



AppleDirections

Inside This Issue

Editor's Note: Dr. Amelio Speaks	2
IndustryWatch: Profiting From Gil Amelio's Experiences at National Semiconductor	3
Apple Debuts New Server Hardware	11
PC World Gives Apple Top Spot in Two Studies	13
MessagePad 130 With Newton 2.0 and Newton Press	13
OpenDoc Round-Up	14
CD Highlights: System Software/SDK Edition, April 1996	15
Human Interface: Use Your Illusion	16
Client/Server Computing: An Overview for Information Systems Developers	18
Six Assumptions That Can Kill Your Multimedia Title	23
The Changing Landscape of Software Distribution	26
It Shipped!	29
Developer University Schedule	31
Apple Internet Page	32

Apple News

Apple Licenses Mac OS to Motorola

Agreement Allows Motorola To Sublicense Mac OS

Motorola became the most recent—and the largest—Macintosh licensee when it announced its broad Mac OS licensing agreement with Apple Computer, Inc.

The agreement allows Motorola to distribute the Mac OS with computer systems based on the current Power Macintosh hardware architecture, as well as those based on the PowerPC Platform (PPCP)—formerly CHRP—specification released by Apple, Motorola, and IBM in November. The agreement also makes Motorola the first licensee to be able to sublicense the Mac OS to other manufacturers. Apple will continue to certify all Mac OS-based systems to ensure compatibility. Motorola immediately announced that it is licensing the operating system to Nanjing Power Computing LTD, which will manufacture Mac OS-based systems for consumers and educational customers in the rapidly growing Chinese market.

Apple expects that Motorola will further expand Mac OS market share, providing thousands of new customers for your products. It will strengthen the platform's business proposition by providing additional sources for Mac

please turn to page 10

Strategy Mosaic

Looking Forward to the Copland User Experience

By Gregg Williams, Apple Directions staff

The Mac OS has always been known for the high quality of its user experience, and Apple has been working to increase the lead it has over the competition in this area. The upcoming Copland operating system must significantly "raise the bar" in the user-experience realm for the Mac OS platform to offer a superior, "next-generation" user experience. Fortunately, Apple believes it has done so with the Copland user experience.

Many elements, both present and future, will combine to create the overall Copland user experience. Currently existing elements includes the Mac OS user experience as we know it today, OpenDoc, the QuickTime family of technologies, QuickDraw 3D, QuickDraw GX, and international support. Future elements include Cyberdog (which will be integral to Copland) and new user-experience elements that I'll refer to collectively as the "Copland user experience" (see the figure on page 5).

Before you can figure out how your software needs to change to be a full player in the

please turn to page 4

AppleDirections

Volume 4, Number 4

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

Editor

Paul Dreyfus (AppleLink: DREYFUS.P)

Technical Editor

Gregg Williams (GREGGW)

Business & Marketing Editor

Kris Newby (NEWBY.K)

Associate Editor

Anne Szabla (SZABLA)

Production Editor

Lisa Ferdinandsen (LISAFERD)

Contributors

Peter Bickford, Alex Doshier, Dee Kiamy

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

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

Editor's Note

Dr. Amelio Speaks

As we went to press this month, I received a transcript of Dr. Gil Amelio's keynote address at the Seybold Publishing Seminar held in Boston at the end of February. Dr. Amelio has yet to describe his specific plans for leading Apple Computer, Inc., into the future, but the following selection from his remarks at Seybold will give you a general idea of his early thoughts on where he'd like to take the company.

What Apple needs to do is continue to make sure, in this era of common standards and homogeneity, that we stand out as a differentiated platform. That we give the user a unique experience. We want to continue to focus on the richness of your experience as a key element of our strategy going forward.

Clearly, I think users want to be able to focus on their creative nature and not really think about using the computer. They want it to be a lot more natural. That means better tools, faster speeds, and better modeling capability. We all want greater productivity. I think we'll see a more natural interface and I think it will be a more joyful experience. One of the things we don't talk about too much any more is that computers should be fun. We want to go back to the roots of making sure that Macintosh computers are fun while allowing users to be more productive and more creative.

How does Apple get to this bright future, considering the difficulties we face? I don't have all the answers at this point. But I do think there are a few things we can do right off the bat. Number one is we've got to get back to basics and the fundamental business of how we design, how we develop, how we produce, and how we service. And we have to do this far better than we've done it in the past.

We need to simplify the product line. We need to simplify that a lot, because complexity costs money, and we've added costs unnecessarily as a consequence. We need to improve our time to market, get things out when we say we will. We want to build on our key markets, such as publishing, because I view that as one of our

strengths, and, of course, we want to focus on promoting the brand.

We've got to get the organization aligned behind a common and consistent basic strategy. We have to simplify and strengthen the organization. And, we clearly have to acknowledge the critical value of our incredibly loyal customer base and make sure that we can keep them in the fold as we go through the necessary steps to reposition and rebuild the company.

It's not going to happen instantly. Tomorrow morning you're not going to suddenly have all the concerns you have about Apple cured. But, if we can keep the faith during this period of time then in fact we'll be on track to make it happen. I view that as a very, very critical element of the transition.

We don't want to be a slave to market share. What we really are trying to achieve is to make sure that we have an economic value proposition for developers and users so that they know there is a community of substance. In my experience, the way you get market share is not by focusing on market share. Market share is something that's a result of doing a lot of things right. Frankly, in the very recent history of the company, we focused a lot on market share but not necessarily enough on doing the right things. And that's what we're going to try to get back to and that's what I meant by focusing on the basics.

You can read some of Dr. Amelio's ideas about corporate "transformation" (his word for changing the fortunes of under-achieving businesses) in this month's IndustryWatch, on the facing page. We'll let you know about his specific ideas for Apple as soon as we know more ourselves.

Paul Dreyfus
Editor

IndustryWatch: News and Perspective

Profiting From Gil Amelio's Experiences at National Semiconductor

Before Gil Amelio joined Apple Computer last month, he spent several years honing his executive management methods, first as president of Rockwell Communications Systems, a division of Rockwell International, next as president and CEO of National Semiconductor. In both roles he took ailing businesses and led them back to prosperity; to use his term, he "transformed" them into thriving operations.

Dr. Amelio recently published a book, *Profit From Experience*, coauthored with William Simon, in which he describes his approach to management, particularly the techniques he used at National Semiconductor. We thought you'd be interested in a few samples of what Dr. Amelio has to say. What follows is excerpted from *Profit From Experience*.

Magazine writers have described me as a "turn-around manager." It's a label I thoroughly dislike, a "hot button" that launches me into a super-charged explanation of the difference between "turn-around" and "transformation."

A turn-around manager is an executive who, when he's brought in to save a troubled company, follows a ruthless formula. In the extreme case this involves slashing all budgets not directly involved with generating revenues, decimating the research expenditures, selling off a few operations to raise cash, and laying off half or more of the work force.

These quick-fix solutions don't address any of the business disciplines or management problems, but only hack away at every corner that might produce a fast improvement in the profit picture. This creates the appearance of health, and the manager is hailed by shareholders and the analyst community for his remarkable achievement.

He soon departs, significantly richer, beaded for his next position as a short-term hero somewhere else. What he leaves behind is a company no longer able to function productively, a company without the research to create new products, without the staff to support a marketing effort or provide customer service—in short, without any means to sustain cash flow and grow. The extreme turn-around manager is a pirate who sails away leaving a floundering crew on a plundered ship heading toward disastrous storms.

• • •

At [my] first off-site [meeting] with the National Semiconductor executives, I called for a return on equity of 20 percent—which is what the best companies in the industry achieve.

April Apple Directions Online

April's *Apple Directions* will be available by March 15 at the following locations:

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

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

National's ROE was then a large negative number, and you could bear stifled, embarrassed laughter from around the room because what I was asking for seemed so unrealistic and un-achievable. That was my bold vector; subtle as a two-by-four. It was designed to get their attention, and it did.

The company's gross profit was then 24 percent. Another target I set called for a gross profit of 40 percent; three years into the effort, we had already achieved 43 percent. The difference represents some half-billion dollars annually.

These goals seemed unrealistic. But we wanted people to know they were being challenged to make a significant stretch.

If you call for a 10 percent improvement, your employees will just work harder. They'll produce the 10 percent for you, and in normal times that may be worth celebrating. But in a transformation, 10 percent does not do it; you must make demands that require a paradigm change.

At National, we changed the entire corporate accounting system, set seemingly impossible financial targets, revamped the organizational structure (twice) and redefined some of our key ideas about what business we are in. And those are only a few of the "bold actions" we undertook.

• • •

So you're going to inherit the management team left behind by your predecessor. But if the organization is in such need of transformation, they must be part of the problem—they must bear a large measure of responsibility for having allowed the situation to get so badly out of hand. How can you rely on them?

I offer this guideline:

Don't attribute to malice that which you can attribute to ignorance.

Perhaps it's a form of xenophobia that we have a tendency to see as villains those people who approach their work differently than we do—if they don't have the same work habits, if they don't follow the same management principles, if they don't relate to their people the same way, if they don't have the same view of what the problems are, then they must be evil. If one division is doing poorly compared to the others, the manager must be incompetent.

Those who do it my way are okay; everybody else is one of the "bad guys."

A judgmental approach like that courts disaster. As a rule of thumb, you can assume that 80 to 90 percent of your people are competent and well-intentioned. . . . I recommend you start with confidence in your people, let them clearly know what you want, and then wait to see who comes through and who doesn't.

Implications/Opinions: You can learn a lot about how to improve your business from Dr. Amelio's book. (And, as he points out in the introduction, even successful businesses have room for improvement.)

