomplish. The expert will then figure out what needs to be done and, with the user's permission, do the work automatically.) One example of an expert that Apple will include with Mac OS 8 is the Setup Expert. This expert will help users set up a new Mac OS-compatible computer, without them needing to know about the various control panels involved and without forcing them to actually make the necessary changes manually.

Apple encourages you to create your own experts; to help in this, Apple engineers are creating tools to make it easy for you to do so. Experts give you a new way to isolate your customers from complexity and streamline your application's design. In turn, your customers will be more productive and will regard your application as a "power tool" that helps them be successful.

- *Balloons*. You're probably familiar with balloons, which allow users to point to an interface element (a close box, window panel, or icon, for example) and get a short description of the element. (In System 7, balloons are known as *Balloon Help*.) Mac OS 8 will continue to support balloons.

Furthermore, Mac OS 8 will make balloons easier to use. Under System 7, users have to go to the Help menu on the menu bar to turn balloons on and off, an inconvenient process that discourages many users from taking advantage of them. Mac OS 8 includes a feature called *contextual menus*, which makes it easy for users to display a context-sensitive menu of possible actions relating to the item being pointed to. Users can invoke an item's balloon from a contextual menu, thus allowing them to get help on a single item more conveniently than is possible under System 7.

- *Apple Guide*. This is another technology present in System 7. Apple Guide allows users to get step-by-step instructions on how to perform certain tasks. Mac OS-compatible computers include files (called *guide files*) that add instructions for selected system-level tasks. In addition, you can create guide files that tell your customers how to get specific tasks done in your application.

An improved version of Apple Guide, version 3.0, will ship with Mac OS 8. This version

will be integrated with Mac OS 8 and will include enhancements such as text input fields, custom controls, the ability to save and restore data entered by the user, context-sensitive loading of help topics, an enhanced "Look For" mode that searches both the title and the contents of a guide file, and the ability to print summaries of guide instructions.

With Apple Guide, you can create your own "do it for me" buttons that, when clicked, send out a series of Apple events to perform a task automatically. (You can also have "do it for me" work occur automatically when the user advances to a new guide panel.) Through user studies, Apple has found (big surprise!) that users prefer "do it for me" assistance over sequences of instructional guide panels. Apple encourages you to add buttons to your guide panels that automate tasks—your customers will love you (and your applications) for it.

Selling Your Software Worldwide

No matter where you live, the rest of the world represents a larger potential market than the country (or geographic region) you live in. Under System 7, you most likely developed your application to support one scripting system (usually English). Then when you felt ready to move to other markets, you modified your existing application. You were probably able to keep the same code and (to simplify this explanation a bit) only modify text resources to translate your application from, say, English to Spanish. But you most definitely had to change your code to support complex 1-byte scripts such as Arabic and Hebrew, and you had to make even more changes to support 2-byte scripts such as Japanese and Chinese. This left you with at least three code bases that you then had to maintain in parallel.

Mac OS 8 greatly decreases the amount of work needed to create multiple versions of your software for sale worldwide. It does this through the integration of the System 7 WorldScript technologies into Mac OS 8 and the addition of a new data type called *text objects*.

The Need for Text Objects. Developers currently use the WorldScript technologies, which began with System 7.1, to simplify the creation of software that can handle 2-byte and complex 1-byte scripting systems. Although WorldScript certainly helps, it does not handle all the issues connected with multiple scripting systems. For example, a text string contains the data that represents the intended text in a certain encoding (scripting system), but it does not specify which

encoding is being used. That data has to be maintained outside the text string, a solution that increases the software's complexity and also necessitates changing the source code if you want to support a different encoding.

The Mac OS 8 Toolbox managers make it much easier to implement features that, today, you have to write, debug, and maintain yourself.

Text objects simplify your software by hiding the details of text encoding from your code. They are opaque data structures that contain the intended text, the encoding specification for the text, language and geographic locale information, and any other arbitrary annotation to the text. (For example, the Chinese language does not have an alphabet, so it's impossible to sort a list of Chinese strings alphabetically. If Chinese text objects have a phonetic representation of the text attached to them as an annotation, software can use this data to sort a list of words in a way that's meaningful to a reader of Chinese.)

Text objects are the technology that Apple will use to support Unicode and whatever other text encodings need to be supported in the future. Mac OS 8 supports the Unicode 1.1 standard, and it will support future versions of the Unicode standard. The Text Object Manager provides all the routines you need for manipulating text objects, including routines that convert text objects to and from Unicode, Pascal, and C strings.

Apple recommends that you use text objects for all the text that shows up in your human interface—for example, static text in menus and dialog boxes. Because text objects carry all the information needed for them to be displayed properly, an application that uses

them will display itself correctly on any Mac OS-compatible computer that includes support for multiple languages. For example, a text object-based application localized for Hebrew will display itself correctly on a computer with U.S. English system software if the Hebrew script system is installed on it.

Creating International Software. Text objects also make it easier for you to design and write your software with the world market in mind. Here's how you should do this: Instead of creating a version of your product for one market, then modifying your source code to create successive versions, you use text objects, WorldScript, and the superior text engine of QuickDraw GX to create one "international" version of your code.

You must still localize this international version of your application for each scripting system you want to support—but these Mac OS 8 technologies make the process much simpler. Instead of having to modify your source code (resulting in multiple versions that you must maintain in parallel), all you have to do is modify text object resources to create a new version of your application localized for a different market. This is a task that can be done by localizers (leaving your key programmers free to go on to new projects), and you end up with only one version of source code to maintain—a significant improvement over the same situation under System 7.

Apple plans to release versions of Mac OS 8 for Japan, the United States, and Europe "virtually simultaneously," with localized Mac OS 8 versions for every market Apple supports to be available within 12 weeks of the first versions introduced. This means that Mac OS 8 will be a global phenomenon very quickly, and worldwide customers will have a healthy appetite for software that you have localized to their market.

Even if you only expect to market one version of your software at first, Apple strongly recommends that you create an international version of your software and localize it for your first market. The modest additional effort required to do so will pay off for you when you move into additional markets.

The Mac OS 8 Graphics Architecture

Under Mac OS 8, Apple is offering a graphics architecture that is compatible with the past (through QuickDraw) but offers improved drawing, typography, and printing through the integration of QuickDraw GX, QuickDraw 3D, and WorldScript—all of which are always present

on any computer running Mac OS 8. Apple will maintain QuickDraw so that existing System 7 applications will continue to run—and if QuickDraw fits your needs better, you can continue to use it for your Mac OS 8 applications.

However, QuickDraw GX is the graphics technology that Apple will continue to enhance as the Mac OS evolves, and Apple strongly encourages you to use it. In addition, the fragment-based method of loading software in Mac OS 8, algorithmic improvements, and code sharing between QuickDraw and QuickDraw GX significantly reduce the memory-footprint and performance issues that discouraged some developers from using QuickDraw GX in their System 7 applications.

QuickDraw GX is designed to handle all forms of imaging on Mac OS-compatible computers—not just imaging to the screen, but also to PostScript™ and bitmapped printers and other imaging devices. In fact, if you use QuickDraw GX, your application will be able to access advanced PostScript features—including transfer modes, perspective transforms, and 2-byte font downloading—that would be inaccessible under QuickDraw.

QuickDraw GX is an exciting graphics architecture that offers a world of advantages over QuickDraw in four areas—drawing, typography, printing, and acceleration. QuickDraw GX offers too many features to list here (see the book *QuickDraw GX: Programmer's Overview*, available from Addison-Wesley, for that), but here is a brief summary:

- **Drawing.** QuickDraw is based on a resolution-dependent, state-based architecture—that is, the state of QuickDraw itself influences how and where a graphic element is drawn. To change the size, line thickness, or color of an oval, for example, you must change QuickDraw variables and issue the oval-drawing command again.

On the other hand, QuickDraw GX is resolution-independent and object-oriented: Each QuickDraw GX shape is an object that contains all the information needed (for example, line thickness and color) for it to be drawn correctly. QuickDraw GX shapes include everything from points and lines to polygons, multiple-curve paths, and text with multiple styles. In addition, you can combine shapes into a complex shape that can be manipulated as a single entity. QuickDraw GX defines color in a device-independent way and supports multiple color spaces.

At a higher level, QuickDraw GX provides built-in support for a variety of transformations and geometric operations, including scaling, skewing, translation, and geometric arithmetic (for example, creating a new shape that is the “sum” of two other shapes).

- **Typography.** QuickDraw GX provides a simple mechanism for creating and manipulating “ordinary” text—that is, text with a single font and typestyle. It also includes mechanisms for dealing with more complex styled text, including text that combines writing from different scripting systems.

Since text is stored as a QuickDraw GX shape, it can be manipulated using QuickDraw GX transformations and still remain text. This allows you to do things like rotate text or “flow” it onto a curved path and still be able to select and change it afterwards.

QuickDraw GX gives you extremely sophisticated typographic control over text, making it an excellent “engine” for developing sophisticated publishing solutions. Even if that is not the thrust of your application, however, any application that uses QuickDraw GX text automatically gains the benefits of QuickDraw GX fonts, which can contain built-in “intelligence” about how best to display themselves; this can give your application better-looking text with no additional work on your part.

- **Printing.** The printing architecture of QuickDraw GX offers benefits to both your customers and you. Your customers get a more powerful, easier-to-use human interface for printing (including printing by dragging a document icon on top of a printer icon). They also get portable digital documents (PDDs); any Mac OS 8 or other QuickDraw GX users can open a PDD, regardless of whether they have the application or fonts used to create it. (In the past, people have made limited use of PDDs because most people don't have QuickDraw GX installed on their computers. Because QuickDraw GX is an integral part of Mac OS 8, any two Mac OS 8 users will be able to exchange PDDs.)

QuickDraw GX printing simplifies your work by allowing you to use the same programming model to print, regardless of the type of device being printed to. QuickDraw GX makes it easier for you to add application-specific controls to your application's printing dialog box. In addition, Apple designed the QuickDraw GX printing architecture so that you can change or enhance the printing process through the use of software components called *printing extensions*. Using

printing extensions, you can change the printing process in ways that would, under System 7, require rewriting the printer driver itself. For example, you can create a printing extension that controls a printer's sheet feeder or prints a background image on every page.

- **Acceleration.** The Mac OS 8 graphics architecture includes an abstraction layer that supplies implementation-independent hardware acceleration of graphics. This means that, if your application uses the Mac OS 8 graphics architecture, it will automatically gain the benefit of whatever graphics-acceleration card is plugged into the computer, or it will run correctly (but slower) when no graphics-acceleration card is present.

OpenDoc

Like QuickDraw GX, OpenDoc is another Mac OS architecture that is integrated into Mac OS 8. OpenDoc is an open, cross-platform architecture for component software, one that will make applications more useful by allowing them to incorporate new kinds of content. (For example, an OpenDoc-aware word processor can incorporate any other OpenDoc part without having knowledge of that part's structure.) In addition, OpenDoc provides users with a more consistent user experience and a shorter learning curve. Users can add, say, graphs to all their OpenDoc-aware documents through the same graphing part—they need learn only one graphing package, and it presents the same familiar interface wherever they use it.

- **Cyberdog.** In addition to being a technology that is available today, Apple has made Cyberdog the net-browsing solution for Mac OS 8. Cyberdog includes a set of OpenDoc “Internet building-block” parts and an architecture that integrates Internet connectivity into the Mac OS. (For an early look at Cyberdog, see “Putting OpenDoc to Work in 1996” in the February 1996 issue of *Apple Directions* or at <http://dev.info.apple.com/appledirections/feb96/opendoc.html> on the Web.)

Apple encourages you to use Cyberdog, build your own Cyberdog parts, and build new parts that replace and improve on the Cyberdog parts that Apple has supplied. Some companies are already doing so. Spyglass is producing a Cyberdog-aware Web browser. At the Apple Worldwide Developers Conference in mid-May, Netscape officially endorsed OpenDoc, saying that its first project will be to work on a replacement for the Cyberdog browser part.

For more on OpenDoc and Cyberdog, see their Web pages (<http://www.opendoc.apple.com/> and <http://cyberdog.apple.com/>).

"Kickstarting" OpenDoc in 1996.

Apple will be doing a lot this year to promote OpenDoc (including an official introduction of OpenDoc to the public at this August's Macworld Boston) and to make developing for OpenDoc more attractive to you.

To make OpenDoc-based computers immediately useful and to give you a foundation of basic parts on which to build, Apple will introduce a set of "Kickstart" parts in the fourth quarter of this year. This set will include

- QuickTime, QuickTime VR, and QuickDraw 3D viewer parts
- a sound playback part
- a part for displaying 2D images (PICT, GIF, JPEG, and TIFF formats)
- a text container part (a word-processing part that can have other parts contained in it)
- a draw (object-oriented drawing) container part
- a HyperButton part that, when clicked, can play a sound or launch an application, AppleScript script, or file
- a text note part, which lets users annotate and place bookmarks in their documents

All these parts will be scriptable and will include their own Apple Guide support.

Apple will make these parts readily available (eventually bundling them with every Mac OS-compatible computer that includes OpenDoc), and you will be able to ship these parts with your own software.

Using OpenDoc to Support New Technologies. Apple has made the statement that it will use OpenDoc to integrate new technologies into the Mac OS platform. In addition to doing so with its own technologies (the QuickTime, QuickTime VR, and QuickDraw 3D Kickstart parts), Apple is also working on parts that will play Java applets and Netscape Navigator™ plug-ins. Not only does this give you access to these technologies, it also allows you to use them (through Cyberdog) in a new context—your own documents.

Third-Party Support. At the Apple Worldwide Developers Conference, several companies demonstrated OpenDoc container applications and parts. Claris demonstrated a version of ClarisWorks that operated as a container application (with a year-end delivery date), and WorldSoft said its WorldWrite word processor would also be a container application by year's end. Adrenaline Software

showed two dynamite OpenDoc parts, a spreadsheet and a charting editor, and Digital Harbor demonstrated a word-processing part called *WAV*, which showed some intriguing "hooks" that will allow other part vendors to add value.

Becoming OpenDoc-Aware. Apple encourages you to make your software "OpenDoc-aware." This means creating a container application (an application in which users can embed parts) or a container part (a part that can contain other parts). Once you have created OpenDoc-aware software, note the many ways that Cyberdog and the Kickstart parts can help you out:

- You can create OpenDoc parts, knowing that your customers will have useful parts to add to yours and useful containers in which all parts can be placed. (Note also that any part can be the "root" of an OpenDoc document, so you don't necessarily need a separate container part for your part to be useful. If your part is itself a container part, it can be the focal point of truly interesting OpenDoc documents.)
- Your OpenDoc-based development is simplified by the fact that you don't have to implement all the data formats mentioned earlier—the Kickstart parts automatically do that for you.
- You can add Internet access to your OpenDoc-aware solution simply by adding Cyberdog parts to it.

OpenDoc and Mac OS 8. The Mac OS 8 implementation of OpenDoc runs faster and takes up fewer system resources than its System 7 equivalent. Parts written for System 7 will also work under Mac OS 8, but you can also write Mac OS 8—savvy parts that take advantage of the Mac OS 8 architecture—including such features as multiple preemptively scheduled tasks, text objects, and HI objects. By hooking into the Mac OS 8 architecture, you will be able to create more powerful software with less effort.

Open Transport

Mac OS 8 uses Open Transport as its architecture for networking and communication. Under Mac OS 8, Open Transport runs in protected memory and is preemptively scheduled and multithreaded, thus improving its performance and stability when compared to its System 7 implementation.

One of the most important features of Open Transport is its transport independence, which refers to the fact that the Open

Transport API is independent of whatever networking protocol is being used. This means that you need only write your application once, and you can sell it to a wider range of customers, regardless of which networking protocol they are using.

In addition to supplying a single API that allows you to write your program once and still support multiple networking protocols, the Mac OS 8 implementation of Open Transport also offers a benefit called *multihoming*—a feature that will first be released in a future System 7 version of Open Transport. Currently, AppleTalk can service only one kind of connection at a time. For example, a user today would have to go to the Chooser to switch between a connection to a personal printer and a connection to an Ethernet-based e-mail system. With Open Transport multihoming, a Mac OS-compatible computer will be able to maintain both connections simultaneously.

Another advantage to using Open Transport—both today and under Mac OS 8—is that it gives your customers a more flexible human interface that's easier for them to use and configure; you'll benefit by spending less time implementing the human interface portion of your networking application. Open Transport provides a consistent human interface for configuring AppleTalk and TCP/IP networks and lets users interact with this interface at three levels of expertise: novice, expert, and administrator. In addition, Open Transport stores the configuration information that implements a given network connection in the form of a preference file. This makes it easier for users to store and invoke different network connections and for administrators to distribute predefined network connections to users.

For more information on Open Transport, see "Unified Networking for Mac OS Computers" on page 18 of the September 1995 issue of *Apple Directions* (<http://dev.info.apple.com/appliedirections/sep95/unifiednetworking.html>).

Opportunities for Hardware Vendors

The modularity and extensibility of Mac OS 8 create business opportunities for various hardware vendors. Peripheral device vendors, for example, can create plug-ins that allow Mac OS 8 computers to connect to a variety of peripherals, including video-capture devices, scanners, graphics tablets, and laboratory equipment.

In addition, hardware vendors who wish to sell Mac OS-compatible computers can take advantage of the modularity of the Mac OS 8

I/O architecture to offer computers that have unique features. By writing an I/O family for a different bus architecture—FireWire, for example—they can then add peripherals that meet the needs of specific markets.

Conclusions

Mac OS 8 marks a major advance in the evolution of the Mac OS. It is a significant streamlining and enhancement of the operating system, but it is also a transitional operating system

that seeks to maximize backward compatibility for hardware and software.

Because of its transitional nature, Mac OS 8 gives you an extended “window” to make the move from System 7 to Mac OS 8. However, you should be planning your transition to Mac OS 8 now. Apple and its licensees will eventually ship Mac OS 8 on their new computers, and applications that don't make aggressive use of Mac OS 8 technologies will be at a disadvantage in a highly competitive marketplace.

But the best reason for adopting Mac OS 8 is because it allows you to create more powerful, stable, and easy-to-use products. Mac OS 8 represents a new beginning for the Mac OS, and if you're one of its early adopters, you have the chance to profit from its increasingly widespread adoption. ♣

APPLE NEWS

Amelio Announces Strategy

continued from page 1

Copland, will be released in an update to System 7 later this year, while the complete Mac OS 8 is expected to ship by mid-1997.

- Much of Apple's new-product direction will focus around the Internet, since Newton and Pippin will in the future enable Internet access as will a forthcoming lower-cost online information access device.

During his talk, Dr. Amelio emphasized how important it will be for Apple to step up efforts to support developers. Calling for all of Apple to work honestly and cooperatively with the development community, and intending to avoid duplicating developers' efforts, he also announced the following:

- Apple will spend some \$18 million on tools development in the next year.
- \$20 million will be made available this year to assist developers' marketing efforts, helping you get products to market, increasing shelf space for your products, and developing new comarketing programs. (See the news story “Apple Commits \$20 Million to Assist Developers' Marketing Efforts” on page 12 for more details.)
- Apple will reduce hardware costs for developers.
- A new program will be created to reward Apple-only dealers.