You'll also get a glimpse inside the mind of the person who'll be the hub of the Apple development universe for the foreseeable future. If you want to stay in sync with what Apple will be doing in the coming months, we think you'll want to profit from Dr. Amelio's experience and take a look at his book. If you can't find it at your local bookseller, contact Schwartz Business Books at 209 East Wisconsin Ave., Milwaukee, WI 53202; U.S. phone 800-236-7323; U.S. fax 800-231-7323; non-U.S. phone 414-274-6406; non-U.S. fax 414-274-6408; e-mail address SchwartzBB@aol.com.

UMAX Buys Mac OS Compatible Business From Radius

Motorola's agreement to sell Mac OS-based personal computers isn't the only exciting recent news on the Mac OS licensing front. In January, UMAX Data Systems of Taiwan agreed to purchase Radius's Mac OS compatible business, which includes its Mac OS license, PCI designs, vendor and customer contracts, and a research and development staff. Under the terms of the deal, UMAX Data and Radius will form a new company, UMAX Computer Corp., to develop and market PCI-bus, PowerPC processor-based Mac OS-compatible computers. UMAX Computer will also receive a license for exclusive rights to use the SuperMac brand name. UMAX Data Systems will own an 80 percent stake in the new company; Radius will hold the remaining shares.

UMAX Data Systems is a subsidiary of the UMAX-Elite Group, which makes a wide variety of computer products, including scanners, logic boards, graphics accelerators, and software. UMAX-Elite Group is also the second largest system board maker in Taiwan, which today provides the infrastructure base for much of the worldwide personal computer industry. (Major PC vendors depend heavily on Taiwan for low-cost logic boards.)

Implications/Opinions: The UMAX/Radius agreement is good news for the Apple community on several fronts. First, UMAX Computer is capable of manufacturing and shipping high volumes of Mac OS-compatible systems. Second, while UMAX Computer will continue to target the high-end graphics user that was Radius's focus, it appears that UMAX will also pursue the low-margin/high-volume market. Finally, the relationship with UMAX will not only boost Apple's presence in

the rapidly growing Asian personal computing market, but should also provide another source of local language application development for the Mac OS.

mTropolis Games, Multimedia Development Environment Ships

Amid all the Macintosh and Newton development tools announcements at January's San Francisco Macworld (see "New Developer Tools Announced at Macworld" on page 19 of the March issue of *Apple Directions*), there's one important announcement we missed—at least until now. That's the shipment of mTropolis by mFactory, Inc., a developer based in Burlingame, California. mTropolis is an object-oriented environment for game and multimedia developers. The development environment itself is available only on the Macintosh computer; it requires at least a 68020-based Macintosh system and 5 MB of RAM. The software produced with mTropolis can be run on either Macintosh or Windows systems without any modification. mTropolis contains literally hundreds of reusable text, graphics, sound, and video objects; creating titles is as easy as dragging and dropping the objects. The environment is based on mFactory's core technology, which it calls *mFusion*. By extending mFusion's optimized C++ core with your own code, you can add other objects to mTropolis and change the behavior of existing objects.

Implications/Opinions: Apple loves mTropolis, because it's currently a Macintosh-only environment that produces truly cross-platform code. Developers and multimedia authors who have already worked with it attest to its intuitive user interface and the speed of development/authoring it provides. If you develop multimedia titles or games, we think mTropolis is worth a look. To find out more about it, visit the mFactory Web site (see page 36 for the location) or contact mFactory directly (e-mail—info@mfactory.com; phone—415-548-0600; fax—415-548-9249). ♣

Excerpts are used with permission from Profit From Experience, published by Van Nostrand Reinhold, © 1996. All rights reserved.

Strategy Mosaic

Copland

continued from page 1

Copland user experience, you first need some knowledge of what user-experience elements will be added to Copland—which is what the first part of this article covers. The last major section of this article, "Preparing for the Copland User Experience," (page 9) lists some things you need to do or think about before you modify your software to take maximum advantage of Copland's

user-experience advances.

With that said, I'll begin describing the major elements of the Copland user experience, which are grouped into three main categories: information access, customization and control, and assistance.

Information Access

Copland will improve the Mac OS user experience by providing richer ways for users to get information from the computer. Here are the main user-experience features that Copland will provide to accomplish this.

- *Navigation Services and a new document model.* The Navigation Services API (application programming interface) does two main things: It replaces and greatly improves the classic standard file dialog box (used for choosing and saving files), and it provides a direct-manipulation interface for people who don't want to deal with the standard file dialog box at all.

The main problems with standard file are that it isn't (standard, that is) and that it gives users too little information. Currently, the appearance of a standard file dialog

box can change, depending on how you use it. The Navigation Services API will allow you to customize the choose/save file dialog box while still presenting the user with a standard appearance. In addition, the Navigation Services dialog box will also give the user more information than today's standard file dialog box does—for example, a file's preview image and the Get Info box data.

To accommodate users who prefer using the Finder, Copland will introduce a new direct-manipulation model for creating and saving documents. Under this

model, when users create a new document within an application, your Copland-savvy application should immediately save the new document to disk in the default folder that has been specified by the user's preferences; also, the icon for the new document should have the same name as the document's window (for example, "Untitled"). Users can then move or rename the document's icon as they desire. In addition, Copland will notify your application whenever the user changes an open document. (Your application should use this information, for example, to change a document's name in its window's title bar to reflect a change made from the Finder.) This new document model enhances the user experience by giving customers a choice in how they wish to manipulate files—from the Finder or from within the active application.

In addition, all windows in Copland will contain a small icon to the left of the window's name in the title bar. Users can directly manipulate the item the window represents by dragging the item's title-bar icon to a destination elsewhere on the screen. If the destination is on the same storage device as the item, this action moves the item; if it is on a different storage device, this action copies the item.

- *Better direct manipulation.* The Macintosh user interface was designed around low-capacity storage devices—a 400K floppy drive, to be exact. Now, users routinely deal with hundreds or thousands of folders (and gigabytes of storage). Because of this, simple actions—in particular, moving an item to another location—have become cumbersome tasks that involve cluttering the desktop with numerous open windows before users can drag an item to a new location.

The Copland user experience includes several new interface elements to make it easier for users to manipulate items directly. In addition to the title-bar icon, mentioned earlier, Copland adds the *spring-loaded folder*, which causes a folder to open when an item is dragged over it. By dragging an item over successive folder icons, a user can drop the item into a deeply embedded folder with one movement. When the user finally drops the item into a window, all of the intermediate windows opened during the extended drag operation close automatically, thus reducing screen clutter.

Another interface element is called the *pop-up window*; it allows windows to remain open without cluttering up the desktop. Under Copland, users can drag a window to the bottom of the screen, where its title bar is shortened to allow multiple title bars to show. When the user drags an item onto a shortened title bar, the full window displays itself, and the user can drop the item into the window (see the figure on page 6). When the user activates another window, the pop-up window minimizes itself again at the bottom of the screen.

- *Saved find and find-by-content searches.* Imagine the user being able to save a System 7.5 Find File search specification in a form that looks like a folder. When the user opens the "folder," it opens into a window that displays every icon that meets the search specification. That's a pretty accurate description of the new Copland user interface element called a *saved find*, and almost everyone who has seen it has given it a very enthusiastic response.

In addition to the matching criteria allowed by today's Find File function (search based on the file's name, type, creation date, and so on), the Copland saved find can also perform relevancy-ranked *find-by-content* searches—that is, it can select files based on whether the files themselves (not just their filenames) contain specified text (see the figure on page 7). This feature is made possible through the use of a very fast Apple-developed text-indexing engine called *V-Twin*.

- *Contextual menus.* Balloon Help has been Apple's way of allowing users to get help on an item by pointing at it. Copland extends this ability with contextual menus, which allow users to display a pop-up menu containing

various useful functions that are specific to the item being pointed at. Depending on the item itself, this menu may contain items to show the balloon for the item being pointed to and the Guide for the active application (if they exist), as well as additional items dictated by the application or system software (for details, see the figure "A contextual menu" on page 8).

- *Multi-threaded, scalable, customizable Finder.* Under Copland, the Finder will be multi-threaded. This means that, for example, users will no longer have to wait for a file to be copied or a window to display its contents before doing other work. (The latter operation is often time-consuming when the window being opened is from a remote volume.) This means that the Copland Finder will seem more responsive to users, and they can initiate time-consuming operations without worrying about them interrupting their workflow.

In addition, the Copland Finder will be "scalable"—that is, it will finally answer the "one size does *not* fit all" criticism that is sometimes leveled at today's Finder. The scalable Finder will allow a range of

Information access	Customization and control	Assistance
Navigation services	Appearances (themes, settings, scenarios)	Help
New document model	Workspaces	Balloon Help
Spring-loaded folders		Apple Guide
Pop-up windows		Tips
Saved find and find-by-content		Delegation
Contextual menus		Triggers
Scalable Finder		Notifiers
		Scheduled Tasks
		Experts

Copland user-experience features.

customizations that will allow a Mac OS computer to be fitted to a user's age, expertise, and preferences. One customization that I've been waiting for is the ability to show, hide, and reorder the columns of information in the list view of any window. (Finally, I'll be able to see the ends of my long filenames!)

Customization and Control

Customization has always been a welcome part of the Mac OS experience. Who hasn't renamed the hard disk (mine is named "Mega Gerbil"—don't ask), selected just the right desktop pattern, and replaced the plain-vanilla system beep sound with something that has more personality? Some regard customization as a superficial feature, but it's something that makes many people

feel more comfortable with their Mac OS computers—which is important.

Copland takes customization to new levels with what are called *appearances*—specifications of what the human interface looks like. In fact, Copland engineers have added an Appearance Manager to the Mac OS Toolbox, so that you have a standard way of changing human-interface elements and don't have to write custom code—which may break in the future—to do so.

Also, a new visual "look" for the desktop is something the next version of the Mac OS needs to look attractive to prospective buyers. However, Apple found it impossible to create one "look" that would be attractive to all buyers. This led to the creation of appearances, which provide a framework for giving users a

choice of new "looks" for the desktop.

Copland's overall appearance is dictated by several elements. The first element, called *themes*, dictates what the user interface—which includes windows, menus, and controls—looks like. (For examples of possible themes, see page 8 of "Copland: Technology for Customers' Sakes," in the June 1995 issue of *Apple Directions*.) The second element, called *settings*, includes such user choices as screen savers, desktop animations, sounds, and more.

Together, the choice of a theme and a collection of settings constitutes what Copland engineers call a *scenario*. Copland will include a choice of predefined scenarios. Most users will customize their Mac OS computers quickly and easily by choosing a predefined scenario. However,

power users will also be able to mix and match themes and settings if they feel the need to do so.

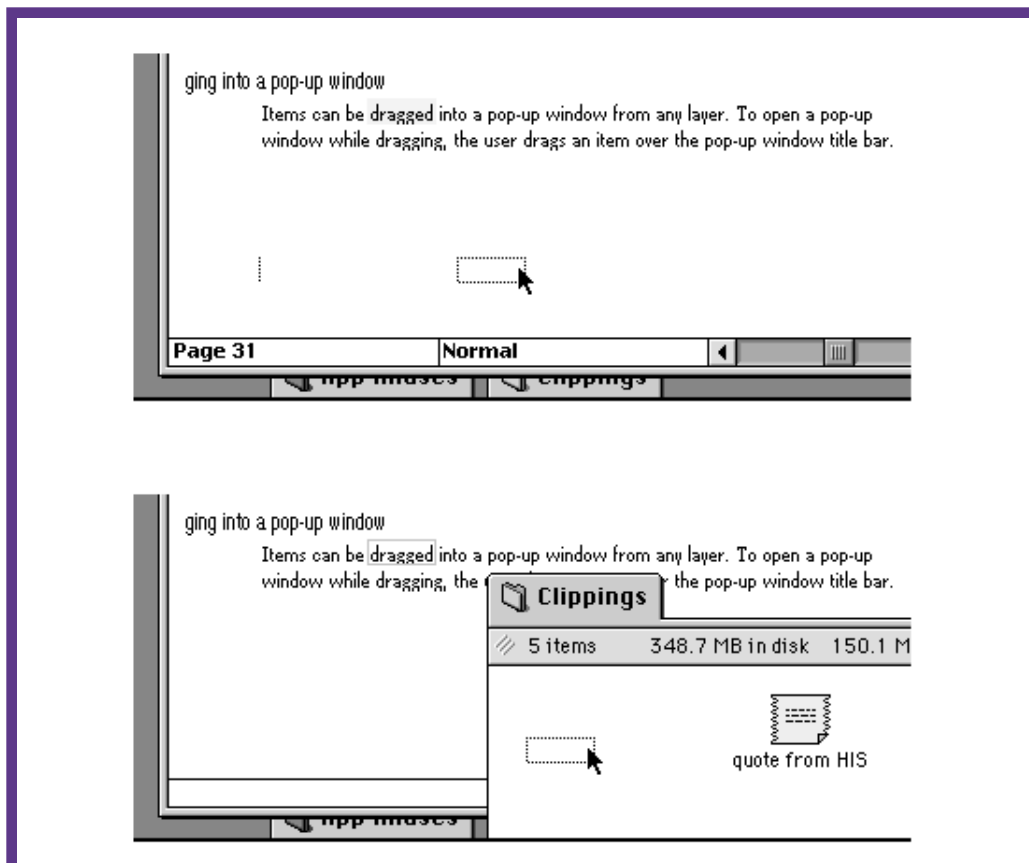
Copland will supply a basic set of appearance elements, and there will certainly be a market for third-party themes and settings. In fact, Apple will provide an OpenDoc-based theme development toolkit that will enable human interface designers to create themes without having to directly manipulate the resources that store theme information.

However, the level of customization that Copland allows is at odds with the fact that many Mac OS computers are used by more than one person. To allow multiple users to take advantage of the customizations that Copland offers, the Copland designers added a feature called *workspaces*. Workspaces allow different users to identify themselves when they begin using a Mac OS computer; the computer then presents them with a customized computing environment. A user's workspace interacts with that person's personal choices, including preferences files, system font, Startup Folder, Shutdown Folder, and Apple menu items. In addition, a workspace will include its own appearance scenario, as discussed earlier.

Another important aspect of workspaces is that they allow the person who sets them up to control the amount of access the workspace user has to the system (for example, whether or not the user can access the entire hard disk or just the user's folder). This gives the computer's administrator some control over how much the workspace users can affect the shared computer.

Assistance: Help

The assistance component of the Copland user experience includes three separate areas: *help*, *delegation*, and *experts*. I'll start by



Opening a pop-up window. When the user drags an item over the pop-up window title bar (top image), the pop-up window opens to its full size (bottom image).

describing the three ways in which Copland can give the user specific help: Balloon Help, Apple Guide, and tips.

Today, Balloon Help gives users a quick answer to the question, "What is this item?" It will continue to work under Copland, and Apple encourages you to make greater use of it in the future. Apple believes that contextual menus (discussed earlier) will make users more likely to use Balloon Help to learn about an individual item without having to manually turn Balloon Help on, then off.

Apple will make numerous improvements to Apple Guide. The current version of Apple Guide, version 2.0, added such features as backward compatibility with System 7.0 and 7.1 (allowing these earlier versions of System 7 to access existing guide files), asynchronous sound playback (playing a lengthy sound will not cause Apple Guide to wait until the sound finishes), and global guide files (available regardless of which application is active). Apple Guide 2.1, available later this year, will provide support for OpenDoc.

The biggest changes will come with Apple Guide 3.0, which is the version that will ship with Copland. Among other features, this version will be tailored to the Copland architecture and will allow 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 panel sequences.

Copland will also add a new level of help called *tips*, implemented through a body of software called the Tip Manager. Users will be able to invoke a Tips window, which will contain a list of tips relating to recent

user actions (see the figure "The Tips window" on page 8). Apple will supply a database of tips, and you can add tips that relate to your application.

Users will be able to specify three levels of tip activation. First, they can elect to be shown tips only when they select the Tips command in the Help menu. Second, they can have the Help menu flash when the system decides that it has a relevant tip to show. Third, they can have the system automatically show a tip.

If you create tips for your application, you can easily add code to your application that will allow the Tip Manager to alert the user to your tips (assuming, of course, that the user has selected the second or third level of tip activation).

Here's an example of how such an automatic alert might work. Say you've written a tip

Human Interface Help at WWDC

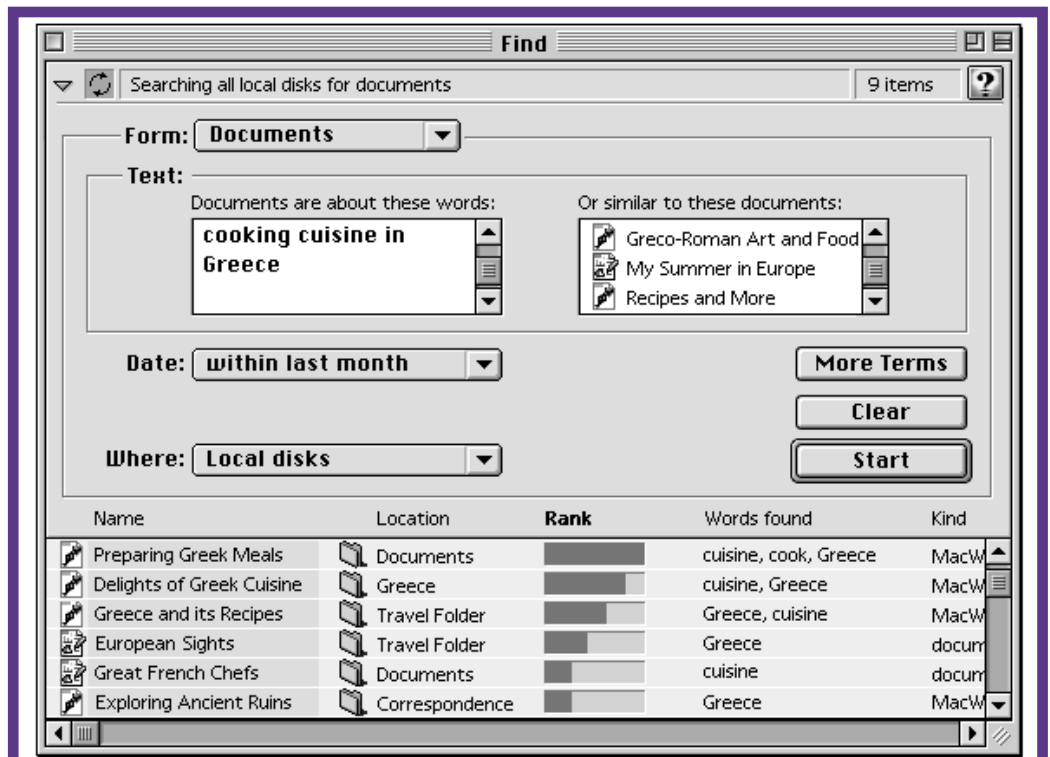
If you are planning to attend the Apple Worldwide Developers Conference (May 13–17, San Jose CA), be aware that Apple will be hosting a Human Interface Lab there. Apple human interface people will be at the lab to answer your human interface questions and critique your applications. Details will be available in future WWDC materials. In the meantime, you can get information on the WWDC from the Web page listed in the "Internet Resources for This Issue" box on page 36.

that reminds users that they can close a window using Command-W. The Tip Manager keeps a numeric value associated with each tip. All you need to do is add code to your application that calls a certain Tip Manager routine whenever the user clicks a window's close box. This routine increments the value associated with your tip. The Tip Manager monitors this value and

alerts the user to the availability of this tip (based on the tip-activation level the user has chosen) when the tip's value exceeds a set threshold value.