To hear the entire address, you can find a RealAudio version of it at the WWDC WebCast site (<http://wwdc.apple.com/webcast/archives/raarch.html>).

You can also find more about Apple's strategy and plans for the future at the following two Apple Web sites:

- <http://ThePlan.apple.com>
- <http://www.apple.com/leadership/>



Apple's New Organization

In late May, Apple Computer announced that it will re-form its corporate structure to carry out the company's new strategic directions, which were outlined by Apple CEO and Chairman Dr. Gil Amelio at the Worldwide Developers Conference. (See “Amelio Announces Apple Strategy at WWDC” on page 1.)

Under the new organization, six senior executives will report to Dr. Amelio. They are

the chief operating officer, chief administrative officer, chief technical officer, chief financial officer, senior vice president of corporate marketing, and vice president of strategic planning. Here are some of the details.

Executive Vice President and Chief Operating Officer Marco Landi, formerly president of Apple Europe, Middle East, and Africa, will be accountable for the overall implementation of Apple's business plans. He will oversee six distinct profit and loss centers, including four newly announced product divisions, as well as Claris Corporation and AppleAssist, another new organization. In addition, he will be responsible for segment marketing and worldwide sales. His organization is as follows:

- The Power Macintosh Products group, led by Senior Vice President Fred Forsyth, will be responsible for the family of Power Macintosh desktop products and PowerBook

“Just the Facts, M'am”

Facts. No fluff. That's what Apple Directions Express gives you.

Apple Directions Express is the online business bulletin sent directly to your e-mail in-box from Apple Computer, Inc. It's a biweekly summary of up-to-the-moment news from Apple for busy computer industry professionals, and includes pointers to other locations for details.

Subscribe by sending e-mail to adirections@thing1.info.apple.com. In the Subject field, type the string “subscribe <your real name>”.

products. Forsyth will also be responsible for worldwide operations.

- The Imaging division, directed by Senior Vice President Terry Bailey, will be responsible for products including printers, scanners, digital cameras, and other peripherals.

- Claris Corporation, headed by Claris President and Apple Vice President Guerrino De Luca, will continue to function as a subsidiary of Apple, maintaining its own sales, marketing, and operations staff.

- The Information Appliance Products division, temporarily headed by Marco Landi, will develop low-cost, special-purpose products—such as emerging network-access systems—that fall outside the traditional personal computer category. These products will be based on Newton and Pippin technologies.

- The Servers and Alternate Platform Products division, managed by Senior Vice President Howard Lee, will oversee the development and marketing of products based on the PowerPC platform (or Common Hardware Reference Platform, also known as CHRP) and other hybrid personal computer products. In addition, the division will be responsible for the entire line of both Mac OS–based and non–Mac OS–based server products and interactive media products.

- The AppleAssist division, temporarily managed by Marco Landi, is Apple's newly formed support organization dedicated to providing life long assistance to owners of Apple products.

- The Segment Marketing group, headed by Senior Vice President Dr. Terry Crane, will be responsible for assessing the product and solutions needs of customers throughout the world, and working with Apple's product businesses to bring those products and solutions to market. The group's charter includes market strategy formulation, market analysis and forecasting, customer research, competitive analysis, and solutions development. The focus will be on selected areas within enterprise/scientific and technical, publishing and authoring, education, and home and small office/home office (SOHO) market segments.

- Worldwide Sales, headed by Senior Vice President John Floisand, will be responsible for maintaining customer relationships around the world and implementing an optimized channel strategy. The organization's roles will include channel relationships, customer requirements feedback, fulfillment strategy development, solutions implementation, and

sales for all Apple products, including products based on Newton technology.

Also reporting to Dr. Amelio is Executive Vice President and Chief Administrative Officer George Scalise, who will oversee developer relations, licensing, human resources, external relations, facilities, communications, and Apple's legal department.

Dr. Amelio will carry out the functions of chief technical officer until a permanent leader in this area is named later. The following functional areas will report to the chief technical officer:

- AppleSoft, headed by Senior Vice President Ike Nassi, will be responsible for operating systems, application programming interfaces, tools, and platform marketing.

- AppleNet, led by Vice President and Chief Scientist Larry Tesler, will be responsible for Apple technologies for the Internet and intranets, ensuring that Internet capabilities are integral to all of the company's products.

- Apple Advanced Technology Laboratories, managed by Apple Fellow and Vice President Dr. Donald Norman, will oversee the development of concept products and core technologies to be incorporated into future Apple products.

- The Apple Reliability and Quality Assurance group, headed by Vice President Mike Connor, will oversee Apple's reliability and quality programs on a company-wide basis.

Executive Vice President and Chief Financial Officer Fred Anderson, also reporting to Dr. Amelio, will provide strategic, structural, and procedural leadership in financial management; shareholder and investor relations; tax, customs, and records management; mergers and acquisitions; and information systems.

Senior Vice President of Worldwide Corporate Marketing Satjiv Chahil, Apple's top marketing executive, will lead a strengthened organization responsible for direction setting in overall marketing and advertising campaigns. His organization will also ensure alignment in the execution of all marketing programs.

Dr. Amelio's sixth direct report will be Vice President of Worldwide Strategic Planning Douglas Solomon. His team will be responsible for development and implementation of the strategic planning process for the company. Dr. Solomon will also manage the corporate development function.



Apple Commits \$20 Million to Assist Developers' Marketing Efforts

As we reported in our page 1 news story, "Amelio Announces Apple Strategy at WWDC," Apple is setting aside \$20 million over the next 12 months to help you reach customers by increasing the visibility and availability of third-party solutions through comarketing and channel marketing programs. Currently, Apple Developer Relations Vice President Heidi Roizen is working with her organization to develop and implement a plan for accomplishing this goal.

Here's what Heidi recently said about how she and her team are developing plans to spend that \$20 million:

"The goal is to allocate \$20 million over the next 12 months in the most efficient, effective way possible to help you get your products in front of customers. This money will be spent roughly half in the United States and Canada and half in Apple's markets outside the United States. We're working on programs now that will best leverage this money in both the traditional channel as well as in new and emerging channels. The funding is likely to be split among channel initiatives, 'virtual'/Internet initiatives, comarketing/collateral materials, and some pull advertising. We intend to use these funds to help demonstrate the breadth and diversity of a variety of existing and new solutions for the Mac OS rather than programs that focus on one or two products. In other words, we won't be funding specific proposals to support marketing of individual products.

"In order to maximize the leverage, we want your input so we can create standard programs that can either be shared by all and/or be targeted to particular market and/or technology segments. Therefore, I encourage you to give ideas to Jonathan Fader and his team in Developer Marketing, or to International Developer Relations Manager David Krathwohl if you work outside the United States. We'll keep you in the loop as the programs become reality.

You can send your input by e-mail to Jonathan Fader at fader@applelink.apple.com, or to David Krathwohl at krath@apple.com. You can also send your ideas to us at a.directions@applelink.apple.com, and

we'll be sure to send them to the right people. As soon as details about the new marketing programs are available, we'll report them in *Apple Directions*.

In the interim, one of the best ways to help us get the word out about your Mac OS products is to be sure they're entered in the It Shipped! database, which we broadcast as widely as possible. To enter your product, visit the It Shipped Web page (<http://dev.info.apple.com/itshipped.html>).



Study Confirms Macintosh Superiority in Ease of Use and Customer Satisfaction

People who use both Macintosh computers and Windows 95-based computers say that the Macintosh makes them more creative, productive, and satisfied, according to a recent study by Evans Research Associates, an independent market research company. Study participants also rated the Macintosh as being easier to use, more enjoyable, and better for graphics, multimedia, and running educational software than Windows 95-based systems.

The study was conducted by Evans Research in March and April 1996 and included 150 people who regularly use Macintosh computers running System 7.5 and Windows 95-based personal computers in business, home, or education. The study found that these "dual users" rated the Macintosh computer higher in 14 different categories, ranging from ease of use to overall satisfaction.

In the category of overall satisfaction, 52 percent of the dual users rated the Macintosh computer higher while 39 percent preferred Windows 95-based machines. For making the user creative, the Macintosh was preferred by 53 percent versus 27 percent for PCs with Windows 95.

Macintosh computers running Macintosh System 7.5 won over Windows 95-based PCs in the following categories: satisfaction, productivity, enjoyability, stability, ability to make the user creative, graphics and publishing features, multimedia features, and ability to run educational software. Macintosh computers

also came out on top in six ease-of-use categories, including overall ease of use, ease of learning, ease of connecting peripherals, ease of setup and installation, ease of connecting to a LAN, and ease of troubleshooting.

"The study proves what Apple has said all along: Despite all the hype, the Macintosh computer is still the easiest computer for creating, learning, and getting work done," said Michael Mace, Apple's director of Macintosh platform marketing. "It's especially pleasing that this endorsement came straight from those who can judge best—real people who use both Macintosh computers and Windows-based PCs."

More details from the study can be found on the Web (<http://www.apple.com/whymac>); you can also request more details by calling 800-825-2145.



WWDC Wrap-Up

Apple Computer, Inc., made an unprecedented number of announcements at this year's Worldwide Developers Conference (WWDC), sending out more than 20 press releases from the event, which wrapped up May 17 at the San Jose Convention Center. We've already told you about Dr. Gil Amelio's opening address, in which he announced his strategy

for the future success of Apple. (See page 1.) The following is a summary of some of the other important items; for our complete WWDC coverage, go to <http://dev.info.apple.com/appledirections/adtoc.html> and look at "WWDC Special #1", "WWDC Special #2," and "WWDC Special #3" under Apple Directions Express.

[Editor's Note: If you subscribe to Apple Directions Express, our e-mail digest of business news from Apple, you've already seen the following items; we sent them to the list server the week of the conference. If you don't already subscribe, you should consider doing so; reading Apple Directions Express assures that you'll see all the developer-related news from Apple, sent to you in as timely a fashion as possible. Subscribe by sending e-mail to adirections@thing1.info.apple.com. In the subject field, type the string "subscribe <your real name>".]

Apple Internet Solutions Strategy Announced

WWDC attendees witnessed the unveiling of Apple's Internet strategy, which was described by Dr. Amelio and Larry Tesler, vice president of Internet platforms. Apple's intention is to do for the Internet what it did for personal computing: to make the most advanced capabilities accessible to everyone who can benefit from them, and to define for the Internet the same media-rich user experience that has

"And that's all we have to say about that"

Apple Directions Express is concise and to the point, saving you the time it would take to pore over a variety of sources to stay up-to-date with Apple. Pointers to Internet locations make it easy to get more detail.

Apple Directions Express is the online business bulletin sent directly to your e-mail in-box from Apple Computer, Inc. It's a biweekly summary of up-to-the-moment news from Apple for busy computer industry professionals, and includes pointers to other locations for details.

Subscribe by sending e-mail to adirections@thing1.info.apple.com. In the Subject field, type the string "subscribe <your real name>".

made Macintosh the platform of choice for creating, learning, and communicating.

The plan includes several major initiatives:

- To make Apple's own platforms (Macintosh, Newton, Pippin) best-of-class for Internet access, content creation and delivery and to move all those platforms from "Internet-optional" to "Internet-standard"; by the end of 1996, all Macintosh models are planned to ship "Internet-ready"

- To enrich Internet standards by cross-licensing technologies with other Internet leaders to create an open standard for Internet media, which will bring to customers broader choices, easier access and richer content; wherever appropriate, Apple will contribute its leading-edge technologies to this effort

- To use the Internet to run Apple's own business better, enhancing communication with its customers and developers, expanding efforts to create innovative Web sites, increasing "Webcast" programming, and integrating more community-building activities into much of Apple's Internet presence

Apple Internet solutions will be based on the following key platform components of networking infrastructure: QuickTime, OpenDoc, and Java. Regarding networking infrastructure, Apple plans to support industry standard TCP/IP protocols as a core component of the Mac OS. The first step of this move has been the introduction of Open Transport, which makes TCP/IP an equal peer of AppleTalk and is now available as part of Apple's latest Mac OS release, System 7.5.3. Apple plans to apply its expertise in plug-and-play networking to advance its TCP/IP implementation.

Apple also intends to drive adoption of QuickTime as the industry-standard multimedia format for the Internet. Apple recently announced the creation of QuickTime "fast start" movie playback, which allows users to begin to view a movie while it's still downloading. Netscape Communications has also recently announced that it will be bundling QuickTime with its upcoming Navigator™ 3.0 release. More than 20,000 Web pages today already use Apple QuickTime content, and more than 5,000 use QuickTime VR.

OpenDoc allows Apple to merge otherwise disparate software such as Cyberdog, Java applets, and Netscape Navigator plug-ins. Apple and developers showed a number of OpenDoc parts and containers at the conference, with an emphasis on parts providing Internet access. At the conference, Apple also

provided more details on its plans to leverage Java, saying it plans to embed Java across a range of Apple products and technologies, including the Mac OS, OpenDoc, Cyberdog, HyperCard, Newton, Pippin, and Apple Web servers. Additionally, Apple and third parties showed Java applets using OpenDoc running as part of various Macintosh applications.

Copland Now Mac OS 8: New Web Site Goes Live

At the WWDC, Apple announced that Copland will be known as *Mac OS 8*, revealed the Mac OS 8 logo, and introduced the Mac OS 8 Web site. The first developer release of Mac OS 8 will be made available by the middle of this year; until then, the Web site (<http://www.macos.apple.com/macos8/>) will provide a variety of content to help you understand Mac OS 8 technology and get ready to work with it.

The site features Developer World, a page with complete information from Apple about developing with and for Mac OS 8. It already includes a great deal of technical and marketing information, as well as news, that you can use to start gearing up for next major release of the Mac OS.

Cyberdog 1.0 Ships; Developer Providing "Dog-Savvy" Apps

In the wake of Cyberdog demos stealing the show at last year's WWDC, Apple distributed the first customer version of its OpenDoc-based, integrated Internet-access software suite to this year's conference attendees. A consumer-oriented version of the product, including broad distribution, is planned for later this year.

Cyberdog's integration with the Mac OS and Macintosh applications embeds Internet resources throughout the user's work, making Internet connectivity an extension of the desktop. For example, a user can drag and drop the address of a live Web site to the desktop, driving Internet links from the operating system rather than a browser. Cyberdog also provides DocBuilder, an OpenDoc application that supports inclusion of Cyberdog components along with text and graphics. With Cyberdog's DocBuilder, and the availability of future OpenDoc-enabled word processing packages, users can create intelligent documents that combine text, graphics, data, and information from the Internet.

In addition to browsers for the Web, Gopher, and FTP, access to telnet, notebooks for storing Internet addresses, and a log for

tracking sites visited, Cyberdog offers advanced mail and news reading capabilities. For example, Cyberdog can manage multiple e-mail addresses, allowing users to manage incoming mail from various mailboxes. Mail handlers can identify prioritized and unwanted mail and categorize it for the user. A search mechanism helps find e-mail messages by content, and can also archive Internet newsgroup information. In addition, the message editor for mail and newsgroups supports full-styled text as well as graphics.

Numerous developers are already actively developing or enhancing applications to work with Cyberdog. For example, Macintosh word-processing software vendors Claris, WorldSoft, and Digital Harbor are supporting Cyberdog to enable their applications to develop "live" documents that link to Internet resources. In addition, Addison-Wesley is developing interactive CD-ROM titles that use Cyberdog to retrieve content from the Internet; Corda Technologies offers a graphing component that uses Cyberdog to graph information directly to Web pages; and OnBase Technology is offering users an alternative to the standard Cyberdog notebook.

Cyberdog 1.0 is available free of charge from the Web (<http://cyberdog.apple.com/>).

Cocoa Turns Kids Into Internet Apple Developers

Apple previewed its new interactive media authoring tool for children (ages ten and above) and nontechnical adults at the WWDC. Called *Cocoa*, the tool, along with its accompanying Netscape Navigator plug-in, makes it literally "childs play" to build simulations, games, and interactive worlds into Web pages.

"Just as Sun Microsystems' Java and Macromedia's Shockwave have provided custom, media-rich Web content for technically savvy developers, Cocoa brings this technology into the reach of kids and nontechnical adults," said Larry Tesler, vice president of Internet platforms.

Cocoa uses Macintosh Drag and Drop and a built-in paint tool to provide an intuitive interface. Users create objects, each of which can have independent behaviors assigned to them. Users accomplish the programming by simply demonstrating an object's desired reactions. The system then writes all the code automatically, transparently to the user. An element of randomness can be added to the objects, making them run differently every time.

When randomized objects are embedded in Web pages, a completely novel experience

can be generated every time the Web page is visited. Using this technology, children could use the World Wide Web to build a huge, ever-evolving adventure game, completely designed and implemented by them, playing differently every time through.

Cocoa for Mac OS and Windows 95 will be made available free of charge later this year.

OpenDoc DR 5 Released; Developer Momentum Builds

One of the most sought-after items at the WWDC—other than the Cyberdog “The dog ate my Web page” T-shirt—was the OpenDoc Developer Release 5 (DR 5) CD, which was released to attendees. In addition, a variety of developers displayed their OpenDoc software at the event, evidence that customers can soon expect an abundance of new commercial products using OpenDoc.

OpenDoc DR 5 is an updated collection of tools, components, documentation, and source code. It features a first look at Kickstart viewer components—several multimedia technologies from Apple packaged together as easy-to-access OpenDoc components—and the release of the OpenDoc Developer Framework (ODF), the object-oriented framework for creating OpenDoc and Internet/Cyberdog-compliant software. The first release of ODF guarantees release-to-release binary compatibility and includes sources and binaries for the framework and shared library as well as tools, utilities, and sample parts. DR 5 makes several improvements to the original OpenDoc release, including modifications to the installer script and the editor setup control panel.

To date, more than 350 developers have begun work on OpenDoc-based products for the Mac OS, Windows, OS/2, and UNIX platforms as well as for the Java environment and the Internet. Among the OpenDoc applications on display at the WWDC were

- ClarisWorks, Claris’s office suite designed with OpenDoc components
- Roaster, Natural Intelligence’s tools for developing in the Java environment on the Macintosh computer
- Opal, Digital Harbor’s word processor
- WorldWrite, WorldSoft’s multilingual word processor
- dtf Database ToolKit, Theta Group’s OpenDoc front-end suite for relational databases

You can download OpenDoc DR#5 from Apple’s OpenDoc Web site (<http://www.opendoc.apple.com/>). It also features the

latest list of OpenDoc developers (<http://www.opendoc.apple.com/press/committed.html>).

Aggressive Plans to Incorporate Firewire

You’ve been hearing about the Firewire standard invented by Apple for the last couple of years. Now, Apple plans to aggressively incorporate the new bus technology across its product line, according to its WWDC announcement. “We intend to build Firewire into every Macintosh product we sell. We expect to make the transition to Firewire over the next year by fully supporting third-party adapter cards in our operating system software and by bundling cards in configurations of our desktop machines,” said Howard Lee, senior vice president of Apple’s Macintosh systems division. “We plan to continue this transition by taking Firewire straight to the motherboard.”