Assistance: Delegation

The Copland user experience also includes the ability for your application to specify a deferred task, ensure that it gets performed, and communicate the task's



Find-by-content searches. This Copland feature allows users to search for files that contain certain words or are similar to other documents. The files found are ranked by how closely they match the specified criteria.

status with the user. Apple calls this ability *delegation*, which is a form of assistance that enables the computer to act on the user's behalf, even when the user isn't present. Copland implements delegation through three mechanisms called *triggers*, *notifiers*, and *scheduled tasks*.

One pervasive change in the Copland architecture (when compared to the Mac OS today) is its reliance on event notification rather than polling to handle

the complex interaction of tasks. Today's software often uses polling to repeatedly check some aspect of the computer until a desired condition occurs.

Because polling wastes valuable processor cycles, there is a limit to the amount of polling that all the software (both system and application) on a computer can do without degrading performance.

Copland virtually eliminates polling inefficiency by using an

event notification model instead. Under Copland, a piece of software asks to be notified through an Apple event when a certain condition occurs. It then goes on with its work; when it receives the notifying Apple event, it can execute the code associated with the condition that has just been met.

Using new Copland software called the Trigger Manager, your application can schedule some task to be performed at a later date. Here's an example: Say you want to add a "Backup files at shutdown" menu command to your program. When the user selects this command, your application tells the Trigger Manager, "When the user begins the shutdown sequence, send the Apple event I have specified to me—and if I'm not a running application at that time, launch me and then send me the Apple event." (This instruction, saved by the Trigger Manager, is called a *trigger*.) Your application should also recognize the specified Apple event and back up the appropriate files when your application receives it.

The Trigger Manager also makes it easier for your application to get whatever user input might be needed to set up a trigger. Suppose that, in the example above, you want to let the user specify which of three file sets should be backed up. When the user selects the "Backup files at shutdown" menu command, your application should first use a supplied routine to bring up a dialog box called a *trigger picker*. Your application would then draw three checkboxes that would allow the user to pick which file sets should be backed up. Once the user has dismissed the trigger picker, your application can extract the user's input from it and use that data to specify the details of the trigger.

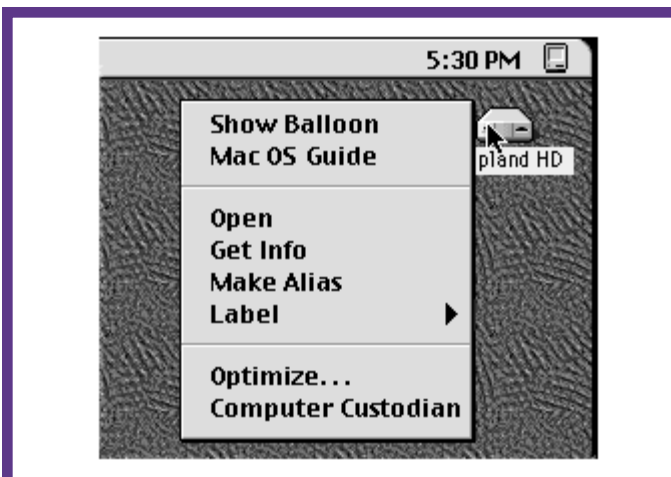
A mechanism related to the trigger is the notifier, which is

controlled by the Notification Manager. Whereas your application can use a trigger to specify a deferred task, it can use a notifier to tell the user whether a deferred task was completed successfully. A notifier might communicate with the user through making an entry in a log file or displaying a dialog box.

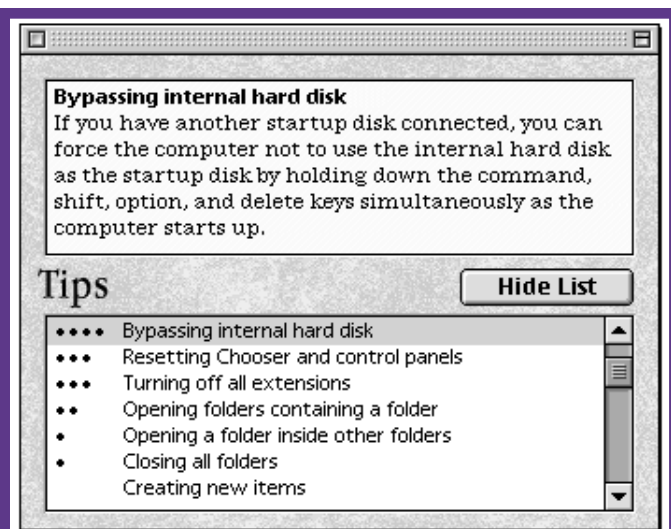
Fortunately, the architecture for triggers and notifiers is modular. Apple will supply certain basic triggers and notifiers with Copland, but you can also create your own. This gives you two development opportunities. First, you can create triggers and notifiers that are tied to your application and make it more versatile. Second, you can create useful triggers and notifiers and sell them as their own product. (For example, you might create a notifier that would notify users by sending them a pager message.)

One variation on triggers is scheduled tasks, which are managed by the Scheduled Task Manager. Without going into technical details, it's enough to say that a scheduled task is a trigger that survives system restarts. (Unexecuted triggers are stored in memory and are lost when the computer is restarted or shut down.) Scheduled tasks are implemented as AppleScript script files that are created "on the fly" by your application. The Scheduled Task Manager, the Trigger Manager, and other system software work together to ensure that the associated script gets executed when the triggering conditions occur.

Of course, the user needs to know what deferred and scheduled tasks are currently pending. Copland will make this information available through the About This Computer window: Each piece of software listed in this window will display a list of all the deferred and scheduled tasks associated with it.



A contextual menu. In this example, the user points at a hard disk icon and causes its contextual menu to appear. Such menus contain, from top to bottom, standard balloon and guide menu commands, menu commands inserted by the active application (in this case, the Finder), and experts relevant to the item being pointed at.



The Tips window. When the Tips window opens, it supplies a ranked list of tips based on recent user actions.

Assistance: Experts

The Copland user experience adds one final level of user assistance: experts. In contrast to Apple Guide, which teaches users how to do a well-defined task (like changing the screen depth), experts help users achieve a desired goal without teaching them how to complete it. One example of an expert is the Setup Expert, which will be included with Copland. This expert will help users perform all the little tasks needed to properly set up a new Mac OS computer—for example, getting the user connected to the local network and Internet, and setting up the computer's owner name and password, time and date, default printer, and geographic and cultural preferences.

Experts work by asking a user a set of questions, formulating a series of actions that will accomplish the desired goal, and (after getting the user's permission) carrying out those tasks. I need to point out, however, that a Copland expert is not simply a sequence of dialog boxes that the user makes selections from. Instead, a properly designed expert should isolate the user from as many details and make as many decisions for the user as possible. In addition, Apple will provide tools that will help you create your own experts.

Preparing for the Copland User Experience

Now that you know about the major features that the Copland user experience will provide, I can talk about various ways in which Mac OS software will need to change to make the best use of these features. The following is a laundry list of things you can do now to ensure that your software makes the best use of Copland user-experience features:

- *Implement Apple Guide.* Users value Apple Guide, and

your application will stand out if you include Apple Guide support. Don't worry about "losing" the work you do today to an incompatible future version of Apple Guide—today's guide files will continue to work in the Copland time frame and beyond.

Also, Apple studies have shown that users prefer guides that do the work for them rather than showing them how to do the work themselves. Apple-supplied guides in the Copland time frame will feature more "do it for me" assistance, and Apple recommends that you do the same. (For more information on how Apple Guide works, see "How Do I Use Apple Guide?" on page 14 of the September 1994 issue of *Apple Directions*.)

- *Make your application scriptable.* Scriptable applications (that is, ones that other software can direct to do work by issuing them Apple events) will become more important in the Copland time frame. For example, Apple Guide accomplishes "do it for me" assistance by issuing Apple events to the Finder and other running applications. If your application is not scriptable, this may limit how useful your application is in a world that includes guides and experts.

In addition, applications that follow the Open Scripting Architecture object model automatically make available a wealth of information about them and their current state. Software can query such applications and make good use of the information. (This is true even today, but Copland will make it more likely that such information will be put to use.)

For example, if your application implements the object model, you could create a Copland expert that, through Apple events, might ask your application what windows it has open. Using that information, it could deduce what documents are open and do

What You Can Do Now

Even before Copland arrives, there are a number of things you can do to prepare for the transition. The following items are covered more fully in the section "Preparing for the Copland User Experience":

- implement Apple Guide
- make your application scriptable
- implement Macintosh Drag and Drop
- use the Preferences folder properly
- support document translation
- use Balloon Help
- isolate custom code

something useful—such as copying the documents to a backup volume.

- *Implement Macintosh Drag and Drop.* Make sure your application supports Macintosh Drag and Drop, especially to and from the Finder. Copland will make heavy use of drag-and-drop behavior, and users will regard your application more highly if it integrates smoothly into the Copland environment.

- *Use the Preferences folder properly.* Ideally, users should be able to throw away their Preferences folder and not lose important data. Unfortunately, some applications use the Preferences folder to store important, non-preference data.

Though this practice has no serious consequences today, you should revise your application to put nothing but preferences data in the Preferences folder. The reason? Copland's implementation of workspaces includes a Preferences folder for each user; data that needs to be "visible" regardless of which user is active will not always be visible if it's located in one user's Preferences folder.