Firewire is the cross-platform implementation of the IEEE 1394 high-speed serial bus standard already licensed by Texas Instruments, Fuji Film, Sony, and others. IEEE 1394/Firewire offers a fast real-time digital connection for many kinds of consumer electronic and personal computer applications. The Firewire standard is the first to permit attachment of new digital consumer products—including digital video disks, digital videotapes, advanced televisions, set-top boxes, music systems, personal computers, and personal computer peripherals—directly to a personal computer.

Apple said it expects support for IEEE 1394/Firewire to be available in the Mac OS in the first half of calendar year 1997, and that its Pippin 1997 reference platform will also incorporate the standard. Select configurations of Apple desktop systems are expected to ship in 1997 with preinstalled IEEE 1394/Firewire PCI host adapter cards and software. Standard configurations of all 1998 Apple notebook and desktop products are expected to ship with IEEE 1394/Firewire built into the main logic board as a standard feature.

QuickTime VR 2.0 Features Announced

As expected, Apple revealed the feature set from its forthcoming QuickTime VR 2.0, the next release of the virtual reality programming and playback environment for the Mac OS and Windows. In late May, Apple also shipped the first developer release of the API, and it hosted developers at a “kitchen” for use of and feed-

back on the new API.

Some of the technical details of QuickTime VR 2.0 follow.

New API. The new API lets multimedia title developers incorporate QuickTime VR functionality into their applications. Using the API, you’ll be able to

- display and manipulate panoramas and objects
- combine still images into panoramas
- turn QuickTime movies into panoramas
- combine QuickDraw 3D objects into panoramas
- incorporate directional sound and speech input
- incorporate specialized transition behavior between panoramas
- override QuickTime VR’s default behavior for event handling, cursors, and “hot spots”
- incorporate specialized hot-spot information unique to the application
- manipulate QuickTime VR memory usage

Internet Location Hot Spot Support.

QuickTime VR 2.0 hot spots will be able to include Internet address information. Users who click a hot spot in a panorama or on an object can be taken to Web sites and other Internet addresses. For example, a user will be able to click on a painting inside a panorama of a museum and go to an Internet site that describes the painting’s history.

Object Support. With QuickTime VR 2.0, QuickTime VR objects will have zoom and translation capabilities. Users will be able to zoom in and out and shift the image up, down, left, and right while zooming. This lets the user take a closer look at a particular area of a QuickTime VR object. Objects will also have hot spots, including Internet addresses and any other kind of hot-spot information. By clicking a hot spot, a user can transition to a panorama, to another object, to another part of the same object, or to any application-specific information.

Tool improvements. QuickTime VR tools will support the new features listed above. For instance, the QuickTime VR scene editor will allow hot spots to be added to objects and panoramas, and will allow Internet addresses to be used as hot-spot destinations, so panoramas and objects can be combined with other information into an online virtual reality world. The stitcher will use considerably less memory for non interactive stitching of images into a panorama. Reliability has also been improved. ♣

Technology

CD Highlights: System Software/SDK Edition

Feature: Inside Apple Game Sprockets

Human Interface: Now It Can Be Told

OpenDoc Human Interface FAQs: Menu-Item Issues, Open Selection, Controls, and More

CD HIGHLIGHTS

System Software/SDK Edition, July 1996

This month I'd like to introduce a new utility, the DevCD SSW Catalog (located in the System Software folder). This catalog contains information about system software on Developer CDs back to Volume 1. Now, if you think I've forgotten to ship Elbonian System 7.5.4, instead of plugging your entire collection of Developer CDs into your drive looking for it, a simple search can determine almost instantly which CDs the software is (or is not) on. Please try it out, and send any comments and feedback to me at alyx@apple.com.

Now, a brief word about system software version numbers. Some of you may occasionally wonder what a folder name like "Norwegian 2-1.0.1-7.5.3" actually means. The last number (7.5.3) is the name of the software; for example, System 7.5.3. The middle number (if present) is the software version; in this case, the software has been revised once since the original release. Finally, the first number is the localization version; this refers to changes in the localizable strings in the software, but not the code.

So, in addition to updates to ColorSync 2.0.1, Interfaces and Libraries, Macintosh CD-ROM Setup, QuickDraw GX, ScriptX, and the Sound Manager, here are this month's new and revised packages.

Apple Game Sprockets DR5

This folder contains version DR5 of Apple Game Sprockets, a bunch of technologies designed so that game developers can spend more time making games cool than making them work. This package also includes the QuickDraw 3D RAVE SDK.

JPEG Sample

This folder contains a demo/sample application for QuickTime JPEG compression. Among other things, it can open a JPEG file, allow the user to view the image, and save the image back to disk. The code is not fully

polished or documented, so be prepared to find some bugs. If you need to read, write, and view JPEG images, then this source codeshould be useful.

Live Scroll 1.0

Live Scroll is a bare-bones application demonstrating one approach for implementing live scrolling (also known as active or dynamic scrolling) while scroll bar thumbs are tracked.

Live scrolling is useful because it offers a better human interface for finding a specific place in a long document. Scroll arrows allow the user to position a document within its window, but they are often too slow when the user needs to scroll a long way through the document. While the scroll thumb can be used in this situation, it is less than ideal because the user can't see the result of the scroll until the thumb is released. As a result, the user needs to move the thumb two or more times before finding the desired position.

An application can overcome these problems by implementing live scrolling, which gives a user direct control over the scrolling speed and provides complete visual feedback as scrolling occurs. The Live Scroll example also demonstrates the two different types of action procedures (for 680x0 and PowerPC processor-based architectures) and the use of the new accessor routines, which are provided as the first step to being Mac OS 8-savvy.

Mac OS 8 Overview

An overview of the technologies, features, and benefits of Mac OS 8. In addition to describing all key features of Mac OS 8, this Adobe™ Acrobat file includes color screen shots and diagrams of the new operating system.

Mac OS 8 Tour

The Mac OS 8 Tour is an interactive demonstration designed to introduce the viewer to

Mac OS 8 features and benefits. It includes demos of capabilities such as multitasking, Find by Content, online experts, and new Finder features.

Network Watch (DMZ) v1.5

The DMZ program demonstrates NBP (Name Binding Protocol) lookup calls under both Open Transport and Classic AppleTalk to obtain a list of all NBP entities by zone. Users can specify the lookup for specific objects and types. This program also demonstrates how to send AppleTalk Echo protocol packets to a specified system and how to receive the Echo protocol response.

DMZ checks for the presence of Open Transport and uses it to perform network lookups; otherwise, it uses the Preferred AppleTalk calls. Under DMZ, the program demonstrates the use of placing the DDP node, used to send the Echo protocol packet, into raw mode, so that the return packet contains the long DDP header information.

Open Transport 1.1GM

Apple Open Transport is the modern networking and communications subsystem for the Mac OS. It's based on industry standards and brings a new level of networking connectivity, control, and compatibility to Mac OS computers, while preserving and enhancing the hallmark of the Macintosh and Mac OS—built-in support for easy-to-use networking.

This release features the Open Transport 1.1GM and 1.1b16 Installer and Debug Installer, plus the Software Development Kit. See the About This Folder document for details.

OpenDoc

OpenDoc is an open industry-standard architecture for building reusable, distributed,

please turn to page 26

Inside Apple Game Sprockets

By Robert Megantz, TacTec

New Technologies Ease Development of Better Games

Apple Computer, Inc., has finally gotten serious about taking advantage of the Macintosh computer's strengths as a game platform. First Apple came up with a hardware architecture—the Power Macintosh platform—that works great with games, and a bunch of new technologies that can be used to create “killer” applications. But that's not all: Apple has just released version 1.0 of the Apple Game Sprockets Software Development Kit (SDK), a set of libraries that makes writing games a lot easier, and the company is strongly supporting game developers.

There are several terrific things about Game Sprockets: First, with minimal programming work, you can incorporate the sprockets into your games and give your customers access to sophisticated features that previously required a lot of work from you. Second, the sprockets provide Apple-endorsed, standard ways of increasing performance and enhancing your game's user experience. In other words, you don't have to write directly to the hardware or in other ways “break the rules” of Mac OS programming as spelled out in *Inside Macintosh* (and thereby risk having your game not work with future Apple hardware and system software) to have your games get the performance they need to stand out in the market. Finally, Apple is giving Game Sprockets away to help you increase the number and quality of game titles written for the Mac OS platform. So, why the big change? It's an interesting story.

Mac OS Games: From Underground to Mainstream

The Macintosh computer has traditionally been sold by Apple as a productivity and educational tool. In fact, to bolster the platform's image as a “serious” system at a time when some analysts tried to write it off as a toy, games were given short shrift. However, over the last few years Apple engineers have worked together with game developers on some underground projects, which have resulted in a variety of seriously great Macintosh games.

Since the release of the first Power Macintosh system in March 1994, many have come to recognize that the computer provides a superb platform for games. Its RISC processor is faster and more powerful than the CISC processors used in other personal computers. For example, the PowerPC processor's floating-point unit (FPU) can efficiently perform the complex mathematical calculations, such as multiply-and-accumulate operations, that are needed to implement 3D graphics and other key multimedia technologies.

The Power Macintosh platform also tightly integrates hardware and software. One of the advantages of such integration is that it can give multiplatform games, such as *Doom 2*, *Dark Forces*, and *X-Wing*, higher resolution and faster frame rates on the Power Macintosh computer than on other platforms. As Ben Calica, Apple's senior product manager of Game Technologies, says, “The Mac is not only a toy—it's a really great toy.”

At the same time, Apple management began to realize that games are big business, and that the market for Power Macintosh games is expanding. According to International Data Corporation, game and entertainment software accounted for 37 percent of U.S. consumer software market revenue in 1995. That's nearly \$1 billion, almost 34 percent more than game revenues in 1994. Of Macintosh computer owners in all markets, 61 percent say they play games. Around half of the purchasers of Power Macintosh 7500 and Power Macintosh 8500 systems bought their computers for home use, and 35 percent of those home users spend at least five hours per week playing games.

To give an example of how good a business game development can be, Domark Software, now a division of Eidos Interactive, started writing games for the Macintosh platform in 1994 and has made a genuine success out of it, as you've read in *Apple Directions* before. Domark sold 50,000 copies of its *Flying Nightmare* flight simulator—its first Macintosh game—and the Macintosh version of *Absolute Zero*, Domark's third game for Mac OS systems, racked up sales of 25,000 units in the first four months of its release. Paul Baldwin, vice president of Marketing at Domark, says, “In the last two and a half years, the Mac market has been very viable for us. First and foremost, the Mac market is starved for games.

As a result, developers who release titles can look forward to a very eager and loyal customer base who will more than likely buy your next game as well. Second, the Mac market is cost-effective. Unlike the saturated PC games market, a developer does not have to spend large amounts of money to make some noise in the Mac market. Savvy developers can make a splash via cost-effective media like the Mac catalogs, AOL, and the Mac user groups.” (For more of Paul's insights on developing and marketing games, check out “Confessions of a New Macintosh Developer” in the November 1995 issue of *Apple Directions*.)

Most 680x0 games written for the Mac OS can be played on Power Macintosh systems through emulation. Apple Games Evangelist Mark Gavini estimates that the top 10 percent, including *Descent* from MacPlay, the top-selling game last year, are optimized to take advantage of Power Macintosh RISC performance. Native PowerPC code gives games access to a wide variety of performance enhancements, including larger screen sizes, more textures, faster frame rates, and stereo sound.

The conclusion was inescapable: It was time for Apple to support game development aggressively, making it easier to create the most technologically advanced, seriously fun games for the Macintosh platform. In turn, Apple would sell more of the hardware that the games run on. Apple then had to focus on what support was needed and how Apple could best serve the needs of developers.

The Evolution of Game Sprockets

The first step was pretty obvious; Apple decided that if it wanted to know what game developers needed, it would just ask them. Last December, Apple held secret meetings with 25 key game developers (a group that came to be known as the “Game Developers Advisory Council”) to show them what Apple was doing—namely, to demonstrate the Game Sprockets concept—and to solicit feedback. As a result of the meetings, Apple was able to tailor Game Sprockets to more closely meet your needs as game developers.

For example, Apple engineers changed the design of one of the sprockets, *InputSprocket*, based on comments received from council members, including LucasArts Entertainment.

please turn to page 20

Now It Can Be Told

Apple's First Annual Human Interface Design Excellence Awards

By Peter Bickford

It was Tuesday night of the Worldwide Developers Conference, and I found myself wearing a tuxedo in front of a couple thousand or so T-shirt-clad developers. It was the first ever Human Interface Design Excellence (HIDE) Awards: "the Heidis," a contest designed to recognize the best and brightest in the world of Macintosh human interface design. The mistress of ceremonies was Heidi Roizen, our new vice president of Developer Relations—who, as a bonus, turned out to have the perfect name for the role. I was playing the part of Vanna White, passing her the envelopes containing the names of the four winners. These were the products that could soon proclaim themselves as the Most Innovative Interface, Most Elegant Interface, Best Look and Feel, and Best Overall Interface of 1996.

For me, this was the big finish to a saga that began last December at the Human Interface group's holiday social. That was when Mark Auerbach from Human Interface Marketing and I finally got a chance to discuss an idea he'd had about holding some sort of "interface design contest." "Great idea!" I offered, little realizing that I was just then volunteering to coordinate the whole thing. I won't bore you with the details of what this entailed, but let's just say that staging a major industry award like this falls somewhere between planning a wedding and the Allied invasion of Normandy. On the bright side, at least nobody involved with the HIDE Awards wound up getting killed or wearing a tacky bridesmaid's dress.

But by Tuesday night, it was all over except for the ceremony. The judging was done, the trophies had arrived, preview commercials for the finalists had been running all day between sessions, and Heidi was ready to go. All I had to do was smile, try not to act too nervous, and avoid dropping the crystal trophies as I handed them to Heidi.

For those of you who missed it, here's a rundown of the finalists and winners, along with my own take on what makes each one special.

The Finalists . . .

After the submissions—almost a hundred of them—were reviewed, the top twenty or so products went before the full panel of five judges for final judging. There, they were scored from 1 to 100 on aesthetics, information presentation and interaction, depth of design, ease of learning, joy of use, appropriate use of Apple technologies, designing for users' needs, adherence to interface guidelines, and new ideas. It was an incredibly tight field, and often the winners of each category edged out their competitors by just a few points. Although they may not have picked up a HIDE award this time out, all of the following programs did a terrific job with their overall interface design.

Poser by Fractal Design

A finalist for Most Elegant Interface, Poser is one of those programs that seems more intelligently designed each time you use it. It's a 3D program with one specific purpose: to model the human body in various positions. This is historically one of the most difficult problems in 3D art, and previous tools tended to leave the result looking wooden and unnatural.

One of the problems is that the different parts of the body do not move alone—they work in conjunction with each other. For instance, if you take a model of a person whose arm is extended with the palm down, then turn it so that the palm is facing up, it's not just a matter of showing another portion of the forearm. In order to have the result look realistic, you must make small adjustments to related body parts, and even change the extension of the triceps muscles near the shoulder. Poser uses a built-in knowledge of these anatomical relationships so that when you move one part of the body, the related parts do the right thing. Just as important, Poser knows the range of motion for the various limbs, so that you can't unintentionally "break" an arm or leg by pulling it beyond its range.

Poser's thoughtful design extends into other areas of 3D rendering as well. For instance, other programs typically make you choose between the quick but blocky-looking "wire-frame" mode of viewing objects and the much slower shaded modes. In practice, this leads to constant mode switching as you attempt to quickly move your figures into their proper forms, while still trying to get a feel for how they'll look when they're finally rendered. Poser, on the other hand, is smart enough to switch into wire-frame mode temporarily whenever you move part of the figure, then go back to the previous shading mode once you release the mouse button. It's an obvious but brilliant insight that other 3D programs would do well to copy.

Retrospect 3.0 by Dantz Design

Retrospect 3.0 was a finalist for Best Look and Feel. If we'd had an award for Most Improved From the Original Version, it would have won, hands down. The creators, Larry and Richard Zulch, laughed as they recalled that several years ago, they got a registration card in from a group of users at Jet Propulsion Laboratories. Written in the comments portion of the card was something to the effect of "I'd say that your program (Retrospect 1.3) needs a rocket scientist to figure it out . . . except that we *are* rocket scientists and we can't make heads or tails of it."

From the very beginning, Retrospect held a dominant share of the Macintosh backup software market. Moreover, there was never a question of whether it had enough power and features to handle almost any backup situation. It still would have been vulnerable in the market, however, to a competitor with an easier, more accessible interface.

Rather than ignore this problem, Dantz got to work. Version 2.0 was a fundamental breakthrough, giving structure to the maze of options from the first version, and giving non-expert users access to the full range of its features for the first time. With version 3.0, they lowered the bar even further, eliminating engineering jargon, refining the program's look, and adding facilities such as "EasyScript," which walks users through the more complex aspects of automated and network backup.

Overall, Retrospect 3.0 is a terrific example of how careful design and progressive refinement of your product's interface can let you have your cake and eat it too: You can build in enough power to satisfy the most advanced users, while at the same time letting novices get their work done.

Passage to Vietnam by Ad Hoc Interactive

This gorgeous CD-ROM takes you on a journey across Vietnam with some of the top



photographers in the world. It was a finalist for Most Innovative Interface, primarily due to its unique navigational device known as the "quebe" (shown above). This is a six-sided cube that can be spun and rotated by the user. Each side gives the user access to different functional areas of the program, including background interviews, screen-to-screen movement, and program help.

For a multimedia program, the quebe offers several advantages over more conventional techniques for navigation. It obviates the need for a menu system, which, for all its strengths, can be confusing to novices. The quebe also scores over a row of buttons or navigational icons in that it occupies much less screen space, and even "spins away" off screen during the parts of the program in which its features are not needed.

Graphing Calculator by Ron Avitzur

Graphing Calculator was the Susan Lucci of this year's awards, being nominated for Best Look and Feel, Most Innovative Interface, and Best Overall Interface. By the narrowest of margins, however, this brilliant program never managed to grab the award.

If you've got a Power Macintosh, chances are you've already used Graphing Calculator. Programmed by independent contractor Ron Avitzur with help from Greg Robbins and Steve Newman, it's a program that fundamentally rethinks how mathematical equations are entered and displayed on a computer.

On a computer, standard practice to represent, say, "x to the second power," is to

represent it as " x^2 ". Type the same thing into Graphing Calculator, however, and it becomes " x^2 ". Moreover, you can easily enter the most complicated of formulas, including multiple levels of division, exponentiation, and functions—ones I won't even attempt to model on the word processor I'm using to type this article.

But that's not nearly the end to graphing calculator's inventiveness. Type in the equation " $y = x + 3$ " and watch the graph instantly appear. You can then reach in and grab the axes to see different parts of the graph. Plot a 3D function like " $z = x^2 + y^2$ " and the graph appears in 3D space, rotating slowly to show you all sides of the figure. Maybe its best trick, however, is that you can instantly balance any equation, like " $y = x^{(1-x)}$," simply by dragging any part of it from one side to another—like magic, the equation rearranges itself. Graphing Calculator sets a new standard for information presentation, and one which I suspect mathematical programs, spreadsheets, and word processors will spend years attempting to follow up on.