- *Support document translation.* You should make sure that your application supplies type and kind resources. That way, any service that uses the Translation

Manager then has access to the text information in your documents. This becomes especially important in the Copland time frame, because the "find-by-content" feature uses the Translation Manager to extract the file's text to see if the file meets the search criteria.

- *Use Balloon Help.* As I mentioned earlier, Copland will make Balloon Help easier to use. Make sure your application supports Balloon Help, so that it will be there when users ask for it.

- *Isolate custom code.* The Copland high-level Toolbox (which includes the Window, Menu, Dialog, Control, and Appearance Managers) will offer standard ways to do things that today require custom coding—for example, floating windows, keyboard equivalents for menu commands, and grid or other non-standard menus. By isolating custom code in your application today, you will be more easily able to replace it in the Copland time frame with code that will continue to work in post-Copland operating systems.

In addition, there are other things you should do once Copland becomes available. I'm including the following list because it wouldn't hurt for you to start thinking about those changes now:

- *Use experts.* When designing a Copland expert, it's not enough to display a series of panels that step users through the same sequence of actions that they would have taken to accomplish a given task manually. Rather, the philosophy behind experts is to do as much work as possible for users. This includes designing your expert to get the maximum amount of information from the fewest possible questions. It also includes asking questions that elicit from users what they want to do (then taking care of the details), rather than burdening them with questions that ask them exactly how the task is to be accomplished.

- *Add deferred tasks.* Everybody hates waiting during time-

consuming tasks that somehow tie up the computer. Copland makes it easier to add features to your application that will allow users to define deferred and scheduled tasks. Since a task can use Apple events to control multiple applications, you will be able to automate complex tasks for your users. Even now, you should start thinking about what kinds of deferred tasks your users would like to see.

- *Make your application scalable.* In the past decade, the Mac OS community has learned that it is virtually impossible to balance the conflicting priorities of number of features, ease of learning, and ease of use. In addition, the community has also learned that "one size does not fit all"—that is,

that one feature set does not fit all users' needs.

One of Copland's goals is to be scalable, and when you redesign your application for Copland, you should consider modifying it to offer some level of scalability to your users.

Start Planning for Copland

Among the goals of Copland are increasing user productivity, delivering a computing environment that is superior to those of competing platforms, and providing a growth path for future enhancements to the Mac OS computing experience. The Copland user-experience features contribute to the realization of these goals by improving user access to data, giving users new ways to custom-

ize and control their computing environment, and increasing the computer's ability to assist users in getting their work done.

The previous section lists various things you can do now to prepare for changes the Copland user experience will bring. But you should also step back and take a look at how the Copland user experience will enrich users' interactions with the computer and the software that runs on it. Copland will be in your hands soon. You need to start thinking now about how you will enhance your software to make it a compelling product in the post-Copland world. ♣

Apple News

Motorola

continued from page 1

OS-based systems and a broader range of part suppliers as well as feature-set, price, support, and distribution choices.

"We believe that Motorola, a key player in this industry, has the resources and technical strengths, especially in the communication area, to help enrich the Mac platform and to expand its benefits to new markets and new customers," said Dr. Gil Amelio, chairman and CEO, Apple Computer, Inc. "This agreement is a major milestone in our licensing program, and is a clear demonstration of our commitment to open licensing."

The recent announcement heralds the beginning of a second phase in Apple's Mac OS licensing activities, which were first announced in September 1994. The first phase has involved companies that build Mac OS-based

systems using Apple proprietary hardware design. The second phase will involve companies developing their own hardware designs based on the PPCP specification, which defines a PowerPC RISC processor-based architecture for computers that can run a variety of operating systems, including the Mac OS, AIX, OS/2, Windows NT, and Solaris.

In its own release announcing the agreement, Motorola said its Motorola Computer Group (MCG) division intends to "provide a range of low-cost, multiple operating system solutions based on the PowerPC Platform to support demand for the Mac OS. Motorola plans to distribute its Mac OS-based products to the enterprise marketplace worldwide by leveraging its existing corporate reseller, aggregators, and system integrator distribution channels. MCG is also planning to add new corporate resellers that focus on solutions for the enterprise marketplace."

Additionally, Motorola said, "MCG will use its recently

announced joint venture with Panda Electronics Group in China to distribute Mac OS-compatible systems to that rapidly expanding marketplace. Nanjing Power Computing LTD will use Panda Electronics distribution channels to distribute Mac OS-based desktop systems to consumer and education markets in China. In addition to this joint venture between MCG and Panda Electronics, Motorola has established partnerships with several organizations in the People's Republic of China to provide software development and localization support."

Apple clarified that it's not asking Motorola to take over existing Macintosh product lines or markets. For example, Apple will continue to compete aggressively in the Chinese market, where it has a strong foothold in the publishing and education sectors, even though Panda Electronics will actively market Nanjing Power's Mac OS-based systems. Apple views Panda's and Nanjing's efforts as complementary to Apple's, especially in the

business and consumer markets. Further, having additional providers of Mac OS-based systems in China is expected to drive product innovation, variety, and availability, as well as the development of localized applications.

The licensing agreement applies to versions 7.5.x of the Mac OS, and includes access to the next major release of the Mac OS. It applies to the 16 local language versions of the Mac OS that Apple has so far approved for licensing, including the following: U.S. English, U.K. English, Italian, Dutch, French, Spanish, Portuguese, German, Swedish, Norwegian, Danish, Finnish, Korean, Traditional Chinese, Simplified Chinese, and Japanese.

For further details about Motorola Computer Group and its plans for Mac OS-based computers, visit the Motorola Web site listed in "Internet Resources for this Issue" on page 36.

Apple Debuts New Server Hardware

At Boston's February Seybold Conference, Apple Computer, Inc., introduced its new "industrial strength" AIX-based Network Server line designed to compete in high-volume network environments. Apple also unveiled more powerful versions of its current Mac OS-based Workgroup Server line. Additionally, Apple announced the Apple Internet Server Solution 2.0, which couples third-party Internet server software with the latest Workgroup Server systems to provide complete Mac OS-based World Wide Web server solutions.

Apple is introducing its new servers at an opportune time: According to International Data Corporation (IDC), worldwide server sales increased more than 50 percent in 1995, reaching a total of about 850,000 units sold. IDC predicts that server sales will jump another 50 percent in 1996, driven by sharply increased demand for Web servers.

"Apple has leveraged its experience in the workgroup server market to provide solutions for a wider range of server requirements," said Brodie Keast, Apple's vice president for worldwide product marketing. "Organizations can now choose Apple server solutions not only for their workgroup server needs, but also for the demanding tasks of high-traffic World Wide Web sites, large-scale commercial publishing environments, and mission-critical data management."

Developer Opportunity For Network Server Administration Software

The new Network Servers introduce a software infrastructure called *AppleTalk Services*, which includes a special version of the

AppleTalk protocol stack and a UNIX® router. From a developer standpoint, the most interesting aspect of AppleTalk Services is a new application programming interface (API) for building server administration software.

With the AppleTalk Services API, you can write software that lets network managers control the Network Servers from a Mac OS client system on the network. Such software sends an Apple event from the Mac OS client to the Network Server; AppleTalk Services then "translates" the Apple event into one or more UNIX commands that are then carried out by the server.

Apple is encouraging developers to write administration software based on AppleTalk Services, which will make it far easier for network managers to control the highly complex AIX-based servers. Instead of having to learn UNIX commands, managers will be able to work with the much easier-to-use interface employed by Mac OS clients on the network.

The remainder of this article provides additional details about the new server products.

Network Servers: For Mission-Critical Business Environments

The Network Server line includes the 150-megahertz (MHz) Network Server 700/150 system and the 132-MHz Network Server 500/132 system, both of which are based on the PowerPC 604 microprocessor. The Network Servers feature a unique, highly expandable design that lets you quickly add and replace components—some without even shutting down the systems. The Network Servers don't include the Mac OS, and they won't run existing AppleTalk-based server applications. However, the servers are binary-compatible with AIX, which means they'll run the thousands of currently

available applications written for IBM RS-6000 systems.

The Network Server 700/150 includes a 150-MHz PowerPC 604 processor and 1 megabyte (MB) of level-2 memory cache, and can be expanded to use up to 512 MB of memory. The Network Server 700/150 has up to 32 gigabytes (GB) of internal disk capacity, with room for up to eight internal hard disk drives of up to 4 GB each. The Network Server 500/132 and 700/150 systems come with two built-in Fast/Wide SCSI-2 channels for optimizing hard disk performance and include two industry-standard PCI buses and slots for up to six PCI cards.

The Network Server 500/132 features a 132-MHz PowerPC 604 processor, 512K of level-2 memory cache, and can be expanded to use up to 512 MB of memory. It can support up to 24 GB of internal disk storage, with room for up to six internal hard disk drives of up to 4 GB each.

The Network Servers' unique enclosure allows easy component access as well as physical security, with key-controlled access to all major parts. All of the main server components, including the logic board, can be replaced within 60 seconds by experienced, if not technical, personnel. Disk drives and cooling fans for both systems are "hot-swappable," meaning they can be changed while the computers are turned on. Additionally, the Network Server 700/150 features two hot-swappable power supplies. Experienced personnel can remove, service, and replace one of the two power supplies while the other supply powers the system, without having to take the server offline.

"The new Network Servers are the perfect choice for commercial publishing environments, especially those already using Macin-

tosh computers," said Brodie Keast. "In addition to high reliability, the new Network Servers provide the high performance required in production departments where storing, moving and manipulating extremely large graphics files is common."