And the Winners Are . . .

Most Elegant Interface: *Alphabet Express* by School Zone Publishing

Alphabet Express proves that rich, enjoyable programs don't need to be complicated to



use. It may also show that any kid's program that stars a talking crocodile as a train conductor can't go far wrong.

The program is aimed at children 3–6 years of age and is designed to teach them alphabet skills. What makes the program so elegant is the absolute seamlessness of the interface. From the very beginning, it was

clear that the designers had taken enormous care to make sure the user never "fell out of the interface" and had to spend time wondering what to do next or how to control the computer. Even the program's error messages are done in the colorful, friendly style of the main program.

The way the program guides users is equally impressive. As the program begins, the lead character, an amusing crocodile dressed up as a train conductor, introduces children to the program. He leads them through the various parts of the program, and pops up to offer help whenever users' actions indicate that they may be becoming frustrated or confused. The world of "serious computing" would do well to study how *Alphabet Express* offers active assistance in a style that is natural, engaging, and genuinely useful.

Best Look and Feel: Adobe Illustrator 6.0

The Look and Feel award is given to the product that best embodies the Macintosh human interface guidelines and design philosophy. This year's winner was Adobe Illustrator 6.0, narrowly edging out Retrospect 3.0 and Graphing Calculator for the honor.

The veteran Illustrator comes into this competition with the benefit of a decade or so of continuous revision and improvement. Unlike so many of its competitors, however, Illustrator has avoided the practice of taking last year's model, recoloring its screens using whatever new graphical fad was peaking at the time (3D bevels, skinny scroll bars, tons of icons, and so on), and shipping it off as the new version. Instead, what strikes you when you first use Illustrator is how spare and clean its overall look is. Each dialog box, menu item, and tool palette appears carefully and deliberately arranged. New features have been incorporated into the overall design instead of appearing tacked on as an afterthought. As a result, Illustrator is simultaneously one of the most powerful drawing programs you can buy, and one of the least cluttered.

Most Innovative Interface and Best Overall Interface:

***You Don't Know Jack* by Berkeley Systems**

Finally, this year's big winner was *You Don't Know Jack*, a hip, irreverent program by Berkeley Systems that simulates a TV quiz show. Considering all the programs we reviewed for this year's contest, Jack might

Now It Can Be Told

continued from page 18

seem like a strange pick at first: there are only a few keys to press, it's not a "serious" productivity application, and it comes from a company best known for putting flying toasters on most of the world's computers. But interface design is more than just menus and commands: It's the entire experience the user has with the program. And total user experiences don't come any better than this.

Even while the CD is still loading, *Jack* is already drawing the user into its world. The speakers fill with backstage "chatter" giving bits of instruction to the members of the crew, cueing singers, and so on. While this is going on, a stage manager's voice asks the players to

type their names, and gives them short instructions on how to play. Shortly afterward, you hear the producer warning "ten seconds" before show time, and calling for the light crew to "lose the desktop." Bits of off-screen television commercials seem to conclude, and the screen goes dark. Then the show's theme music begins. By the time the announcer welcomes you and your fellow players to the show, you firmly believe in the game's reality.

If you start the game again, the stage chatter, commercials, and so forth, are all different. Even the tacky jokes the announcer makes as he tells you how to "buzz in with your answer" change from time to time.

And the really amazing part: all this care went into the CD loading/introduction time. You haven't even started playing yet.

You Don't Know Jack shows this same sort of detail and polish throughout every aspect of the program. There's never a time when some stray interface glitch or point of confusion causes you to fall out of the game's unique reality for even a moment. It has absolutely flawless execution from beginning to end, stays fresh for experienced players, and is so accessible that even my computer-averse dad gave it a go. It's engaging, fun, beautifully implemented, and a joy to use. All in all, it's a great interface and fantastic product.

*Till next time,
Doc*

TECHNOLOGY FEATURE

Game Sprockets

continued from page 17

Aaron Giles, lead Macintosh game developer for LucasArts, says, "We've been directly involved working with (Apple) people on InputSprocket."

David Chait, vice president of Product Design at Reality Bytes, worked with the DrawSprocket development team to ensure access to page-flipping services. (See "DrawSprocket" on page 21 for an explanation of page flipping.) Apple's philosophy throughout Game Sprockets' development has been to provide technology that will help you, not dictate what you must use.

Game Sprockets was designed to reduce game development time and effort by helping you overcome obstacles to Mac OS-based game design. A few game gurus have learned tricks for getting around these obstacles, and Apple has standardized these techniques and added some new capabilities so that others won't have to spend their time inventing them. Ben Calica notes that "games are the only applications where it's not only acceptable to break the rules; it's normal." As Dan Venolia, manager of the Apple Game Sprockets Group, puts it, "We're shredding half of *Tricks of the Mac Game Programming Gurus* book and building that stuff into the system." (*Tricks* is a book by McCornack, Ragnemalm, and Celestin, published by Hayden Books,

1995). By using Game Sprockets, you can concentrate on creating better games rather than worrying about what technologies and design-arounds you need. Aaron Giles of LucasArts says, "Game Sprockets is a fantastic idea. Anything that makes our lives easier is well worth the effort."

Six sprockets, described in the next section, were chosen to be the first in a series. Five of the sprockets are included in the current release, and the sixth (SpeechSprocket) will be available soon. Apple will add additional sprockets as needed. Apple has also dedicated engineering, marketing, and evangelism resources to help you quickly and effectively use Game Sprockets. As Mark Gavini says, "Apple has gone from having one person—a games evangelist—to having a triad of people, one of whom leads a team of engineers, dedicated solely to making great games possible on the Mac OS."

Apple is not in the game business; it's in the business of helping developers create games. All game companies—from big-time developers to small firms, individuals, and shareware companies—are encouraged to use Game Sprockets. You won't have to pay to use it, since Game Sprockets is provided royalty-free to developers. Apple plans to profit by selling more Macintosh computers for playing all the cool new games you're going to design.

Bill Dugan of MacPlay is "extremely impressed that Apple has kicked off the Game Sprockets project. The sprockets demonstrate

that Apple is aware that games are a really important segment of the software market and that games are the main reason why consumers buy computers in the first place. This focus on Apple's part gives us more confidence that the Mac will be a great game platform in the years to come."

So What Is Game Sprockets?

As I've already said, the Game Sprockets SDK is a set of libraries that enhances your ability to create games for the Macintosh. Apple designed Game Sprockets to be compatible with any PowerPC processor-based computer running the Mac OS, including forthcoming systems based on the Common Hardware Reference Platform. Further, by writing to Game Sprockets now, it will be easier for you to move to Mac OS 8 later, because the calls under System 7 and Mac OS 8 will be the same. To give just one example, InputSprocket and the Mac OS 8 input driver hardware model were designed in tandem.

Once you make the move to Mac OS 8, your game will be able to put to good use the multitasking capabilities of the new version of the Mac OS. Games currently use a kind of multitasking by making use of the interrupt state—for example, to separate rendering from game state maintenance. The multitasking capabilities of Mac OS 8 will enhance system performance in this area and make it easier for you to separate individual game tasks so that they run preemptively in the protected memory space.

This section introduces you to each sprocket in more detail. Every game can benefit from your using one or more of the sprockets. You can “mix and match” sprockets to provide the features needed for your title, and you can also use them in nongame applications. Finally, you can deliver the sprocket libraries free of charge to your customers along with your game.

Sprockets in the current SDK include the following:

- DrawSprocket, which provides multiple buffering and display control
- NetSprocket, which handles Internet connectivity and multiplayer game playing
- InputSprocket, which provides interfaces with input devices, such as joysticks
- QuickDraw 3D RAVE, which supports software and multiplatform hardware graphics rendering and graphics acceleration hardware
- SoundSprocket, which serves as the interface with the Sound Manager and can make sounds from your game seem to be coming from specific visual locations (through technology called *3D sound localization*)

In addition, as I mentioned earlier, there is a sixth sprocket under development—SpeechSprocket, which provides voice command and control.

You'll find that some of the sprockets allow you to implement really cool operations that were previously very hard to accomplish. One example is QuickDraw 3D RAVE—a layer between your game's rendering engine and the hardware that transparently allows software rendering to be accomplished by any compatible hardware acceleration card. According to Ben Calica, “You should start writing to RAVE now so your games can support hardware acceleration, which will soon be pervasive.”

Other sprockets simplify the implementation of features that have traditionally been difficult to handle. For example, DrawSprocket provides a game-oriented graphics model. Everything on the screen is a game interface element—there are no menu bars, control strips, system input windows, or rounded corners. Further, DrawSprocket lets the game reconfigure the video modes (for example, resolution, depth, and timing) to suit the content, without causing icons and windows to move. You can also easily take advantage of double buffering, which few developers know how to use even though it's been around a long time. Another sprocket, InputSprocket,

lets you handle joysticks as plug-and-play devices.

Specific descriptions of each sprocket follow.

DrawSprocket

DrawSprocket lets you safely take over the screen, customize it to best fit your game's content, and display graphics at high frame rates. With DrawSprocket, your game can

**Apple is not in the
game business;
it's in the business of
helping developers
create games.**

easily configure the display, taking advantage of any system-specific display features. Any game features not directly supported by the system hardware are emulated by software. DrawSprocket automatically gives you the best possible performance (such as smoother playback and faster frame rates) on the system the game is being played on.

DrawSprocket takes care of some of the current problems of dealing with system features. A blanking window lets you hide the desktop during game play. After a user configures the display and is ready to begin playing the game, DrawSprocket completely hides the desktop, menu bar, and other system resources and changes the screen to the best pixel depth and resolution. (Previously, if users changed screen resolution to play a game, they could interfere with the desktop.) DrawSprocket also makes it easier to deal with multiple monitors, and its next release will support cursors larger than the 16-by-16-pixel cursor now commonly supported. When the game is over, everything is automatically restored to its original state.

When you draw scenes in a game, there are usually intermediate stages in the process that you don't want the user to see. With DrawSprocket's double and triple buffering, you

can display the contents of the front buffer while drawing the next scene into back buffers; when a scene in a back buffer is finished, it is transferred to the front buffer for display. Traditionally, the back buffer is in system memory and the front buffer is in video memory. In some systems, however, both buffers are in video memory, in which case you can have your game simply point to the buffer containing the next screen refresh, a technique known as *page flipping*. For the next release, the DrawSprocket team is working on high-performance “blitters,” which copy pixels from memory to the screen.

Other DrawSprocket features include gamma fading, by which colors fade nonlinearly so that bright colors don't remain visible after dark colors disappear, and fade-outs to colors other than black. You can provide overlays and underlays, set a maximum refresh rate so that your game runs at a constant speed, and manipulate colors in a lookup table. Future releases will include pixel scaling with optional linear interpolation and support for stereoscopic graphics. DrawSprocket also has a special debugging feature that keeps system resources visible, even behind the blanking window.

NetSprocket

Multiplayer games are what's hot these days. NetSprocket handles the transport of game data, making it easy for you to design multiplayer games that operate over internal or external networks. You use NetSprocket to make your game known to prospective players on the network, collect players, and distribute game data to all participants. Any network protocol supported by Apple's Open Transport networking architecture—including TCP/IP, AppleTalk, and modems—can be used. (For your information, in Japan, Open Transport is available only on Mac OS-compatible computers that use the PCI bus; this may affect whether or not you decide to use the NetSprocket technology.) With TCP/IP, your games can operate over the Internet. When Open Transport supports other network protocols, NetSprocket will support them as well; you won't need to change your game code at all.

The goal of NetSprocket is to dramatically decrease the development time of multiplayer games. Most game developers aren't networking experts, and adding networking capability to games can be a lot of work. If you use NetSprocket, you don't need to know anything

about networking; instead of having to use the Dialog Manager, the List Manager, AppleTalk, sorting functions, and so on, you can let your customers host or join games by including just one function call.

You can easily change your single-player games to work over a network. In less than a day, Jamie Osborn, Apple's NetSprocket development engineer, changed Glypha III, a shareware game by John Calhoun, from a single-player game to a two-player game that can be played over the Internet.

NetSprocket automatically keeps track as players join or leave a game, updating its routing information and informing the other players of changes to the list of participants. It includes an interface for hosting multiplayer games and having additional players join in; you can use it "as is" or as a prototype to simplify development of your own interface.

Message routing is transparent to the game, and you can specify best-effort or guaranteed delivery, depending on your needs. NetSprocket also handles fault tolerance, a critical component of any network game architecture, by automatically adjusting for the disappearance of a player due to a game malfunction or network failure.

A major problem with networked games is the delay between the time when one player does something and the time when other players see the result. This delay, known as *latency*, is caused by uncertainty in the delivery time of data packets over a network. Latency is somewhat network-dependent; while it may be possible to define and, to some degree, control delays on smaller, internal networks, delay over the current Internet is impossible to predict. Some games (such as battle games) must have low latency to be effective, while other games (chess, for example) can tolerate longer delays. The latency problem has not yet been solved, but NetSprocket will support any improvements that come along to reduce latency. Because NetSprocket hides the way computers are connected to each other, more sophisticated topologies (for example, multiple server architectures) can be integrated as NetSprocket evolves.

Another interesting and useful benefit of NetSprocket's transport-independent messaging is that people playing a game over one network (AppleTalk, for example) can join another group playing the game over a different network (say, the Internet) through their respective servers, without needing a direct

connection. That means if my computer is connected by AppleTalk to a server that has an Internet connection, I can play with someone on the Internet even though my computer has no direct connection to the Internet. Game companies who offer dial-in pay service with guaranteed bandwidth can use this feature to connect those customers with other players using free, no-guarantee service over the Internet or another network.

InputSprocket

InputSprocket provides an easy-to-use, consistent way to deal with joysticks and other input devices, including the new breed of input controllers. If a Mac OS game employs InputSprocket, a player can use any compatible input devices to control it, simply by plugging them into the computer and playing. Aaron Giles of LucasArts thinks InputSprocket is "fantastically important."

Currently, each joystick has its own interface to the system, and each one must be emulated on the keyboard or mouse. When you build InputSprocket into your game, device drivers will give standard information about device controls. This makes it easy for games to configure control options, and in general simplifies interactions between games and input devices. With InputSprocket, your game can configure, synchronize itself with, and track data from input devices.

InputSprocket bases the communication between your game and an input device on certain elements, including the information

that describes the input device's controls, such as its axes, thrust levers, buttons, and direction pads. During play, the game automatically determines the state and transitions of the elements through polling or a simple event queue.

SpeechSprocket

Speech recognition can add a new dimension to your games, engaging your customers and enhancing their experience. Games that offer powerful and richly immersive environments still use "point and click" control, which is often not the best or most convenient method. You can use SpeechSprocket to provide "ask and tell" control, which will make your games stand out and rise above other titles competing for shelf space. You can add SpeechSprocket to your games quickly and easily, with maximum efficiency and performance, minimal impact on production schedules, and zero impact on your tools budget—SpeechSprocket, like all the other sprockets, is free.

SpeechSprocket is a driver for InputSprocket that allows users to control a game by spoken commands. The driver communicates with Apple's Speech Recognition Manager, an extension to the Mac OS. So, to use SpeechSprocket, you need to include the SpeechSprocket driver, InputSprocket, and the Speech Recognition Manager with your game.

The Speech Recognition Manager is available now, providing a full-featured API that can

Marketing Your Games

Mark Gavini, Apple's games evangelist, has a few ideas to help you market your games:

- Make sure your company is well represented in the distribution channel, both retail and mail order.
- Advertise your games in Macintosh and game publications.
- Use game-oriented media such as Inside Mac Games, a CD-based electronic magazine and Web site (<http://www.imgmagazine.com/>).
- Distribute playable demos through the various online services and CD collections. If it's not possible to distribute a playable demo, post a QuickTime trailer featuring some of the coolest scenes from your game. Provide your demo or trailer to publications.
- Let Apple help you. First, send several copies of your game to Mark Gavini and Ben Calica. Include a script that will allow the best parts of the game to be viewed in two to five minutes. Mark and Ben will take a look at it and, if they feel they can help, they'll put it in front of press contacts and other people at trade shows and executive keynotes. They prefer games that showcase specific Apple technologies, such as QuickTime, QuickTime VR, or QuickDraw 3D RAVE, or games that were developed using Game Sprockets. Mark promises, "If your game looks great on a Mac, we're gonna make you look good in as many places as possible."

be used to quickly and easily add voice command and control to any application—and it's free. Using the Speech Recognition Manager, you can create sophisticated hierarchical vocabularies that you preconfigure. Its Toolbox includes many useful features, including "push-to-talk," which allows you to enable the microphone by clicking a button or pressing a key, and a feedback window, which provides a crucial user interface element. SpeechSprocket, which has been optimized for games, implements a one-to-one mapping for each function, so a voice command will be associated with a particular button or function key.

You can obtain the Speech Recognition Manager extension, technical documentation, and software development kit from Apple's Speech Web site (<http://www.speech.apple.com>). SpeechSprocket is currently under development and will be released soon on the Game Sprocket Web site.

If you use SpeechSprocket, you can add a base level of voice command and control to your game with minimal effort. It lets your application handle voice commands in the same way as any other input. You can build voice commands into a new game or add them later to a game not originally designed to handle them. When a player sets up the game, a dialog box presents a speech option for each command. The player types in the spoken phrase to be used (for example, "rocket launcher") for a command. After that, each time the player says the magic words, the command is executed.

SpeechSprocket takes advantage of the fact that the PowerPC processor generally has enough extra cycles to implement speech recognition in your games. The number of cycles required depends on the system used, the acoustical environment, and other tasks going on at the same time. You can control the percentage of total processor cycles devoted to speech recognition in your game. If your game is CPU-intensive (for example, if you're delivering a game with a high frame rate), the Speech Recognition Manager provides a call ("SRIdle") that you can use to adjust the percentage of CPU cycles devoted to speech recognition.

To judge how well SpeechSprocket works, you should try it out; it's very easy to implement. Performance is optimal when a limited number of multiword phrases are used. For example, you'll get excellent results using a vocabulary of 20 words, which should be more

than enough for many game applications. Longer commands, like "rocket launcher" or "machine gun," are easier to recognize than shorter ones—"gun" or "rock," for example.

SpeechSprocket currently does not include echo cancellation, so in most games you probably won't want to leave the microphone on all the time—it's better to enable it through a keyboard command or on-screen button. This "push-to-talk" capability is part of the Speech Recognition Manager Toolbox, and can be easily implemented to allow your games to use speech recognition in noisy, real-world environments. There is some latency, but the recognition is generally pretty close to realtime.

Apple's speech program wants to hear from you—let them know what you think of SpeechSprocket and what you'd like to see in future releases. You can e-mail them at plaintalk@applelink.apple.com. If you use SpeechSprocket in a game, they, along with the Game Sprockets people, can help you promote it.

QuickDraw 3D RAVE

Apple devised QuickDraw 3D to provide a standard way of adding high-quality 3D renderings to personal computer software. Game programmers, however, are less concerned with high-quality graphics than they are with high frame rates. Hardware acceleration can increase frame rates, but current hardware acceleration cards are all different. If you build QuickDraw 3D RAVE into your game, it will be able to use any RAVE-compatible card for graphics acceleration; if no card is installed, rendering will be performed in software, without your having to do anything else to your program.

RAVE (Rendering Acceleration Virtual Engine) is a low-level 3D driver for the Power Macintosh computer. Put another way, it is a software layer designed to support the low-level rasterization operations required for interactive 3D rendering. While in some respects RAVE is similar to a 2D drawing API (application programming interface), there are some important differences. First, 3D drawing requires a Z (depth) value, which is used to perform hidden surface removal. Second, it requires support for double buffering. (See the earlier "DrawSprocket" section for a description of double buffering.) Finally, special 3D rasterization modes (such as texture mapping) must be supported for 3D rendering to work.

RAVE can be used in several ways. For game developers, RAVE provides a highly optimized,

flexible way to access hardware acceleration. It also allows hardware accelerator card vendors to ship 3D acceleration cards that work with a variety of games and other applications without requiring that the user do anything but plug in the card. In these cases, RAVE operates as an application programming interface (API). In a broader sense, RAVE provides a hardware abstraction layer that allows system software (such as QuickDraw 3D) to utilize a variety of hardware without code changes; in this sense, RAVE operates as a system programming interface (SPI).

RAVE was designed to provide the speediest 3D graphics rendering possible. To maximize performance, it provides the minimum possible overhead between your game and the drawing engine. Calls to a RAVE drawing engine do not pass through an intermediate manager layer; instead, the game makes calls directly to the selected rasterizer's code. Thus, calling a rasterizer through RAVE provides the same level of performance as linking the engine directly with the game.

QuickDraw 3D RAVE also works with the Windows 95 and UNIX® operating systems. (Although the other sprockets do not work across platforms, they've been designed so that features can be ported easily to other platforms.) Apple is working with accelerator card makers, including Diamond Multimedia and ATI Technologies, to provide RAVE accelerator cards that will work on any computer that supports the PCI bus, with separate drivers for each platform. Because QuickDraw 3D RAVE accelerator cards will be on the market for the 1996 holiday season, you'll want to be sure your game supports RAVE by then so that your customers have access to blazingly fast 3D rendering.

SoundSprocket

Better sound is one of the best ways to improve a player's overall game experience; research has shown that improving sound quality will also enhance the perceived quality of other aspects of a game, including the visual components. When built into your game, SoundSprocket lets you improve your game's sound by providing 3D sound localization capability for sounds played through Sound Manager channels. That is, SoundSprocket can make it seem as if certain sounds are coming from specific visual elements within your game.

For example, the sound of an on-screen car can move along with the car—up, down, left,

right, or closer or farther away from the user. The 3D spatial filtering that makes this happen is performed in real time using either a small share of the PowerPC processor's cycles or a separate processor card. Your game does not need to know how the processing is being implemented.

For 3D localization, the user and one or more sound sources are assigned positions and velocity vectors in 3D space, and various types of processing, including Doppler effect, distance attenuation, reverb, and spatial positioning, are carried out on the sounds. In addition to Apple's localization processing, you can use other algorithms. SoundSprocket provides high- and low-level interfaces for managing the sound filtering: The low-level interface lets you send messages to the sound components, and the high-level interface provides a collection of functions that help you create those messages. The high-level interfaces work together with, but don't depend on, QuickDraw 3D. To specify how much of the CPU is to be devoted to filtering, your game can provide a set of algorithms of varying complexity. If many CPU cycles are available, you can specify very high-quality filtering; if the game's frame rate is affected, you can use a less computationally intensive algorithm.

Who's Using Game Sprockets?

A lot of game developers are using or looking at Game Sprockets. Many are working with Apple to optimize existing sprockets and plan future ones. Here are a few examples of developers already at work with Game Sprockets.

- Reality Bytes' Havoc was the first game on the market to use QuickDraw 3D RAVE. Jon Chait, president of Reality Bytes, explains why: "We used RAVE because Apple invited us to get involved way before the game came out. Game code is about as complex as you can get, and Apple's desire to work with us in defining how to interface with their code lowered the barriers for everyone. Apple's willingness to disclose new technologies before they're 100 percent done allowed us to have the greatest impact—using RAVE allowed us to get to the marketplace earlier than we would have otherwise. If you work with developers early, you get a great result."

Jon Chait believes that the entire game community will benefit from Game Sprockets, because it lowers the cost of (and many technological barriers to) adopting and maintaining a code base on the Macintosh platform.

Because Game Sprockets gives game developers access to so much Mac OS technology for so little work, they can write games from the ground up for the Mac OS rather than porting them. "[Game] Sprockets is very much aligned with Reality Bytes' development goals," says Chait. "We don't do ports; we do great games that attempt to exploit the latest and greatest technologies for the Mac."

Chait feels that, before Game Sprockets, Reality Bytes had a competitive advantage because its staff knew all the hacks that could make its games stand out from the crowd. Game Sprockets gives all developers access to those hacks, leveling the playing field and opening it to newcomers. Game Sprockets also lets high-end Mac OS developers like Reality Bytes feel confident that current and future hardware will support them and forge ahead with new technologies and capabilities.

- LucasArts plans to release Outlaws, a Sprockets-based game set in America's Old West, in time for the 1996 holiday season. For 1997, the company is planning a Game Sprockets-based version of Jedi Knight Dark Forces II, the sequel to the popular Dark Forces. Aaron Giles of LucasArts says, "Game Sprockets provides a lot of needed technology and simplicity. We're looking forward to using sprockets, throwing away all the hacks we're using now, and having a nice standard interface that's compatible with current and future versions of the Mac OS."

- Sierra On-Line demonstrated Indy Car Racing II at Apple's just-concluded Worldwide Developers' Conference. Indy II is a full-simulation racing game that uses the Speech Recognition Manager to provide at least 50 voice commands. Users can change a tire, check the tire temperature, change the spoiler angle, fuel the car, and do lots more without having to use time-consuming, hard-to-learn keystrokes. As a result, the game experience has been enhanced. Rick Genter of Sierra says, "Apple's Speech Recognition Manager is giving us a feature in our Mac OS platform game that games on other platforms can only dream about."

- Bungie Software's Marathon II, running with QuickDraw 3D RAVE, was shown at the last Computer Game Developers Conference. Eric Klein of Bungie (and the former Macintosh games evangelist for Apple) says, "We think that Game Sprockets has a lot of potential. We're evaluating it now and looking forward to its release. We've done some

technology demos with some of Apple's strategic partners, including ATI (who made the QuickDraw 3D RAVE accelerator card used in the demo), and the results look very promising."

- MacPlay has implemented DrawSprocket in Star Trek: Starfleet Academy, a 3D space-action game, scheduled for release by the end of this year. MacPlay's Bill Dugan notes, "We're getting really high frame rates with DrawSprocket; plus, if some user out there has a weird video setup that allows really fast drawing to the screen, DrawSprocket will accelerate our game without our having to know anything about the user's system. DrawSprocket lets us leverage Apple's hardware automatically."

- Domark Software is preparing to release Flying Nightmares 2, the sequel to the best-selling Flying Nightmares flight simulator, as well as Tank Commander and Deathtrap Dungeon, all of which will use NetSprocket.

What Should You Do?

If you're convinced by what I've said about Game Sprockets, and you want to start working with it, the first thing you'll want to do is to visit the Game Sprockets Web site (<http://dev.info.apple.com/games>) and download the SDK. When you register at the site, Apple will keep you updated on news about Game Sprockets; you can also revisit the site regularly to keep up-to-date.

Think about how you can use Game Sprockets in your new game or to update current games. The five existing sprockets are final and can be downloaded and used in your games. SpeechSprocket isn't yet final, but you can look at it to get an idea of how you can use it. More developed code is coming soon: Apple is aware of your shipping schedules for holiday season sales, and plans to have usable code available in plenty of time to meet those schedules.

Game Sprockets is hot stuff—it's going to make your life a lot easier and your games a lot better. Check it out and see for yourself. ♣

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

OpenDoc Human Interface FAQs

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

Q. I'm working on a viewer that doesn't support any selection of the content. I plan to support the Copy command, but do I really need to support the Select All command that your guidelines for viewers say to support?

A. Yes, we want users to have a predictable user experience as they move from one part to another. If your part copies all the data to the Clipboard when users invoke Copy, they may wonder why this doesn't happen in other parts as well. Your part editor also needs to provide visual feedback that the user has selected all the content—for example, for a picture editor, you might use the "running ants" border around the content of the part.

Q. From reading the OpenDoc Human Interface Guidelines, I'm not sure if I should remove Cut, Paste, and Paste As from my viewer. One place in the guidelines says to disable these commands, but then later the guidelines say to remove all editing commands from viewers. Which is it?

A. Generally we recommend that you remove controls that will never be available to the user. However, removing the standard menus is not a good idea. Partially this is because of the Mac OS architecture. Without position-independent menu IDs, part developers must register command numbers and associate a command number with a menu and item. If a part alters the base menu bar, it must unregister the standard commands and reregister them for the shell to continue to work. This is a lot more work and is prone to error. In addition, from the human interface perspective, there is a benefit to leaving these menu items in the Edit menu—that way, your viewer's menus are consistent with the menu layout that users have come to expect.

Therefore, we recommend that the commands Cut, Paste, and Paste As be disabled instead of removed. We will update this section of the guidelines to clarify this point.

Q. What happens when a user tries to make changes to a read-only document?

A. Today the user would be able to make changes and use Save As to create a new document with those changes. OpenDoc uses Save a Copy instead, so while the changes are saved in a new document, the current document remains open. Since changes to a read-only document cannot be saved, the user must close without saving.

This is a tricky question. We think it would be nicer if your part didn't allow changes, but you may have to make lots of checks within your code to implement this. We know this is difficult to implement, and we certainly can't require this behavior. Using the OpenDoc Development Framework (ODF) will lessen this problem, because it makes some of those checks for you.

In addition, you should know that we anticipate adding access controls to individual parts in the future—for example, the ability to control which users have permission to see, modify, move, and delete any part. So, in preparation for these future features, consider structuring your code so that it checks for access before allowing operations that would modify a part. Today if you call `ODDraft::GetPermissions`, you will get the current permissions.

The root part's editor may choose to put an adornment in the window frame to indicate read-only status—for example, the "crossed pencil" adornment in the lower-left scroll bar. (OpenDoc itself cannot do this, because it doesn't know what controls the root part may put in the window frame.)

For more information, see the sections on read-only documents discussed in the *OpenDoc Programmer's Guide*, pages 247 and 551–552.

Q. My part supports neither linking nor embedding. Do I need to support the Paste As command?

A. The Paste As command allows the user to override the OpenDoc merge/embed rules and specify the translation and creation of links. In your case, you might simply disable this command because you don't support linking and embedding. However, your part may benefit from translation. We are looking at how to make translation even more powerful in the future, so we believe you should keep the Paste As command for that purpose.

Q. What is the desired behavior for Open Selection for intrinsic content?

A. Let's look at an example. Suppose you have a spreadsheet part. Now suppose that the user selects a couple of cells that contain only numbers and formulas (intrinsic content to the spreadsheet) and then invokes Open Selection.

According to page 248 of the *OpenDoc Programmer's Guide*, you don't have to implement this behavior, but if you do implement it, you should do so as View In Window. Your question caused us to rethink this and we discovered the result is unpredictable. Here's why.

Suppose some other cells contain embedded parts, such as chart parts. If the user selects a cell that contains only a chart part and invokes Open Selection, it is clear that a new window containing only that chart should be opened. Likewise, if the user selects two charts and invokes Open Selection, there should be two new windows—one for each chart. However, if the user selects some numbers (intrinsic content) and a chart—what should happen? Clearly it is important to see the chart in a separate window, yet if you follow the guideline on page 248 of the *OpenDoc Programmer's Guide*, you'll have a conflict. To avoid this conflict, we recommend *not* implementing Open Selection for intrinsic content.

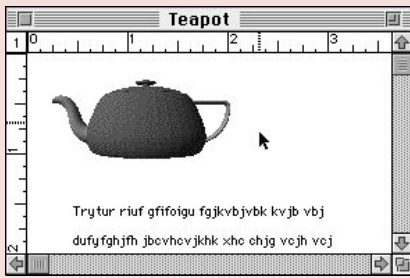
So, in this situation (a mixture of intrinsic content and embedded content) we instead recommend that your editor perform the Open Selection operation on the embedded parts and ignore the intrinsic content within the selection.

Q. I'm working on a QuickTime movie viewer. How should I handle the movie controls of QuickTime?

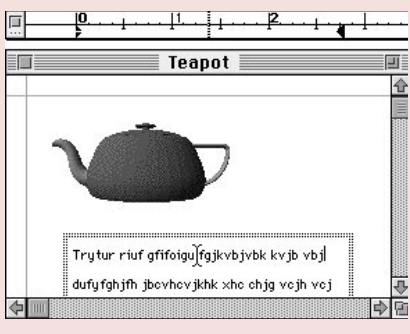
A. If you look at different applications today, you'll see that they handle this differently. The reason for this is that they were designed for different situations. For example, in a WYSIWYG word processor you typically don't see the movie controller, because showing it would interfere with the layout. In another container—for example, in a Web page or a kiosk application—the layout is less important than easy access.

We have been working on a mechanism for the container to give hints to embedded parts. For example, when a container is in "layout"

Root part rulers



Rulers for embedded parts



vs. “run” mode, embedded parts may behave differently. We think this same mechanism can be used to give hints to embedded parts about whether layout or access is most important.

Unfortunately, this hint mechanism isn't available yet. So, your part may want to have a setting that controls whether the controller is in a floating window or in the window frame (see pages 579–580 of the *OpenDoc Programmer's Guide*). We think that, by default, con-

trollers for technologies such as QuickTime and QuickDraw 3D should be in floating windows. However, when your part editor is the root window, you may place the controls in the window frame.

Q. What about overlaid frames?

A. This brings up a problem we've discovered. In the *OpenDoc Programmer's Guide* (pages 531–533) and in the OpenDoc Human Interface FAQs column in the October 1995 issue of *Apple Directions*, we said to use overlaid frames or floating windows. We now know there are a couple of cases that we failed to consider, so we no longer recommend using overlaid frames.

So, you may wonder what to do instead. If your part is the root part, you can put controls inside the window. (In the next question, we'll talk about what happens to the controls when the root part is inactive.) If your part is embedded, continue to use a floating window for all controls as described on pages 531–533.

Rulers are a special case of controls. They're special because they indicate layout information, so the user needs to readily see and access the ruler. The ruler also needs to indicate the relative location of the pointer (cursor). ClarisWorks does this in its rulers. If your part is the root part, it can put its rulers inside the window, as shown in the figure “Root part rulers.” Notice that the ruler includes indicators of the position of the pointer. We encourage you to include this very useful feature.

When your part is embedded, we recommend that the ruler be placed at the top of the

window, which is often at the top of the screen (see the figure “Rulers for embedded parts”).

Use this ruler for all your editor's parts. Simply reset the origin, tabs, and similar elements as each part becomes active. For example, when the user activates a different text part in this document, the text editor will simply update the ruler to match the origins and similar elements of the new text part. A side effect of this is that if the user should move the document window or ruler window, the ruler will still make sense.

When you place the floating window, you may need to move the document window down a bit. You may also have to resize the window so that the bottom isn't off the screen. We're working on a solution to replace overlaid frames, but we don't have anything to share yet.

Q: What happens when my root part has a control inside the window and my part becomes inactive?

A: If the control is a scroll bar, nothing happens. The scroll bar still scrolls the whole document. If an embedded part is active and the user scrolls the document, do not activate the root part. However, for something like a ruler, don't remove it. Removing it will cause the document to reformat. You should dim the ruler itself, by painting it light gray. Also, remove any indicators that apply only to the root part—for example, tab indicators and numerals.

There is at least one more situation in which you shouldn't disable or dim your control when your part becomes inactive—namely, for a tool bar that controls the entire document. We'll talk about this in next month's column. ♣

CD HIGHLIGHTS

continued from page 16

cross-platform software. This release includes OpenDoc 1.0.4, documentation, development essentials, and sample code.

OpenDoc is both a specification and a set of technologies used to build components. Developers who build their applications in accord with the OpenDoc architecture can provide software that integrates and cooperates with other OpenDoc software.

OpenDoc Development Framework

This folder contains the OpenDoc Development Framework (ODF) Release 1, which supports

OpenDoc 1.0.x for Macintosh only. If you want to do Windows part development, you should use the ODF 1.0d8 release, which supports both OpenDoc Developer Release 1 (DR 1) for Windows and OpenDoc DR 2 for Macintosh. The OpenDoc API (application programming interface) changed so much since DR 1 that it became impractical for Apple Computer to continue to support both APIs in the same code base. Apple is still committed to supporting OpenDoc for Windows when that API reaches a level that allows Apple to do so.

Since the 1.0d11 release of ODF on the OpenDoc DR 4 CD, portions of ODF have been moved into a SOM-based shared library

to reduce the size of resulting sample parts and provide better code reuse. Currently, this library contains most of the foundation and OS layers. As development continues, Apple will attempt to move more of the framework into this shared library. You should be aware that, as a result of this move, this release requires your parts to be dynamically linked to the ODF library.

See the ODF D11 Read Me document for further details.

PC Card Manager

This folder contains the PC Card Manager Software Development Kit for PowerBook 5300 and

190 series computers; it can also be used for PowerBook 500 series machines with PCMCIA expansion modules. This SDK includes system interfaces, a PowerPC library, full Acrobat documentation, plus sample code to get started with PC card software development.

QuickDraw 3D 1.0.6

QuickDraw 3D is Apple's cross-platform 3D graphics library. In this folder you'll find the 1.0.6 release of QuickDraw 3D for the Mac OS.

QuickDraw 3D goes beyond providing 3D graphics; it provides an integrated solution for both end-users and developers. QuickDraw 3D encompasses a standard file format (3DMF), an acceleration layer, input architecture, high-level geometries, and extensibility.

QuickDraw 3D is a shared library for PowerPC processor-based computers running the Mac OS. The API is in C, and provides support for development using C++. QuickDraw 3D allows for immediate mode and retained rendering. The API is object-based and provides a large number of geometry types. The file format accommodates both text and binary modes, with encoding for "endianness" so that files can be transported to other platforms.

Snippets Update

This folder contains six new and revised code snippets from DTS.

- *Calc ControlRgn.* This snippet shows how to call a CDEF to have it return the control's region. It works for both PowerPC and 680x0 code.

- *LaunchWithSize.* The LaunchWithSize snippet launches an application using a memory partition that it sets rather than the one in the application's 'SIZE' resource. LaunchWithSize patches Get1Resource so that it watches for the 'SIZE' resource, and then changes it so that LaunchApplication creates a memory partition of a different size than that requested by the application. This is useful for multimedia developers who ship software on CD-ROM but want to adjust the size of their 'SIZE' resource depending on available memory, CPU, or other factors.