Apple chose IBM's AIX 4.1.4 UNIX operating system to run the Network Servers, because many publishing customers currently use UNIX-based servers in demanding server environments. The operating system, which Apple calls *AIX for Apple Network Servers*, ships separately from the servers.

Apple added high-performance network and I/O drivers to the Network Servers and a new version of the AppleTalk protocol stack, as well as the AppleTalk Services software infrastructure. Recent performance benchmarks conducted by Apple indicate that Apple's Network Servers offer superior Internet and Macintosh file server performance when compared to competing UNIX-based servers from Sun Microsystems.

Apple will initially provide two utilities based on AppleTalk Services—the Disk Management Utility to simplify disk administration and CommandShell, a multi-windowing VT102 terminal emulator included with AIX for Apple Network Servers. Also bundled with AIX for Apple Network Servers are several CD-ROM discs with evaluation copies of third-party software, including the following:

- American Power Conversion's PowerChute PLUS, which provides UPS monitoring and shutdown management
- Helios's EtherShare, EtherShare OPI, and PCShare, which provide Macintosh and PC file services and Open Prepress Interface (OPI) services
- IPT's uShare and CanOPI, which provide Macintosh file services and OPI services

- Legato's NetWorker for UNIX, ClientPak for Macintosh, and the Legato Data Backup Utility, which manage network backup, archiving, and recovery

Included with each Network Server is a one-year on-site hardware warranty, with lifetime hardware phone support and 90-day software phone support. The servers are expected to be available worldwide by early April. Network Server 700/150 system prices range from U.S. \$11,829 to \$14,999; prices for Network Server 500/132 systems range from U.S. \$8,819 to \$10,969, depending on configuration. The AIX for Apple Network Servers Accessory Kit, which includes the AIX operating system, retails for U.S. \$1,489.

Workgroup Servers: For Application Services, High-Performance AppleShare Networks, and Web Publishing

The new Workgroup Server 8550/132 and Workgroup Server 7250/120 systems, which replace the NuBus™-based Workgroup Servers, incorporate faster PowerPC processors, and the Peripheral Component Interconnect (PCI) bus. The Workgroup Server 6150/66 system remains in the product line as Apple's most inexpensive server.

The Workgroup Server 8550/132 uses a 132-MHz PowerPC 604 processor, while the Workgroup Server 7250/120 is based on a 120-MHz PowerPC 601 processor. All three run Macintosh System 7.5.3, which includes Open Transport 1.1, and feature built-in Ethernet and LocalTalk capabilities with support for AppleTalk and TCP/IP protocols.

The new Workgroup Servers feature improved performance over their predecessors in the product line. Said Jim Betlyon, president and CEO of Luminous Corporation (a recent spin-off

from Adobe Systems, Inc.), "In early benchmark tests, Color Central 3.0, our popular OPI image server software, showed a tenfold increase with a peak throughput of about 50 MB per minute on the Workgroup Server 8550/132, in contrast to a throughput of less than 5 MB per minute running Color Central version 2.5 on the prior Workgroup Server 9150/120."

The Workgroup Server 8550/132 is designed to meet the needs of publishing and other performance-demanding customers. It includes three PCI slots, two SCSI DMA buses for transfers of up to 10 MB/second, one 7200-RPM 2 GB hard disk with bracket and cable for a second drive, a quadruple-speed CD-ROM drive, 512 KB level-2 cache, 24 MB of memory expandable to 512 MB, an optional 4-mm DDS-2 DAT drive (to 12 GB compressed data), and eight ports for a wide range of peripheral options. For customers who purchased the 68040-based Workgroup Server 80 or the PowerPC/NuBus-based Workgroup Server 8150, Apple plans to provide a logic board upgrade to the Workgroup Server 8550.

The Workgroup Server 7250/120 is targeted for small businesses and schools; it includes three PCI slots, one SCSI DMA bus to connect up to seven SCSI devices, a 1.2 GB hard disk, a quadruple-speed CD-ROM drive, 256 KB level-2 cache, 16 MB of memory expandable to 256 MB, and eight ports for a wide range of peripheral options. The Workgroup Server 7250/120 enclosure is designed for easy access, making it simple for customers to add memory, PCI cards, and even another hard disk drive.

The low-priced Workgroup Server 6150/66 comes bundled with the Apple Internet Server Solution (see "Apple Internet Server Solution 2.0" on this page)

to create and publish information on the Web, and features a 66-MHz PowerPC 601 processor, a 1.2 GB hard drive, a quadruple-speed CD-ROM drive, 256K level-2 cache, and 16 MB of memory expandable to 72 MB.

Included with each Workgroup Server is a one-year on-site hardware warranty with lifetime hardware phone support and 90-day software phone support. Apple also offers fee-based direct support options. The Workgroup Servers are already shipping worldwide: Depending on configuration and which of the three software bundles the customer selects (see below under "Workgroup Server Software Bundles"), prices for the Workgroup Server 8550/132 range from U.S. \$6,449 to \$7,999, while the Workgroup Server 7250/120 costs from U.S. \$2,889 to \$3,719. The Workgroup Server 6150/66, which ships only with the Apple Internet Server Solution, is priced at U.S. \$2,299. (Prices outside the United States may vary.)

Workgroup Server Software Bundles

Three software solution packages are available for the Workgroup Server 8550/132 and 7250/120 systems, whereas the Workgroup Server 6150/66 ships only with the Apple Internet Server Solution. The first of these three, the Application Server Solution, is designed primarily to provide e-mail or database services. It includes the following software:

- FileWave from Wave Research for automated file/data distribution
- Now Contact and Now Up•to•Date from Now Software for contact management and workgroup calendar capabilities
- Viper Instant Access, a fast data lookup utility from IT Design
- 4-Sight Fax from 4-Sight L.C. for network-based send and receive fax transmission

- netOctopus and Virex from Datawatch for cross-platform asset management and computer virus protection

- Skyline/Satellite from the AG Group for monitoring network traffic

The second software bundle, called the AppleShare Server Solution, contains AppleShare 4.2.1 file and print server software. This version incorporates a PowerPC native file server engine and native XTI interfaces of Open Transport 1.1 to increase server performance and to support up to four simultaneous network segments, including 100BaseTX. AppleShare 4.2.1 also supports up to 3000 open files and up to 250 concurrently logged-on users, and runs at twice the speed of AppleShare 4.1. The AppleShare Server Solution bundle also includes the following software:

- Server Manager from Santorini Consulting for remote administration of an AppleShare server
- AppleShare Client for Windows, which allows Windows-based PCs to share files and printers on AppleTalk-based networks
- Apple Remote Access Multiport Server Software for remote connections to the server
- Retrospect Remote from Dantz Development for backing up data on the server and networked clients (DAT configurations only)
- the Workgroup Server Solutions CD version 2

Apple Internet Server Solution 2.0

The third software bundle for the Workgroup Servers is Apple Internet Server Solution 2.0. It's specifically designed to allow Webmasters to easily and cost-effectively publish HTML, graphics, audio, video, and database information on the Web. The software pro-

vides a cross-platform solution that allows access by all World Wide Web clients, whether they use the Mac OS, Windows, UNIX, or other system software. This solution includes a wide range of software, including the following:

- StarNine's WebSTAR 1.2.5, which turns a Workgroup Server into a powerful Web server; this version capitalizes on Open Transport 1.1 to provide faster Web server performance

- Adobe's PageMill 1.0, a ground-breaking application that allows users to create sophisticated Web pages without learning HTML

- Progressive Networks' RealAudio Server 1.0, from which allows users to provide audio content at their Web sites

- EveryWare Development's Tango and Butler SQL, which allows Webmasters to create powerful Web database solutions and many other useful applications

- Maxum Development's Net-Cloak 2.0, which improves security, simplifies server development and maintenance, and allows Webmasters to control exactly how a page looks to an individual client

- Apple's MacDNS domain name server software for the Mac OS, which allows a Workgroup Server to be a primary domain name server and enables Webmasters to efficiently spread a large site's load among several Workgroup Servers, using round-robin load balancing

- Netscape Communications' Netscape 2.0, a leading Web browser that lets Webmasters view the content they develop

- Adobe's Acrobat Pro 2.1, which allows Webmasters to create or translate documents into portable document format (.pdf), a popular cross-platform Internet document standard

The Apple Internet Server Solution CD-ROM comes with a variety of sample Web pages

(home pages, catalog pages, and forms), sample databases, online documentation, and "how-to" tutorials to help Webmasters design their sites. *WebMaster Macintosh*, the bestselling Macintosh Web server paperback and CD-ROM by Jeff Evans and Bob LeVitus, is also included.

PC World Gives Apple Top Spot in Two Studies

Although *PC World* magazine's main focus is the Wintel world (IBM-compatible computers running Microsoft Windows software), when it does industry-wide comparisons, it includes Apple and the Mac OS. So it's no faint praise that *PC World*, in both its February and March 1996 issues, gave Apple Computer, Inc., the top spot in two separate studies conducted by the magazine.

On page 145 of its February 1996 issue, *PC World* rated the Mac OS as the best 32-bit operating system—over Windows 95, Windows NT, and IBM OS/2 Warp. Similarly, in the March 1996 article "Homeward Bound or Bound for Trouble?", the magazine—for the third straight year—gave Apple Computer top marks for reliability and service.

Mac OS Best 32-Bit Operating System

For its February issue, *PC World* conducted tests that asked eight "typical users" to do a variety of common tasks, then rate the operating systems they used on a scale of 1 to 5. The rating categories used were disk and file management, desktop customization, hardware and driver installation and use, and on-screen help. The composite scores were the following:

Macintosh System 7.5.2	3.6
Windows 95	3.4
Windows NT Workstation 3.5	12.9
OS/2 Warp 3.0	2.8

Apple's biggest win was in the on-screen help category, where it bested Windows 95 by a score of 4.6 to 3.4. According to the article, Apple Guide "received particularly glowing reviews from most of our testers."