- *ModifyMouseAccl.* ModifyMouseAccl demonstrates the use of the Cursor Device Manager to adjust the mouse acceleration setting for the standard Apple mouse. This quick sample obtains the CDM record associated with the standard mouse, saves the acceleration settings, then replaces the settings with some custom values defined in the program. You can also specify a tablet-like setting or one that

makes the mouse extremely sensitive to movement; to do so, you set NOACCLCURVE to the desired value. A discussion of the acceleration resource is provided in the "Space Aliens ADB" technical note.

Once the new values are set, the program calls CursorDeviceSetAcceleration so that the Cursor Device Manager calculates the new acceleration curves to be used by the cursor draw routine. After waving the mouse around to see that the new acceleration curve is in effect, you press the mouse button to have the program restore the original acceleration settings and quit.

Note the presence of the included "glue" file CursorDevices.Glue.c. This glue file is used in conjunction with the CursorDevices.h header file, which has been available since E.T.O. 18. Use this glue file to link a native PowerPC processor-based program with the Cursor Device Manager routines.

- *NoPowerOffKey 1.2.* Starting with System 7.5, the Power key on the keyboard allows users to turn off the Macintosh. However, this behavior isn't appropriate for some classroom or kiosk settings, so the NoPowerOffKey extension prevents the Power key from turning off the Macintosh. To use it, drop it into your System Folder and restart.

Version 1.2 fixes a minor concern, removes one line of assembly-language code, and adds a Finder help string that provides a help balloon for the extension. (Technogeek for "minor concern": My definition of the UniversalProcPtr for the PowerOff gestalt function didn't include space for a return code. I was using a pre-release version of the System 7.5.3 technical note. It shouldn't have caused any problems, but I'm being picky.)

- *PrintDialogMagic.* This snippet is a short example of how to prevent Print dialog boxes from waiting for user response (which is useful for automated printing).

- *SndPlayDoubleBuffer.* This snippet is a complete example of how to use SndPlayDoubleBuffer to read a sound from disk. It shows how to parse an AIFF, AIFC, .au, 'snd' resource, and WAVE header; how to read a resource map without using the Resource Manager; how to use completion routines and callbacks; and how to stop and start playing a sound from anywhere within the file.

SndPlayDoubleBuffer reads and plays AIFF, AIFC, .au, 'snd' resources, and WAVE files, while allowing random access within a sound, much like QuickTime does. It uses the SimpleApp framework to factor out the Macintosh inter-

face code, leaving only the important sample.

SoundApp

SoundApp is a sample application for demonstrating the Sound Manager. It comes in two parts: SoundApp.c and SoundUnit.c. SoundApp.c handles the application's management of memory, errors, user interface, and so on. It requires the use of SoundUnit.c to handle all of the sound routines. SoundUnit code handles the Sound Manager part of the application. You can use this stand-alone unit in your own application.

Speech Recognition Manager

This folder contains version 1.5 of Apple's new Speech Recognition Manager. Version 1.5 of the Speech Recognition extension embodies the first officially supported developer release of the Speech Recognition Manager.

The draft documentation included in the SR 1.5 Documentation folder describes version 1.5 of the Speech Recognition extension and the API and Toolbox it embodies. Version 1.5 contains many features that were not available in version 1.4.1 or earlier versions of the extension.

Speech Synthesis Manager

This folder contains information that tells you how to use Apple's English and Spanish text-to-speech software to let applications talk. It replaces an earlier package entitled Text-to-Speech, and contains everything you need for getting started with Apple's Speech Synthesis Manager: documentation, interfaces, new sample code, libraries, speech synthesis system extensions, and installer sources.

You can obtain a free license to distribute this software with your applications. Contact Apple's licensing group for more information about licensing Apple software (phone: 512-919-2645; AppleLink: SW.LICENSE; Internet: SW.LICENSE@applelink.apple.com).

STD File Saver 2.1

STD File Saver is a sample print driver written in C. It's based upon the STD File Saver 1.1 sample in Pascal that has been the only print driver sample available since 1991. Because it writes to PICT resources, it's not too useful as is, but it does illustrate how to write a print driver with a minimum of overhead.

Alex Doshier
Developer CD Leader and
Online Content Librarian
devcd@apple.com

Business

Feature: By augmenting traditional search tactics with some unique Macintosh developer community resources, you can significantly increase the chances of finding just the right programmers for your company. This article reviews a number of hiring strategies that have worked well for other successful Macintosh software companies.

Finding and Hiring Great Macintosh Programmers

By Kris Newby, Apple Directions staff

Behind every great Macintosh software company is a great Macintosh programming team. These programmers are not only the chief architects of your products, they're often the "keepers of the vision" in your company. One or two of these visionaries can easily make the difference between a mediocre product and an award winner. The best programmers do much more than grinding out a daily quota of code. The great ones are renaissance men and women. They're self-taught human interface experts. They have an innate ability to turn new technologies into salable features. They're good at articulating a product vision to other team members. And they dream in five colors.

What's the best way to find these key programmers? Well, there's actually no single hiring strategy that works for everyone. But by augmenting traditional search tactics with some unique Macintosh developer community resources, you can significantly increase the chances of finding just the right programmers for your company. This article reviews a number of hiring strategies that have worked well for other successful Macintosh software companies.

A Snapshot of the Macintosh Programmer Market

To get a big-picture perspective on the Macintosh programmer market, I spoke with David Small, president of Scientific Placement, a U.S. search and recruiting firm that specializes in the commercial software industry.

"Right now the market for Mac programmers is hot," says Small. "There are a lot of companies hiring, and there's a scarcity of qualified candidates. What makes it even harder to fill these positions is the fragmentation in the programmer market."

According to Small, this "fragmentation" is due to cultural, salary, and specialization barriers that often prevent qualified programmers from moving from one position to another. Small feels that by having a good understanding of these barriers *before* you talk to candidates, you can increase the chances of attracting the best programmers to your company. Here are some points to consider when formulating your overall hiring strategy:

- **Geographical barriers.** The largest concentration of Macintosh programmers is in California's Santa Clara ("Silicon") Valley. There are other sizable programmer populations in Boston, Seattle, and Austin. Differences in climates, cultures, and housing costs often make programmer moves to and from these areas difficult.

Small sums up one of the most difficult barriers: "The high home prices in California are real deal-busters." (For example, "fixer-upper" three-bedroom homes in Palo Alto, one of the nicest areas of Silicon Valley, start at about \$500,000. This is far beyond the range of a typical software programmer.)

Before you start recruiting programmers out of your geographic region, make sure you're being realistic about your ability to move *and keep* programmers from different geographical regions. (CE Software, for example, found that it's hard to sell some Californians on the advantages of living in Iowa, so they've opted to recruit heavily from local colleges.)

If you decide to recruit out of your immediate area, be prepared to present information on relative housing costs and good neighborhoods at interview time. (Check out the Web sites <http://www.homefair.com> and <http://www.homes.com> to get information on comparative housing costs in different regions.) Better yet, take a candidate on a driving tour of nice

places to live. (If you're in Silicon Valley, show them the more affordable houses in Morgan Hill, Almaden Valley, or the Santa Cruz hills.)

Small says that Scientific Placement has had success relocating engineers to Silicon Valley by focusing on candidates who are willing to rent rather than buy a house. He says, "These individuals often come from a lower salary job market and their new employers are willing to give them a larger salary increase. Rental costs in Silicon Valley are proportionately lower than home ownership costs, and the higher salaries make it affordable for these applicants to move."

- **Salary barriers.** Another barrier to hiring out of your area is the salary curve difference between regions. A typical Macintosh programmer earning a \$65,000 salary in Austin, Texas, could command as much as \$85,000 in Silicon Valley or as little as \$50,000 in Ann Arbor, Michigan.

"This salary gap is particularly a problem for companies in small university towns, such as Champaign, Illinois; Ann Arbor, Michigan; and Raleigh-Durham, North Carolina," says Small. "Companies in these areas are able to hire eager 22-year-old computer science graduates at near slave-labor wages—at least for awhile. But once their company grows beyond ten employees, this advantage turns into a disadvantage. At this point, many of these companies deplete the local supply of computer science graduates, and they're faced with bringing in programmers at national-level wages. In a tight job market, many companies try to avoid disrupting their internal salary curves by hiring contractors. Paying contractors high hourly rates has in turn created a secondary short-term job market. There's usually some programmer crossover when otherwise career-minded employees are lured away by lucrative contractor rates. This simply creates more

openings on the permanent side."

To avoid this problem, Small advises small college town startup companies to think long term and plan for the day when they'll have to compete in the national job market. This means gradually moving engineering salaries up to more competitive levels.

Small also points out that prospective employees tend to be more impressed with salary dollars than with benefit plan dollars. He says, "If you offer someone a dollar's worth of employee benefits, it isn't likely to be valued as much as would a dollar's worth of additional salary."

- **Specialization barriers.** As the Mac OS grows in complexity, more programmers get locked into specialties, reducing the number of qualified candidates for your specific programmer needs. As an employer, you need to anticipate these shortages. For example, with the Mac OS 8 developer release shipping soon, you should be looking for device driver and OpenDoc programmers now. Another good way to cope with these shortages is to set up company-internal training programs. (See the "Growing Your Own Macintosh Programmers" section on page 30 for details.)

Maximizing Employee Referrals

By far, the preferred way to find great programmers is through recommendations of trusted friends and colleagues.

"We tell our best programmers to spread the word about open positions, because they recognize talent," says John Powers, president of guideWorks, LLC. "And by hiring programmers that our technical leads respect, we also

increase the probability that these new hires will fit in with the rest of the team."

Geoffrey Clements, director of TypeMaker, a software company based in the United Kingdom, adds, "The best way to find the good Macintosh programmers in Europe is through word of mouth, since the only publications that all European Macintosh developers read are in Apple's developer mailings."

Word-of-mouth hiring campaigns, however, aren't always fast, aren't foolproof (best friends aren't always the best programmers) and don't often work well in smaller towns. To turn up the volume on your word-of-mouth hiring campaign, make sure you e-mail new job listings to your internal programmers, so that the listings are easier to forward to colleagues. In addition, make it worth their while to find good candidates: Consider offering referral bonuses to employees and suppliers. Most developers find a \$500 or \$1,000 bonus a real bargain compared with a typical recruiter's fee of \$5,000 to \$10,000.

Leveraging the Internet

Many Macintosh developers have had tremendous success posting open job requisitions to the virtual programmer coffee pot—the Internet. The advantages to using Internet ads instead of newspaper ads are numerous: Ads can be posted immediately, they run longer, and they're visible worldwide. Web sites also offer you a less-expensive way to advertise. For example, you can post a 30-day worldwide job listing on the Career Mosaic Web site for just \$150, while a typical two-column ad in the *Boston Globe* or *San Jose Mercury News* can

cost between \$1,500 and \$2,500. (See the "Macintosh Programmer Recruitment Resources" box on page 31 for more information about resources mentioned in this article.) What's more, by having candidates e-mail you résumés, you don't have to wait for "snail mail" and human resource department processing time.

Allan Rousselle of Bare Bones Software (creators of BBEEdit) recently voiced a great truth about hiring Macintosh programmers over the Internet: "If you have a great product, programmers will come to you."

To facilitate this natural selection process, make sure your company's Web site includes a facility that makes it easy for programmers to submit a résumé. (And conversely, make sure there's someone on the receiving end to review these résumés.)

In Rousselle's previous position at Natural Intelligence (creators of the Roaster development environment for Java), he found programming specialists by distributing job notices through targeted Internet list servers. "I found some great Macintosh database specialists by sending a notice over a 4D database list server and a Cornell alumnae list server."

One of the best Web sites for posting technical positions is the Monster Board. Christopher Evans, the chief senior engineer at Natural Intelligence, regularly posts his programmer positions here, and he says it's one of the most popular job search sites in the Boston area. This site also includes international job listings, has a nice search engine, and includes hyperlinks to more than 40 other job search sites.

Two other good boards for posting programming positions are Career Mosaic, sponsored by Bernard Hodes Advertising, and the Career Expo site maintained by *High Technology Careers* magazine. (You can access both of these sites from the Monster Board.) Since Career Mosaic is run by a full-service ad agency in Silicon Valley, you can also get them to help you create newspaper ads, online job fairs, and custom job-listing Web sites. Career Expo is used by more than 500 employers, and similarly offers online job fairs. (These online job fairs cost about \$5,000, a real bargain compared to the \$20,000 to \$100,000 price tag for onsite fairs.)

Here's one last clever idea for finding talented programmers on the Internet. Howard Shere, president of Green Dragon Creations, often browses through shareware bulletin boards on the Internet, looking for programmers with talent.

Ten Ways to Recruit Great Macintosh Programmers

1. Offer employee referral bonuses.
2. Post a position on the "Monster Board" (see the "Macintosh Programmer Recruitment Resources" box on page 31).
3. Send a notice over a targeted list server.
4. Search shareware BBS's for talented programmers.
5. Add a "Send Résumé" button to your Web site.
6. Sponsor a college internship program.
7. Contact college career centers.
8. Hire a Macintosh-savvy search firm.
9. Place an ad in *MacTech Magazine*.
10. Take juggling lessons at the Worldwide Developers Conference.

Growing Your Own Macintosh Programmers

One of the challenges that “bleeding edge” software companies face is finding programmers with experience in their emerging technology area. Natural Intelligence, a company that creates Internet-related software, has found that the best way to cope with this situation is to “grow their own” programming specialists by running a college internship program. (For a list of where to post these internships, see the “Macintosh Programmer Recruitment Resources” box on page 31.)

“We provide new computer science graduates with an intensive six-month training program,” says Christopher Evans, Natural Intelligence’s chief senior engineer. “This way programmers are trained in the specific skills they need to succeed here, above and beyond the basics they received in college. We’ve also been able to create quite a diverse team of developers, given the growing percentage of women and minorities graduating from universities these days.”

Tom Cavanaugh, product development manager at CE Software, also grows his own: “We’ve started volunteering time and resources to the computer science departments of local universities, such as Drake, University of Iowa, and Iowa State, so we’ll have first access to their top graduates. We’ve also started a college internship program that enables student programmers to work side-by-side with an experienced programmer every Saturday.”

Using University Career Centers

Deborah Feo, vice president of Optimization Alternatives, has found another way to hire the best and brightest programmers from University of Texas at Austin—she makes personal calls to top students *before* the big software companies start their formal college recruiting visits. Because her 15-person software firm is low-profile and specialized—they develop custom software for the transportation industry—she has to work extra hard to sell programmers on her company. Her most successful hiring strategy to date has been to call up the University of Texas’s job placement center in January or February and have them e-mail her the résumés of all computer science seniors with grade-point averages above 3.5. She then sorts through the 40 or so résumés they send, calling the most promising candidates in for interviews.

“By hiring the top students, we know we’re getting smart, hardworking programmers. The

students are almost always flattered when we call them personally, and this gives us an edge in competing with larger firms.” Feo’s firm also pays programmer salaries well above the Austin average.

David Small elaborates on the pros and cons of college recruiting: “Most college

**One of the best Web sites
for posting technical
positions is the
Monster Board
(<http://www.monster.com>).**

placement offices do a nice job of rounding up new graduates, preparing their résumés, and setting up interviews for employers. The price is right, too—it’s free. College recruiting is seasonal and can only be done once or twice each year. Relocation costs are low, but interviewing costs may be higher, because of the large number of interviews that top students schedule. Training and employee turnover during the first two years are the significant cost elements of a college recruiting program. Each employer can only visit a few college campuses, and there are sure to be some great prospects missed.”

For those of you who choose to prospect for talented programmers within the computer science departments of colleges and universities, here are a few time-saving tips. According to Diane Warkentin in Apple’s college recruiting group, Apple has found a number of good Macintosh programmers at Stanford University, Dartmouth College, University of Michigan, and University of Texas at Austin. Another gem of a resource is the university database run by Christine DeMello at the Massachusetts Institute of Technology. This Web site (see the “Macintosh Programmer Recruitment Resources” box) includes hyperlinks to just about every major university Web site in the world.

Programmer Search Firms

Of course, the downside of hiring new programmers is that you have to train them, and quite often, these new graduates don’t stay long in their first jobs. If you’re more interested in quickly finding seasoned programmers—those with three or more years of Macintosh experience—using a search firm or independent recruiter may be your best bet.

Another reason to turn to a professional recruiting firm is that it will increase the probability that you’ll find the best possible hire in the least amount of time. Chances are you’re so busy doing your own thing—product development—that you have little time to browse shareware boards, cruise the Internet, or manage the time-consuming tasks of recruiting and screening candidates. Professional recruiters offer you access to a database of established contacts and their knowledge of search techniques. These service firms charge contingency fees to the employer, but they might turn up a better candidate with a lower salary or more attractive relocation situation. What’s more, these fees only have to be paid if you actually hire one of their referrals. Recruiting fees seem large *until* you compare them with the cost of hiring the wrong person, then spending six months looking for a replacement.

Two of the most visible search firms in the Macintosh programmer community are Scientific Placement and Trattner Network. Both of these firms advertise in major Macintosh publications and attend events such as the Macworld Expo and Apple’s Worldwide Developers Conference (WWDC).

Scientific Placement recruits Macintosh developers year-round, and they have five to six Macintosh recruitment specialists on staff. To date, they have a résumé database of more than 2,100 Macintosh programmers: 80 percent of these are looking for full-time positions and 20 percent are available for contract work. This nationwide firm has offices in Houston, northern California, and Austin, and they typically charge fees equal to 20 percent of a successful hire’s base annual salary.

Trattner Network is a Silicon Valley firm that specializes in contract programmers, though they do have some full-time Macintosh programmers in their database. They also have a Web site where you can list available positions.

Traditional Approaches: Print Ads and Events

Newspaper ads are still an effective means of recruiting if you want to attract programmers in a targeted geographical area. Douglas Castor, manager of reservation technology at Holiday Inn Worldwide, finds half of his new hires through local Salt Lake City newspaper ads

and half through recruiters. He says, "Newspaper ads often catch the eye of recruiters, who have contacts within software companies that have staff interested in moving."

If you're looking to staff up a large department, magazine ads can keep a steady stream of résumés flowing into your company.