In its overall recommendations, *PC World* said, "Even with the long strides Windows 95 has made in usability, our testers gave the Mac OS 7.5.2 a slight edge in overall usability. The Mac also aced our graphics speed tests, one of many strengths prized by artistic users. These virtues aren't so compelling that you should scrap your PC, but they're reason enough to consider a Mac (or Mac clone) when you shop for a computer." (*PC World* went on to say that the 100-MHz Power Macintosh 7500 completed the assigned graphics tasks "in well under half the time of any other 32-bit OS.")

Apple Best Overall in Reliability and Service

The March issue's article "Homeward Bound or Bound for Trouble?" gives results of a detailed consumer report on PCs based on the experiences of 17,800 readers. The survey measured readers' experiences with 19 different PC companies in the areas of out-of-box problems, "later-life" problems, and length of time it took to reach customer service and support staff.

Respondents also reported on how long it took a manufacturer to resolve a problem, whether it was resolved, and whether they would purchase products from the same company again based on the service and support they received. The survey also measured customer satisfaction.

Each of the 19 companies surveyed was rated in nine key

reliability and service categories. *PC World* readers judged Apple's support the best in the industry, giving Apple's customer support staff the highest possible ratings in technical knowledge, courtesy, and follow-up. The company also received "best" ratings in the survey's three product reliability categories. The magazine noted in the article, "Apple: Excellent reliability backed by top-notch service. Gotta love them Macs."

Responses for the March report were gathered between May and November 1995. The *PC World* Reliability and Service survey is conducted twice a year, and is the largest survey of its kind on PC reliability and service, according to the magazine.

The *PC World* rating is just the latest in a series of recent top awards for Apple in the area of customer service and reliability. The Apple Assistance Center recently received the International Benchmarking Clearinghouse's Best Practices Award. Additionally, Apple received the top ranking for quality and reliability from the *Home PC* magazine survey published in November 1995 and the best overall ranking in J.D. Power and Associates' 1995 *Desktop Personal Computer End User Satisfaction* study.

MessagePad 130 With Newton 2.0 and Newton Press

The MessagePad 130 recently became the newest member of Apple's Newton-based line of personal digital assistants (PDAs). In addition, Apple Computer, Inc., shipped Newton Press for Mac OS, software that makes it easy to convert documents from Mac OS-based com-

puters into a format that's easily viewed by Newton PDA users.

MessagePad 130

The MessagePad 130 device, based on the recently released Newton 2.0 operating system, offers a new, more durable, non-glare screen with user-controllable backlighting for easier viewing in a broader range of lighting conditions. Newton developers will have access to the backlight APIs (application programming interfaces) so they can build backlight controls into their software.

The MessagePad 130 also provides additional memory. It ships with an additional 512K of system memory, bringing the total to 1 MB of system memory, which will allow for better performance with communications solutions such as TCP/IP and wireless LANs. The addition of system memory will also allow for better performance with multitasking support. The system heap is also expanded to utilize the extra memory.

The MessagePad 130, thanks to the Newton 2.0 operating system, provides improved handwriting recognition over earlier systems, allowing for easier data entry; it recognizes printing, cur-sive, and a combination of both. It has a built-in notepad, to-do list, datebook, telephone log, and address file for organizing personal and business affairs, as well as Pocket Quicken (in the United States only) to help organize personal and business expenses. The MessagePad 130, now available worldwide, is priced at U.S. \$799. (Prices vary in other locations.)

Newton Press

Newton Press is an application that lets computer users easily convert documents for viewing on Newton PDAs. It's currently available only for Mac OS systems; a prerelease version of Newton Press for Windows is expected to

be available free of charge online in the first half of 1996.

Users can create electronic reference documents or Newton "books" from Mac OS—based files by dropping files onto the Newton Press icon on the Mac OS computer. This automatically formats the document so it can be viewed on a Newton PDA. Newton Press supports a variety of word-processing, text, and graphics documents supported by the Claris XTND technology for the Mac OS. With Newton Press, word-processing files, e-mail messages, graphics, and text blocks from Mac OS computers can be made available to reference, annotate, fax, or print from the Newton PDA.

OpenDoc

Round-Up

OpenDoc continued to gather momentum in February: Claris announced that the new version of ClarisWorks due later this year will provide OpenDoc container support, and IBM released the beta version of OpenDoc for AIX. A variety of new developers also announced their support for OpenDoc, bringing to 350 the number of developers who have said they'll deliver OpenDoc-based products.

ClarisWorks: An OpenDoc Container

Claris announced that future releases of ClarisWorks, its popular all-in-one software product, will allow users and developers to create customized solutions by embedding OpenDoc parts within its word-processing, drawing, spreadsheet, and database documents; it will also provide a solutions platform for developers. Claris said that

ClarisWorks will provide seamless, transparent file exchange across OpenDoc and Mac OS environments (running on Power Macintosh and 680x0 systems) and Windows 95 environments.

Users will be able to add OpenDoc parts—from a wide variety of developers—to ClarisWorks documents, thereby adding new features to customize their ClarisWorks-based solutions. Parts will be integrated as frames in ClarisWorks documents by means of Macintosh Drag and Drop or an Insert menu command. As frames created with OpenDoc parts are activated, ClarisWorks will "arbitrate" its user interface, displaying the appropriate menu commands and tool palettes for that part.

For example, users might combine ClarisWorks with parts from other developers to add Internet browsing, intelligent mapping, animation, or 3D rendering features to their documents. ClarisWorks users can also look forward to a new generation of Internet-enabled solutions. Parts such as Cyberdog from Apple Computer will allow users to integrate updatable World Wide Web content in ClarisWorks documents.

This announcement gives you just one more business reason to develop OpenDoc software. Because of ClarisWorks' popularity—Claris expects to sell well over 2 million units this year—ClarisWorks OpenDoc container support will provide a broad base of customers able to integrate your OpenDoc parts into their own personalized software solutions.

"ClarisWorks' strong market position will provide a broad, familiar foundation that will become richly extensible for OpenDoc parts developers and customers alike," said Pieter Hartsook, editor of *The Hartsook Letter*; an Internet-based Macintosh market research publication. "This is a great example of how

leading Macintosh developers should use OpenDoc to help their customers reap real benefits from component software."

OpenDoc for AIX Beta

At UniForum, the large UNIX trade show, IBM announced the availability of OpenDoc for AIX (beta version), the first UNIX implementation of OpenDoc. OpenDoc now supports three platforms: OpenDoc for Mac OS began shipping in January, and IBM recently made available OpenDoc for OS/2 Warp Version 3.0. IBM expects to deliver OpenDoc for Windows 95 and Windows NT (beta) later this year.

The beta code of OpenDoc for AIX is available free of charge at the Club OpenDoc Web site. (For the location, see "Internet Resources for this Issue" on page 36.) Club OpenDoc, introduced last November, is a new location on the Web for information about OpenDoc technology, especially IBM's OpenDoc products. ♣

Technology

CD Highlights

System Software/SDK Edition, April 1996

This month we introduce Cary Henrie, our new artist for the System Software and SDK CDs. She's responsible for that killer cover art, but not for the small fib on the back of the SSW disc: This month's new version of Open Transport is 1.1b16, not 1.1b11c3. And, finally, we have a big pile of new snippets from Pete Gontier and other DTS engineers.

Here's some of the other new and revised stuff on this month's disc.

Apple Grayscale Appearance

The Apple Grayscale Appearance for System 7.5 specification provides the information you need to create a grayscale appearance for System 7.5 products. It consists of an Adobe™ Acrobat document and two folders of enlarged SimpleText graphics showing the details of human interface elements.

Developer Notes Update 04/96

The *AppleVision 1710av* and *1710 Displays Developer Note* describes design features of the new AppleVision 1710av display, which has audio/video capabilities, and of the AppleVision 1710, a similar display with video capabilities only.

Int'l Developer Contacts 2/96

This folder contains a list of international developer programs and services contacts, in Acrobat format.

Open Transport 1.1b16

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 1.1b16 Installer and Debug Installer, as well as the Software Development Kit.

OpenDoc 1.0.1

OpenDoc is an open industry standard architecture for building reusable, distributed, cross-platform software. It's both a



System Software Edition

specification and a set of technologies for building components. If you build your applications in accordance with the OpenDoc architecture, you can provide software that integrates and cooperates with other OpenDoc software.

With OpenDoc, you can provide specialized solutions to meet each customer's specific need. OpenDoc enables you to build more powerful, higher-quality software in less time than it takes to develop traditional monolithic applications.

This folder contains OpenDoc 1.0.1, documentation, development essentials, and sample code.

Inside This Section

Human Interface: Use Your Illusion	16
Client/Server Computing: An Overview for Information Systems Developers	18

OpenDoc Development Framework

This folder contains the 1.0d11 release of the OpenDoc Development Framework (ODF), formerly known as the OpenDoc Parts Framework (OPF). ODF is a C++ object-oriented software framework that speeds the development of OpenDoc parts for both the Mac OS and Windows platforms.

This release supports OpenDoc Developer Release 4 (DR4) for Macintosh only. If you want to do Windows part development, use the ODF 1.0d8 release, which supports both OpenDoc DR1 for Windows and OpenDoc DR2 for Macintosh. The OpenDoc API has changed so much since DR1 that it has become impractical for Apple to continue to support both APIs in the same code base. In d11 the following development environments are supported: for PowerPC processor-based systems, Metrowerks CodeWarrior 7.0 and Symantec C++ for Power Macintosh 8.0 (Rainbow); and for 680x0-based systems, Symantec C++ for MPW. For instructions on building parts and the framework, see the specific Build Guide documents located in the Build System/Installation folder. For further details, see the document ODF D11 Read Me.