"Magazine and trade publications offer a big readership advantage over newspapers," says Small. "Magazines, like newspapers, base their advertising rates on circulation, but the percentage of qualified readers may be several thousand percent higher with magazines. In fact, by phoning a magazine for a copy of their BPA International audit statement, you can get

Macintosh Programmer Recruitment Resources

WWW Job Bulletin Boards and Information

The Monster Board
<http://www.monster.com>

Bernard Hodes Advertising
 415-856-1000
 Career Mosaic job bulletin board
<http://www.careermosaic.com/>

High Technology Careers magazine site
<http://www.careerexpo.com>

College and university Web sites
<http://web.mit.edu/cdemello/www/P.html>

Regional real estate prices
<http://www.homefair.com> and <http://www.homes.com>

College Internship Bulletin Boards

Internships page
<http://icpac.indiana.edu/interns.html>

Brain Book's Internship Index
<http://www.reg.uci.edu:80/UCI/SOP/BRAINBOOK/internship.html>

Career Mosaic Internship listing
<http://www.careermosaic.com/cm/cc/cc1.html>

Macintosh-Savvy Recruitment Firms

Scientific Placement
 800-231-5920
 das@scientific.com
<http://www.scientific.com/>

Trattner Resources (primarily focuses on consultants)
 Emily Hoolhorst 415-949-9555 (emily@tratnet.com)
<http://www.tratnet.com>

Print Ads/Mailings

MacTech Magazine
 Print ads 805-494-9797
 adsales@xplain.com
<http://www.xplain.com>

MacSciTech Journal
 Print ads: 508-755-5242
<http://www.macscitech.org/>

Newspaper Web sites
<http://www.newslink.org/>

User Group Connection
 Mailings to programming special interest groups
 408-477-4277 ext. 202
sdecker@ugconnection.org
<http://www.ugconnection.com/>

Standard Rate and Data Service (SRDS)
 This report provides you with detailed information on magazine readership data, advertising costs, circulation numbers, and geographic distribution.
<http://www.srds.com>

Macintosh Programmer Events

Apple Worldwide Developers Conference (beginning of May 1997)
<http://www.apple.com>

MacHack (end of June)
 313-882-1824
<http://www.machack.com>

1997 Computer Game Developers' Conference
 April 26–29, Santa Clara Convention Center, Santa Clara, California
 To reserve exhibit or job fair space, contact Gina Bovero at 415-356-3406.

Macworld Boston (August 7, 1996) and San Francisco (January 1997):
 You can find Macintosh developers hanging around Developer Central, MacSciTech Resource Center, and the UGWUMP (user group welcoming, unwinding, meeting place).
<http://www.mha.com/macworldexpo/highlights.html>

PC Expo New York (June 18–20) Career Fair
 Contact James Henry at 800-829-3976 for information on renting booth space at this career fair and others.
<http://www.shownet.com>

a very accurate breakdown of readership industry breakdowns, job functions, and so on. In addition, you can subscribe to the Standard Rate and Data Service (SRDS). This report provides you with detailed information on various magazines, including readership data, advertising costs, circulation numbers, and geographic distribution. With magazines you also benefit from pass-along readers and impulse buyer responses, resulting from someone thumbing through a magazine."

Though there are a lot of great programming magazines around, like *Dr. Dobbs Journal* and *Software Development*, you'll find the most focused Macintosh programmer audience in *MacTech Magazine*. Advertising rates in this publication range from \$185 for a two-inch column ad to \$1950 for a full-page color ad. If you're looking for Macintosh programmers who have experience in a scientific specialty, try advertising in the *MacSciTech Journal*. This journal also posts job listings on its Web site.

To further leverage the cost of that ad, you could also send a copy to Macintosh user groups across North America. The User Group Connection, based in Soquel, California, typically can mail flyers to these influential groups for less than the cost of postage.

Recruiting at Macintosh-related events can also be fruitful. For informal recruiting, you'll find the most Macintosh programmers at MacHack, held at the end of June and Apple's WWDC, typically held the second week of May. Though organizers of these events discourage overt hiring campaigns, if you need to hire a large number of programmers, it may be worth your while to sponsor an "open house" in a nearby hotel suite.

If you're a game or interactive entertainment developer looking for talented Macintosh programmers, consider attending the 1997 Computer Game Developers' Conference, which will be held from April 26–29 in Santa Clara, California. Of the 5,700 participants in 1996, 47 percent of the attendees develop for the Macintosh. You can also rent a booth at the conference job fair. Last year, for instance, Virgin Interactive needed to quickly build an interactive media division, so they orchestrated a visible recruitment campaign that included a large booth at the job fair and free T-shirts that said, "I'm Looking for Virgin Programmers."

And finally, if you're attending PC Expo in New York City this year, you may want to check out Blenham Group's new career fair,

which is expected to attract a large audience of programmers from the trade show's estimated 130,000 attendees.

Candidate Evaluation Tips

Once you've gotten the word out about your open programmer positions, chances are you'll end up with a tall stack of résumés to evaluate. A few of the developers with whom I spoke offered tips on selecting the best of the bunch.

After an initial round of telephone interviews and before an onsite interview is scheduled, Optimization Alternatives requests code samples from a prospective hire. They feel this is the best way to "get inside the head" of a prospective candidate. "Even a quick glance at a programmer's code tell you whether a programmer is organized and methodical," says Deborah Feo.

Because teamwork is so critical in large programming projects, most companies use a team interview approach, allowing multiple programming team members to evaluate a candidate. Douglas Castor from Holiday Inn's reservations group discussed why he feels this step is essential: "Everyone gets a different cut on the candidate. I get buy-in from the group, help in evaluating the technical level of the candidate, and a test of how well the candidate will mix with the group. The candidate also gets a realistic view of the dynamics within our group."

Optimization Alternatives also stressed the importance of references. (They like to get three or four before they hire anyone.) With the fear of lawsuits, however, getting references can be difficult. To work around this, Feo always asks reluctant references for names of people outside the company who've worked with the candidate (contractors, college classmates, and so on). Most people are more than willing to give you other "unlisted" references, and these contacts often give you a more rounded view of a prospect than the candidate's preferred list.

The Final Step—Selling the Candidate

Finding a great Macintosh programmer is, of course, only half the battle. You then have to convince that person to join your company. Over the years, Scientific Placement has acquired some sobering hiring statistics: On average, the cost of a single hire is equivalent to the time-cost required to process 160 résumés. Once you find the right candidate, a

great way to reduce this expense is to increase your offer acceptance rate. In the course of discussions on locating programmers, several developers mentioned ideas, other than a high salary, for attracting the best programmers to your company.

First, sell your work environment. Green Dragon Creations, based in Water Valley, Mississippi, competes with larger companies by emphasizing their flexible hours, fun people, lack of big-company politics, attractive cost of living, and low crime rate. Optimization Alternatives offers each programmer \$1,000 in "fun money" to spend on enhancing their workspace. One of Feo's programmers used it to buy a fancy joystick for her firm's after-hours Marathon II sessions (the shoot'em-up game by Bungie Software), and another used it to buy a high-quality audio CD player and headphone set.

Finally, I think Gene Wang, author of *The Programmer's Job Handbook* (ISBN 0-07-882137), best summed up how to lure the best and brightest to your company: "With developers, it's not just money that motivates—it's the chance for greatness. When interviewing programmers, I paint a picture of Winning All Reviews (WAR) and setting new standards in our software category." ♣

Kris Newby (newbyk@applelink.apple.com) is the business editor of Apple Directions and a freelance writer based in Palo Alto, California.

[Author's Note: Many thanks to this article's contributors: David Small (das@scientific.com), John Powers (powers@guideworks.com), Geoffrey Clements (gclements@typemake.demon.co.uk), Allan Rousselle (allan@barebones.com), Tom Cavanaugh (tom_cavanaugh@cesoft.com), Christopher Evans (evans@natural.com), Howard Shere (hshere@greendragon.com), Deborah Feo (oasis@io.com), Douglas Castor (networkz@aol.com), Sam Decker (sdecker@ugconnection.org), and Gene Wang (genewang1@aol.com). And just for the record, the resources mentioned in this article reflect the opinions of the author and don't constitute a recommendation by Apple Computer, Inc. If I've overlooked any valuable recruitment resources, please e-mail them to me or share them with other Macintosh developers on the semper.fi list server.]

Listings

IT SHIPPED!

The following 140 products were added to the It Shipped! database between April 15 and May 15, 1996, bringing the total number of products entered into the database to more than 480 since Apple reinstated the program in late 1995. Products developed first for the Mac OS platform are designated with the words "MAC FIRST" on the far right; products available only for Mac OS systems receive the designation "MAC ONLY."

The It Shipped! database is used by Apple employees when they prepare advertising, collateral, and white papers and when they help customers find computing solutions; it's also broadcast to key industry publications.

For more information about the It Shipped! program, visit the It Shipped! Web Page (<http://dev.info.apple.com/itshipped.html>).

To enter your Macintosh product in the database, you can obtain a form from the site (<http://dev.info.apple.com/thirdparty/submission.html>).

You must also send a copy of the product to Apple at this address:

Apple Computer, 1 Infinite Loop, M/S: 301-1ES, Cupertino, CA 95014, USA.

Company	Product/Version	Mac Only/Mac First	Company	Product/Version	Mac Only/Mac First
6S Software	CreditMedic 1.5b		Digital Collections Inc.	EmbARK 1.5	MAC FIRST
Acquired Knowledge, Inc.	Download Mechanic 1.7.2		Digital Collections Inc.	ImageAxs 1.5.2	MAC FIRST
Adobe Systems, Inc.	Adobe After Effects 3.1	MAC FIRST	Eastgate Systems, Inc.	Patchwork Girl by Shelley Jackson 1st Edition	MAC FIRST
Adobe Systems, Inc.	Adobe Premiere.4.2	MAC FIRST	Farallon Computing, Inc.	AirDock	MAC FIRST
Ambrosia Software, Inc.	Apeiron 1.0.2		Farallon Computing, Inc.	Ether10-T StarController	
Ambrosia Software, Inc.	Barrack 1.0.0		Farallon Computing, Inc.	Ether10-T Starlet	MAC FIRST
Ambrosia Software, Inc.	Chiral 1.0.3		Farallon Computing, Inc.	EtherMac Transceivers, Cards, and Adapters	MAC FIRST
Ambrosia Software, Inc.	Maelstrom 1.4.3		Farallon Computing, Inc.	EtherWave Products	MAC FIRST
Ambrosia Software, Inc.	Swoop 1.0.2		Farallon Computing, Inc.	Fast EtherTX-10/100 Cards	MAC FIRST
ATTO Technology, Inc.	ExpressPCI Multi-Channel UltraSCSI (MC-U)	MAC FIRST	Farallon Computing, Inc.	Fast Starlet 100TX/8 hub and 10/100 bridge	
ATTO Technology, Inc.	ExpressPCI Single-Channel UltraSCSI (SC-U)	MAC FIRST	Farallon Computing, Inc.	InterRoute/5	
Big Software, Inc.	Big Business 2		Farallon Computing, Inc.	LocalPath 1.0.3	
Brossco Systems, Inc.	Voyant 3	MAC FIRST	Farallon Computing, Inc.	PhoneNET Connector	MAC FIRST
Casady & Greene, Inc.	Let's Keep It Simple Spreadsheet 1		Farallon Computing, Inc.	PhoneNET StarController	
CFM Inc.	TeamFlow 3.2	MAC FIRST	Farallon Computing, Inc.	PowerPath 1.0.3	
Charles River Media	The Macintosh 3D Handbook 1st Edition	MAC FIRST	Farallon Computing, Inc.	StarRouter	
Cherwell Scientific Publishing	pro Fit 5		Gary Dauphin	Mr. Hex in Louisiana by Bill Michael, version 2	
Connectix Corporation	Color QuickCam for Macintosh 1.0/hardware	MAC FIRST	Geode Software Company	Discovering Origami, version 1	
Connectix Corporation	Color QuickCam for Macintosh 2.0/software	MAC FIRST	Gig O' Byte Music	Song Servant, version 1	
Data Translation	Media 100 2.6		Giles Road Press	A BBS Caller's Guide to FirstClass, 1st Edition	MAC FIRST
Data Translation	Media 100 qx 1		Graham Herrick	Finder Windows	
Datu, Inc.	CharmAnalysis 1.08		Hi Resolution, Inc.	WATCHIT!	
Design Tools Monthly	Design Tools Monthly Vol. 4, No. 5		Inspiration Software, Inc.	Inspiration 4.1b	MAC FIRST
Diamante Software, Inc.	HomeStuff* Personal Inventory Manager 2.0.5		Intuit, Inc.	QuickBooks 4	
Diamante Software, Inc.	PowerRep* Order and Invoice Management System 1.5.1	MAC FIRST	Intuit, Inc.	QuickBooks Pro 4.0 for Macintosh	
			Ionex Consulting Services	IonExpert 1	
			Key Curriculum Press	NuCalc 1.0	
			Key Curriculum Press	The Geometer's Sketchpad 3.0	MAC FIRST
			KEYSPAN	SMART SERIAL 6, 1.2	MAC FIRST

Learning in Motion	Book Exchange 1	MAC FIRST	Philips Media OptImage	MediaMogul 2.2.1	MAC FIRST
Learning in Motion	Measurement in Motion 1	MAC FIRST	Philips Media OptImage	Video CD Toolkit 1.1	MAC FIRST
Learning in Motion	Researcher 1	MAC FIRST	Pierian Spring Software	CampOS Math1	
Logicsu Incorporated	Perfect Copy Techwrite 1.18		Pierian Spring Software	Continent Explorer II 1	
Logicus Incorporate	JOBS for ME 1.2.3		Pierian Spring Software	Digital Chisel 2.1	
Logicus Incorporated	Perfect Copy Biology		Pierian Spring Software	Digital Chisel://HTML 2.1	
Logicus Incorporated	Perfect Copy Classic 1.3		Pierian Spring Software	Interactive Geography 1	MAC FIRST
Looking Glass Technologies, Inc.	Flight Unlimited		Power On Software, Inc.	Net Commander 1	
Mark H. Delfs	Macintosh Startup Screens		Power On Software, Inc.	On Guard 2.1	
Mark of the Unicorn, Inc.	Composer's Mosaic 1.43		Ray Dream, Inc.	AddDepth 2	MAC FIRST
Mark of the Unicorn, Inc.	Digital Performer 1.7		Ray Dream, Inc.	Ray Dream Studio 4	MAC FIRST
Mark of the Unicorn, Inc.	FastLane	MAC FIRST	Ray Dream, Inc.	Ray Dream Studio 1.0	MAC FIRST
Mark of the Unicorn, Inc.	FreeStyle 1.04	MAC FIRST	Reply Corporation	DOS on Mac Card	MAC ONLY/MAC FIRST
Mark of the Unicorn, Inc.	MIDI Express	MAC FIRST	Reprographic Products Group, Inc.	Dai-Nippon Screen Scanner Plug-in for Photoshop 1	MAC FIRST
Mark of the Unicorn, Inc.	MIDI Time Piece II	MAC FIRST	Reprographic Products Group, Inc.	Isomet Drum Scanner Acquire Plug-in for Photoshop 2	MAC FIRST
Mark of the Unicorn, Inc.	Performer 5.5		RES Software, LLC	World of Words 2.0	
Mark of the Unicorn, Inc.	Pocket Express for both Macintosh and Windows		Sierra Solutions	MailRoom, Macintosh Edition MCIMail) 1.1	
Mark of the Unicorn, Inc.	Unisyn 1.14	MAC FIRST	SKY4 STUDIOS	AMT Accelerator 2	
Micro Consulting SA	BiblioMac 2.6	MAC FIRST	Softways	Mr. File 2	
Micro Consulting SA	Office Maker 1.4	MAC FIRST	Sophisticated Circuits, Inc.	PowerKey Pro/Model 200	
Micro Consulting SA	Office Maker Adresses 1.4	MAC FIRST	Sophisticated Circuits, Inc.	PowerKey Remote 2.0.3	
Micro Consulting SA	Office Maker Documentation 1.4		Soul Fry Software	Classic Gin Rummy 1	
Micro Consulting SA	Office Maker Finance 1.4	MAC FIRST	Specular International	Specular Collage 2.0.2	
Micro Consulting SA	Office Maker Management 1.4	MAC FIRST	Specular International	Specular LogoMotion 2.0.1	
Micro Consulting SA	Office Maker Salaries 1.4	MAC FIRST	Specular International	Specular TextureScape 2.0.1	
MICRO WORLD	Close-up lens set for QuickTake 100/150 camera 2.0		SPSS Inc.	SPSS 6.1 for the Macintosh	MAC ONLY
Microsoft Corp.	Microsoft Bookshelf 1996-97 Edition		TechPool Studios, Inc.	LifeART Collections	MAC FIRST
Microspot USA, Inc.	3D World 1		The Cobb Group	THE MACAUTHORITY	MAC FIRST
Microspot USA, Inc.	PhotoFix 3.1		Thomson Learning TOOLS	PAWS in Typing Town 1.0	
Miles Information Systems	Miles Apart	MAC ONLY/MAC FIRST	Thomson Learning TOOLS	The Magic Applehouse 1.0	
MindGate Technologies	LaserMerge Electronic Paper 2		TLA Systems	DragThing 1.6	MAC FIRST
MindVision Software	Installer VISE 3.5		TopSoft, Inc.	FilterTop 1	
MindVision Software	Updater VISE 1.0.2		Total Integration	FASTedit/Deluxe 1	MAC ONLY
Miracle Software Inc.	HTML Web Weaver Lite 3.0		Water's Edge Software	SuperCDEFs 1	MAC FIRST
Miracle Software Inc.	World Wide Web Weaver for Mac .1.1		Wave Research, Inc.	FileWave 2.0 2	
Multi-Ad Services, Inc.	Multi-Ad Creator 4		WaveMetrics, Inc.	IGOR Pro 3	
MUSI Computer Products	PhoneFloater 1.1		Wood River Gallery	Pictures NOW! Animals 1	
Newt Boot	Newt Boot		Wood River Gallery	Pictures NOW! Christmas Cheer 1	
Nisseb softwares	HTML Vocabulary 1.1		Wood River Gallery	Pictures NOW! Funny Bone 1	
Nisseb softwares	Mailto Converter 1		Wood River Gallery	Pictures NOW! Holidays & Occasions 1	
North Plains Systems Inc.	TeleScope 1	MAC FIRST	Wood River Gallery	Pictures NOW! Sports & Recreation 1	
North Plains Systems Inc.	VRML Equinox 1	MAC FIRST	Wood River Gallery	Pictures NOW! Transportation 1	
Philips Media OptImage	Audio2MPEG 1				
Philips Media OptImage	Balboa 2.1				

DEVELOPER UNIVERSITY SCHEDULE

Developer University (DU) offers a broad range of Mac OS and Newton programming instruction through hands-on classes and self-paced training

Advanced C++**5 Days/\$1000**

Classroom

July 22–26, Cupertino

Apple Events/AppleScript Programming**5 days/\$1,000**

Classroom, Self Paced

August 12–16, Cupertino

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

Classroom

June 24–28, Cupertino

July 15–19, Cupertino

August 12–16, Cupertino

September 16–20, Cupertino

Intermediate Programming: 7.5 Topics

Self Paced

Introduction to PowerPC

Online

Introduction to PowerTalk

Online

Introduction to RISC Technology

Online

Macintosh Debugging Strategies & Techniques**3 days/\$900**

Classroom

July 8–10, Cupertino

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

Classroom

June 18–20, Cupertino

July 16–18, Cupertino

August 13–15, Cupertino

September 17–19, Cupertino

Newton Programming: Essentials 2.0**5 days/\$1,500**

Classroom

June 17–21, Cupertino

July 22–26, Cupertino

September 9–13, Cupertino

products. Classes are offered in Cupertino, California and through selected third-party trainers.

Programmer's Introduction to PowerPC

Online

Programmer's Introduction to RISC and PowerPC

Self Paced, Online

Programming with MacApp

On demand—call DU Registrar for more information

Programming with QuickDraw GX**4 days/\$1200**

On demand—call DU Registrar for more information

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

Classroom

July 22–24, Cupertino

September 9–11, Cupertino

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

Classroom

July 8–12, Cupertino

August 19–23, Cupertino

September 30–October 4, Cupertino

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

Classroom

July 22–23, Cupertino

September 9–10, Cupertino

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


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

AppleLink—Developer Support:Developer Services:Apple Developer Services:Developer Information:Developer University

Internet—<http://dev.info.apple.com/du.html>

America Online—Computing:Computing Forums:Development:Mac Development Q&A:Developer University.

APPLE 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. It includes Internet resources from Apple Computer, as well as from other companies and people. Apple sites are designated by an **A**; we can't guarantee the information in the non-Apple resources, but we think you'll find them useful. The list is alphabetized according to the name of each Web site or other type of resource; new resources, or listings that have changed this month are indicated with this mark: 

You'll find this feature particularly helpful when you view it at the *Apple Directions* Web page (located at <http://dev.info.apple.com/appledirections/adtoc.html>). There, all the names of the locations listed in this article are linked to the sites themselves; clicking the names will take you directly to the relevant Internet locations. We'll update this feature every month, based both on what Apple is doing on the Internet and on your feedback.

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

Ambrosia Cafe

<http://www.ambrosiasw.com/cafe.html>

One of several "everything you want to know about the Macintosh" pages.

**Apple CEO Gilbert F. Amelio****A**

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

Gil Amelio's plans and ideas for Apple and the entire Apple community.

Apple Competitive Information**A**

<http://support.info.apple.com/competitive/competitive.html>

Information comparing Mac OS computers favorably with (mostly) Windows 3.1-based machines.

**Apple Computer****A**

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

This entry isn't new, but the Apple Computer Web page is. Check it out.

Apple Developer Catalog**A**

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

The Apple Developer Catalog is now online; check it out! It's a great way to view and order development tools, technical resources, and training products and information if you're developing applications and solutions for Apple platforms.

Apple Developer Services and Products**A**

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

The main page for Apple Computer's developer services.

Apple Developers Listing

<http://www.amsys.co.uk/applelinks.html>

Links to the Web pages of hundreds of Apple platform developers.

Apple Directions Express List Server**A**

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

Apple Directions Express is our biweekly e-mail digest of business news and information from Apple, sent to you over the Internet and posted at this Web site. It includes pointers—live links at our Web site—to other sources for more detailed information. Subscribe by sending e-mail to adirections@thing1.info.apple.com. In the subject field, type the string "subscribe <your real name>".

Apple Education**A**

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

Use online forms located at this site to request product specifications, information about the Apple Education Series (bundled products), and technical support from Apple engineers.

Apple Europe**A**

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

The front door for information about Apple activities—including developer services—in Europe.

Apple Forever**A**

<http://www2.apple.com/appleforever/>

Regular updates about the company and special communications from Apple executives.

Apple FTP Sites**A**

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

ftp://ftp.info.euro.apple.com/Apple.Support.Area/Developer_Services

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

Apple International Developer Services and Products**A**

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

Contains the current list of international Apple Developer Services locations and contacts.

Apple Internet Servers**A**

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

Includes lists of other Apple Web sites as well as Gopher and FTP sites.

Apple List Servers**A**

The following are several pertinent Apple Internet mailing lists, and the addresses you can send messages to if you'd like to subscribe. For each one, type "subscribe <your real name>" in the body of the message you send.

Apple Information Alley

A

infoalley@lists1.info.apple.com

All Press Releases

A

pressrel@thing2.info.apple.com

Software Updates

A

swupdates@thing1.info.apple.com

New Hardware

A

newhdw@thing2.info.apple.com

Newton Press Releases

A

newtonpr@thing1.info.apple.com

What's New on Apple Developer Web Pages

A

devnew@thing1.info.apple.com

Apple Media Program A<http://www.amp.apple.com>

Includes information about Apple's multimedia technologies as well as a searchable database of multimedia developers.

Apple Pacific A<http://www.info.apple.com/pacific/>

Contains information about Apple offices and developer support in the Pacific region, including Japan, Australia, and Latin America.

Apple Software Licensing A<http://dev.info.apple.com/swl/swl.html>

Official information on whether you need a license from Apple and how to obtain one.

Apple Solution Professionals Network (ASPN) A<http://support.info.apple.com/aboutapple/aspn.html>

Download the latest directory of consultants who specialize in Macintosh solutions.

Apple Strategy A<http://ThePlan.apple.com>

A guide to Apple's business strategies.

AppleTalk Network System Support<http://www.netfrontiers.com>

Information about planning, building, and managing networks based on AppleTalk; includes the Apple Certified Server Engineer (ACSE) support page.

Apple Tech Info Library A<http://til.info.apple.com/til/til.html>

Apple's official technical support database—updated daily—with over 12,000 articles on all aspects of Apple products, past and present.

"Ask Apple" Tech Support FAQs A<http://support.info.apple.com/askapple.faqs/askapplehome.html>

Frequently asked questions about Apple systems, and their answers.

Brad's WebSTAR/MacHTTP<http://www.ape.com/webstar/>

A database of all the Macintosh computer-based Web sites that the owner of this site can find—so far nearly 1,000 entries strong.

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

Provides a great deal of OpenDoc content.

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

A listing of all Macintosh computer software conflicts and cures known to the site's owners.

ComputerWare's Macintosh Linkshttp://www.macsources.com/links_all.html

Another compendium of information about the Macintosh computer; includes links to over 800 Macintosh manufacturers and publishers as well as links to more than 200 other Macintosh-related sites, including news-groups, user groups, and file libraries.

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

Another "everything Macintosh" compendium of information for Macintosh lovers.

Cyberdog A<http://cyberdog.apple.com>

The home page for Apple's OpenDoc-based, integrated Internet-access software suite; Cyberdog 1.0 is available for free downloading from the site.

DayStar Digital<http://www.daystar.com/DayStarHome2.html>

Contains information about DayStar's PowerPC upgrade cards and their newly released Mac OS-compatible Genesis MP media-publishing workstation.

Developer Depot Catalog<http://www.devdepot.com>

A catalog of products for Apple platform developers, brought to you by Xplain Corporation, publishers of *MacTech Magazine*

Development Tools A<http://devtools.apple.com/>

Listings of Apple and third-party development tools and a variety of technical documentation and white papers.

Digitool (Macintosh Common Lisp)<http://www.digitool.com/>

Contains information on the Macintosh Common Lisp (MCL) product line.

Electronic Publishing Risks<http://www.poulton.com/eo-why.htm#why>

Free information about the legal and financial risks of online communication, provided by Poulton Associates, which provides insurance and risk management services for U.S. clients.

Firewire (IEEE 1394)<http://www.firewire.org>

The site sponsored by the 1394 Trade Association to foster development and acceptance of the Firewire bus technology.

Gradient—DCE for the Macintosh<http://www.gradient.com/>

Contains information about Gradient's Mac-DCE product, an implementation of OSF DCE Secure Core functionality for Macintosh clients.

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

The Apple Guide home-away-from-home page.

Guy Kawasaki's EvangeList List Server A

All the good news about Apple platforms that Guy can find. For information on how to join, send an e-mail message to macway-request@solutions.apple.com for an automatic reply. (Any message will work.)

Guy Kawasaki's Semper.fi List Server A

A two-way list server that encourages communication between developers and Apple. We suggest you subscribe to the digest version; to do so, send a message to listproc@solutions.apple.com. In the body of the message, type "set semper.fi mail digest".

Happy Puppy's Macintosh Games Page<http://happypuppy.com/games/mac/>

Go to this site to find something to do with all that free time you have (or to divert you from the work you're trying to do).

Hartsook Letter<http://www.hartsook.com>

Excerpts from *The Hartsook Letter*, written by long-time Macintosh market analyst Pieter Hartsook.

 **IBM Microelectronics**<http://www.chips.ibm.com/>


The home page of the IBM division responsible for the company's Mac OS-compatible computer business.

It Shipped!<http://dev.info.apple.com/itshipped.html>

The home page for the It Shipped! program.

Key Apple Developer Relations Contacts<http://dev.info.apple.com/adrcontacts.html>

Intended mostly for use when standard Apple feedback mechanisms aren't working.

 **Linux for Macintosh (MkLinux)**<http://www.mklinux.apple.com>

Go here for information about Linux for Macintosh computers, or MkLinux, being jointly developed by Apple and Open Software Foundation. MkLinux DR 1 is available at the site.

Mac*Chat Newsletter<http://www.cts.com/browse/xxltony>

An online newsletter directed primarily at Macintosh customers to help them make the best use of their Macintosh systems. To subscribe, send e-mail with the string "SUBSCRIBE MACCHAT" in the body of the message to listserv@vm.temple.edu.

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

Find out about the annual MacHack hackers' conference.

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

Dedicated to showing why Mac OS-based systems are better than PCs running Windows 95. Go to this site for details about Apple's contest for the top Macintosh-hosted Web sites.

Macintosh Application Environment<http://www.mae.apple.com>

Contains a sample of the Macintosh Application Environment (MAE), software that lets UNIX workstations run Macintosh applications.

Macintosh Help Wanted<http://www.memphisweb.com/mathew/default.html><http://www.memphisweb.com/nammac/default.html>

Need to find programmers and others to work on developing Macintosh products? Go to these locations for help.

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

A non-Apple site intended primarily for users, with news, shareware, online discussions, and a particularly useful history of Macintosh computers all the way back to the original Macintosh 128K system.

Macintosh PowerBook and Mobile Computing<http://www.info.apple.com/gomobile/>

Complete information about PowerBook computers.

Macintosh Prices, France and United Stateshttp://www.ie2.u-psud.fr/~peirano/mac_us.html

Contains lists of French and U.S. Macintosh prices, as well as a brief history of the Macintosh computer, information on potential future Macintosh and Mac OS-compatible systems, and performance data about most models.

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

A terrific source for Apple and non-Apple Macintosh programming tools.

Macintosh Speech Recognitionhttp://www.vannevar.com/Mac_SR

As described by its owner, "a showcase for Mac speech recognition tips, tricks, and software."

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

A directory of companies with products for the Macintosh computer.

Mac OS<http://www.info.apple.com/macos/>

Go here for the latest information on the Mac OS.

Mac OS 8<http://www.macos.apple.com/macos8>

Apple Computer's official source for technical and business information about Mac OS 8, the next major release of the Macintosh operating system.

Mac QC Links<http://www.seapine.com/qclinks.html>

Information about Macintosh quality-control software and services with links to sites where you can find tools for testing and debugging Macintosh software. To submit your product/service for listing or to request a link to your page, send e-mail to macqc@seapine.com.

MacTech Magazine<http://www.mactech.com>

Contains *MacTech Magazine's* list of Internet locations for Mac OS developers.

Mac vs. UNIX Web Server Performance<http://www.netdreams.com/net.dreams/papers/theTest.html>


Contains a server performance comparison between a Power Macintosh computer and a Sun SPARC workstation, both being used as Web servers. Guess which system wins?

MacSciTech<http://www.macscitech.org/>

The home page of MacSciTech, the association for scientific/engineering/technical Macintosh users.

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

Find out about Metrowerks' CodeWarrior PowerPC development environment.

 **Motorola Computer Group**<http://www.mot.com/computer>

News and product and technology information from the division of Motorola responsible for manufacturing the company's Mac OS-based systems.

Nathan's Everything Macintosh Page<http://www.cs.brandeis.edu/~xray/mac.html>

A virtual treasure trove of Macintosh information, as its name implies.

Natural Intelligence<http://www.natural.com/>

Information about Natural Intelligence's tools and solutions, including Roaster and Roaster Professional.

**Newton**

A

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

Includes information about Newton 2.0 and Newton Toolkit 1.6 as well as the Newton Internet Enabler Beta version.

Nisus Software<http://www.nisus-soft.com/-nisus/>

Click on the Trash icon at this site and see where you end up!

**OpenDoc**

A

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

The place to go for the OpenDoc information, development kits, and software. The latest version—OpenDoc DR 5—has just been made available at the site.

**OpenDoc List Server**

A

listproc@cil.org

Send a message to the above address for OpenDoc news; include the phrase "subscribe OpenDoc-Interest" on the first line (not the subject field) of your message.

OpenDoc Part Ideashttp://www.eng.uci.edu/~sroussey/NetVision/software/od_parts/

A repository of OpenDoc software, as well as ideas for software that could be implemented as OpenDoc components.

**Open Transport**

A

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

Apple Developer Technical Support's site for information about Open Transport, Apple's new communications architecture.

Part Merchant Home Page<http://www.partmerchant.com/>

An OpenDoc part storefront on the Internet, sponsored by Kantara Development, where you can post your new OpenDoc parts and make them available for downloading by customers.

**PC Fairy Tales**<http://www.madmansdream.com/FairyTales/newpcTales2.toc.html>

The new locations for information to help debunk common Macintosh/PC myths.

Pictorius<http://www.pictorius.com>

Contains information about Pictorius Prograph CPX and Peregrine, its visual application client/server database programming environment.

Pippin

A

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

Contains technical information about designing products that will work with Apple's PowerPC processor-based, low-cost CD playback device.

Polymorphic E-zine<http://www.webcom.com/icog/polymorphic/index.html>

A new online magazine about Macintosh programming, focusing in particular on Pascal and C/C++.

Power Macintosh

A

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

Two useful sites for information about Power Macintosh computers.

QuickDraw GX

A

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

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

QuickDraw GX Fan Club<http://www.ixmedia.com/quickgx/quickgx.html><http://www.ixmedia.com/quickgx/subscribe.html>

The first site includes information designed to encourage the use and development of QuickDraw GX; the second provides addresses of two e-mail lists for receiving updates about QuickDraw GX.

QuickDraw 3D

A

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

Everything you need to know about QuickDraw 3D.

QuickTime

A

<http://quicktime.apple.com>

News and technical and marketing information about QuickTime.

QuickTime Live!

A

<http://live.apple.com>

Apple's site for showing multimedia broadcasts of live entertainment, including images, videos, sound, and QuickTime VR.

QuickTime VR

A

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

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

Quinn's Human Interface Subtleties<http://redback.cs.uwa.edu.au/Quinn/WWW/HumanInterfaceSubtleties.html>

Lists the many human interface subtleties that continue to make the Macintosh user experience richer and easier.

Shareware.com<http://www.shareware.com>

As its name implies, a site devoted to providing shareware.

Software Unboxed<http://www.unboxed.com>

Broadcast Software's online distribution site for commercial software; you can post a special "locked" version of your software, using technology provided for free by Broadcast Software. Customers then contact you to purchase a "key" to "unlock" and download the software.

Speech Technologies

A

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

Download the new Speech Recognition Manager SDK free of charge.

Technotes

A

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

Contains all Technotes—new and old—as well as author's guidelines for contributing your own technical notes.

Third-Party Products

A

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

Fill out the form located at this site to add your products to this list.

Ultimate Macintosh Page

http://www.freepress.com/myee/ultimate_mac.html

Contains more Mac OS information and software than you could possibly imagine exists.

User Group Connection

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

Resources and services for marketing your products to Apple's most influential and enthusiastic users: user groups.

Web Promotion Services

<http://www.computer.net/~owlseye/>

An inexpensive service for promoting your Web site to a broad variety of Web indexes and search engines.

Yahoo

<http://www.yahoo.com>

The well-known Internet directory and search engine. Yahoo is trying to strengthen its list of Macintosh-related Web pages; if you run one, or know of one, go to this site and search to see if Yahoo already includes the page you'd like to list. If not, click the Yahoo category to which your page belongs; then click the "add URL" icon at the top of the screen, and follow the instructions that appear.

Internet Resources for This Issue

News

- QuickTime VR BMW site—<http://www.bmwusa.com/>
- "Copland Drivers—Time to Put the Pedal to the Metal" article—<http://dev.info.apple.com/appledirections/dec95/copland.html>
- Errata to December 1995 Copland drivers article—<http://dev.info.apple.com/appledirections/jan96/errata.html>
- "Looking Forward to the Copland User Experience" article—<http://dev.info.apple.com/appledirections/apr96/stratmos.html>
- OpenDoc Web page—<http://www.opendoc.apple.com/>
- Cyberdog Web page—<http://cyberdog.apple.com/>
- "Putting OpenDoc to Work in 1996" article—<http://dev.info.apple.com/appledirections/feb96/opendoc.html>
- Open Transport article—<http://dev.info.apple.com/appledirections/sep95/unifiednetworking.html>
- Dr. Amelio's WWDC address—<http://wwdc.apple.com/webcast/archives/raarch.html>
- Apple's strategy and plans—<http://ThePlan.apple.com> and <http://www.apple.com/leadership/>
- Macintosh ease-of-use study—<http://www.apple.com/whymac>
- WWDC coverage—<http://www.dev.info.apple.com/appledirections/adtoc/html>
- Mac OS 8 Web site—<http://www.macos.apple.com/macos8/>
- List of OpenDoc developers—<http://www.opendoc.apple.com/press/committed/html>

Technology

- Game Sprockets Web site—<http://dev.info.apple.com/games>
- Inside Mac Games Web site—<http://www.imgmagazine.com/>
- Apple's Speech Web site—<http://www.speech.apple.com>

Business

- The Monster board—<http://www.monster.com>
- Career Mosaic job bulletin board—<http://www.careermosaic.com/>
- *High Technology Careers* magazine site—<http://www.careerexpo.com>
- College and university Web sites—<http://web.mit.edu/cdemello/www/P.html>
- Regional housing costs—<http://www.homefair.com> and <http://www.homes.com>
- Internships page—<http://icpac.indiana.edu/interns.html>
- Brain Book's Internship Index—<http://www.reg.uci.edu:80/UCI/SOP/BRAINBOOK/internship.html>
- Career Mosaic Internship listing—<http://www.careermosaic.com/cm/cc/cc1.html>
- Scientific Placement—<http://www.scientific.com/>
- Trattner Resources—<http://www.tratnet.com>
- *MacTech Magazine* ads—<http://www.xplain.com>
- *MacSciTech Journal*—<http://www.macscitech.org/>
- Newspaper Web sites—<http://www.newslink.org/>
- User Group Connection—<http://www.ugconnection.com/>
- Standard Rate and Data Service—<http://www.srds.com>
- Apple Worldwide Developers Conference—<http://www.apple.com>
- MacHack—<http://www.machack.com>
- Macworld Boston and San Francisco—<http://www.mha.com/macworldexpo/highlights.html>
- PC Expo New York—<http://www.shownet.com>

Apple Developer Catalog Ordering Information

To place an Apple Developer Catalog order from within the United States, contact Apple Developer Catalog at 800-282-2732; in Canada, call 800-637-0029. For those who need to call the U.S. office from abroad, the number is 716-871-6555. Or, send e-mail to APDA@applelink.apple.com. The Apple Developer Catalog is also available online on the Web (<http://www.devcatalog.apple.com/>).

good news about how and why Macintosh development is thriving.

Finish your writing "assignment" as soon as you can, and e-mail it to me. You could help convince developers who haven't yet experienced the joys of Macintosh to join the cause.

Paul Dreyfus
Editor