OpenDoc International

OpenDoc J-1.0.1 is a localized version of OpenDoc for Japan. It supports KanjiTalk 7.1.1 or later and runs on a Macintosh computer with a Motorola 68030 or 68040 processor or a Power Macintosh computer with a PowerPC 601, 603, or 604 processor. Included

please turn to page 22

Human Interface

Use Your Illusion

By Peter Bickford

There's an old proverb for columnists and public speakers that goes, "Tell 'em what you're gonna tell 'em. Tell 'em. Then tell 'em what you told 'em." In the spirit of this advice, I'm going to attempt to explain such things as what receptionist services have in common with compiler optimization settings and why a bulk rate stamp or an errant filename can convince your customers that you're a pathetic liar.

If all of this is sounding a bit like a puzzle the Riddler might have left Batman in the last Batman movie, I'll give you a clue: It's all about the power and perils of illusion.

Jumping to Conclusions

There's a small businessman I know who, like most small businessmen, dreamed of one day becoming a big businessman. He was also well aware that although people might like his product or service, they would often be reluctant to trust their faith and money to someone whose office consisted of a desk and computer shoved into a corner of his bedroom. It seemed that if he wanted to have much hope of becoming a big businessman someday, it was important to convince his customers that he already was one.

Accordingly, he hired a receptionist service. Through the miracles of modern phone technology, he was able to forward his calls to their office all the way in Poughkeepsie any time he stepped out. Then, when callers rang up, a bored-sounding receptionist would drone on about how Mr. Smalltimer was unavailable, that they didn't know when he'd be back, and could they please take a message. In short, it was just like a real business.

But one day, Mr. Smalltimer returned to his home office to find a stack of messages from an irate customer. The customer had called and asked to speak with him about an urgent technical matter. But when he got the standard "Mr. Smalltimer is not available, can we take a message?" the caller pressed on. "Do you know when he'll be back?" "Can you check his calendar?" "Do you have his direct extension?" "Did you see him come in today?" and finally, "What sort of idiots are you?" For as it turned out, Mr. Smalltimer's illusion of having a front desk receptionist worked just a little too well. Believing that there really was a receptionist in the office, he asked for things a real receptionist should have known. And when the receptionist didn't give him the information, the caller came to the very rational conclusion that the receptionist was just withholding the information to annoy him. Scratch one customer for Mr. Smalltimer.

One of the most amazing powers of the human mind is its ability to fill in the gaps where information is missing. Give us half the words in a sentence and we can usually still figure out its meaning. Show us part of a picture and we can fill in the rest.

The really fascinating part is that, given partial information about a situation, we'll make up our own harebrained scenario that not only fills in the missing pieces but also explains the situation as a whole.

For example, if I install a new system extension and my computer crashes half an hour later, my first thought is to blame the new extension for being incompatible. If I set my compiler to "Method Q Optimization" and it takes much longer to compile, I just assume it's because Method Q is slavishly analyzing my program in great detail, and probably doing just a bang-up job of optimizing the code. If it turns out that the code runs slower than before, I'll make up yet another theory to explain that ("Well, you know, Method Q is probably best on *large* programs. . .").

When we deliberately show people just enough information to cause them to jump to the conclusion we want, we're engaging in the art of illusion. Whether he realized it or not at the time, our Mr. Smalltimer became an illusionist when he hired his receptionist service. The rest of us are practicing the art of illusion when we write miles of C++ code to convince users that they're piloting a plane in a flight simulator program—or that their computer has a tiny built-in trash can wherein they can dispose of the various "files" and "folders" that are somehow squished into their computer's hard drive.

Human interface design is all about crafting the user *illusion*. This is where we hide the underlying code and complexity of the system, and instead present the user with a much simpler picture of the way the system works.

The Power of Illusion

A good, simple illusion is usually preferable to a complicated reality. I once heard a user talking about how great the Macintosh computer was, saying that "on the Mac, you just put things in file folders—in DOS you have to mess with all these directories and pathnames and stuff." In reality, the two file systems use virtually identical schemes for storing data. The difference is that the Macintosh provides the user with a better illusion of how it all works.

If our program needs to control a relative sound volume level, we *could* let users type in a number for the sound amplification setting—after all, we know that it's just a way to get a value that we can pass along to the sound hardware. In most cases, however, we'll let users set the volume with some sort of slider control. This is really just a graphic that moves—unless we write a reasonable amount of code to give it a function. To users, however, it feels the same as the one on a stereo system that directly controls the electrical flow to the amplifier. They believe our illusion. What's more, users will feel more comfortable with our "volume slider" model than they would with a system demanding that they think about

the underlying engineering model of amplification values and Sound Manager calls.

The master practitioners can pull off illusions on a much grander scale, such as when they let users establish communications and access information on faraway mainframes just by clicking an “inbox” icon on their desktop. When the illusion works, users feel more in control of the system, and become far more productive. They think about “getting mail” instead of tweaking a computer’s serial port settings.

I’d go so far as to argue that the entire Macintosh interface is designed as a grand illusion that shields users as much as possible from having to work the computer, and lets them concentrate on getting work done with a computer. Of course, some parts of this illusion are more polished than others.

Blowing the Gaff

When a con man pulls a scam, he’s setting up an illusion for his personal gain. The setup—the illusion—is called a “gaff.” And if he ever slips up and does something to let his mark know that he’s being fooled, it’s called “blowing the gaff.”

The next time an apparently urgent letter arrives for you in the mail, covered with official markings and labeled “Priority Message,” take a look at the postage stamp before opening it. If this “priority message” bears a bulk rate seal, you’d be better off throwing it in the garbage. Most likely, it’s a sleazy solicitation of some sort. They blew the gaff when their priority message didn’t bear first class postage.

One Macintosh networking program I know had the goal of updating itself automatically whenever a newer version became available. Early on, the programmer realized that he wouldn’t actually be able to update the code of a running program, but he still wanted create that effect for the user. So, while the user stared at an “updating to new version” dialog box, his program was downloading the new copy, switching over to it, then deleting the old copy while renaming the new copy to have the same name as the old one. It was all quite impressive, except that users were able to see the scratch

files being created in the Finder while the process was going on. Those who caught on not only accused the author of lying to them about the process, but also blamed him for any unexplainable file they happened to find on their hard drives.

A good illusion inspires confidence in a system, but a bad one feels like a lie. When you set up an illusion, make sure you’ve done the work to support it. Pay very close attention to the details so that an errant redraw or a slight inconsistency doesn’t give you away as a fake. Moreover, make sure you can reasonably meet the expectations your illusion sets up in the minds of users. Had Mr. Smalltimer been a bit more careful in that regard, he would have had one more customer.

Human interface design is a process of creating a user illusion that works. If it fails often, users begin to second-guess our interfaces and demand lots of information about the underlying system. Their anxiety levels go up and their productivity levels go down as it becomes clear that underneath all the pretty icons lurks something big, scary, and mysterious. This may remind you of certain operating systems, but you’d do well to make sure it doesn’t remind anyone of your application.

Till next time,

Doc

the.doktor@applelink.apple.com

Developers! Curious about Copland? Distraught over a design issue? You can now write to machi@apple.com to get your interface design questions answered by Apple’s top human interface engineers.

Also, go to <http://dev.info.apple.com/developerservices.html> for information on the first annual Human Interface Design Excellence Awards, to be held at this year’s Worldwide Developers Conference. The deadline for entries is April 12, so be sure to download the contest information soon. Winners for the Innovation, Look & Feel, Elegance, and Overall categories will be able to proclaim themselves the Best Interfaces of 1995.

develop

develop

The Apple Technical Journal

develop contains articles, columns, and Q&As that will help reduce your development time. This quarterly journal gives you an in-depth look at code and techniques that have been reviewed for robustness by Apple engineers. The Bookmark CD that comes with it contains the source code for that issue, all back issues of *develop*, Technical Notes, and more. Subscribe now!

800-877-5548 U.S. 815-734-1116 Outside the U.S.

Client/Server Computing: An Overview for Information Systems Developers

Based on Apple's Guide to Client/Server Development Tools and Technologies

Effectively meeting user needs is the major challenge facing corporate information systems groups today. Unprecedented change is transforming the business world, and information systems (IS) productivity is no longer gauged solely by the volume of code generated or by budget performance. IS departments are now measured by added value—that is, how well they convert data to competitive advantage for their corporations.

Apple Computer, Inc., has launched a series of initiatives intended to assure that the Mac OS platform will be a responsible client to mainstream client/server environments typically found in corporate enterprise networks. As a result, information management professionals now have a broad array of Macintosh-specific solutions available for application development, database access, and network connectivity, as well as numerous client/server solutions supporting sales automation, human resources management, and a wide variety of other major business processes. The superior ease of use and productivity available to Macintosh users are being combined with the client/server model to create a distinct competitive advantage for organizations.

For detailed information about Apple's approach to client/server IS development and tools you can use to develop client/server solutions, you can consult Apple's *Guide to Client/Server Development Tools and Technologies*. (This publication was recently

made available on the World Wide Web; see "Internet Resources for This Issue" on page 36 for the location.) What follows is a general overview of client/server computing, including a look at how OpenDoc, Apple's component software architecture, and object-oriented programming will bring a new level of sophistication to IS development. This article is based on the first two sections of the *Guide to Client/Server Development Tools and Technologies*.

Cooperative Processing: The Key to Virtual Integration

Many new computing paradigms have come and gone in the information systems industry. Although many different application design methodologies exist today, most of them have one thing in common: They use some form of distributed, cooperative processing between a user's desktop computer and one or more servers that sit in a back office. The Macintosh computer provides a robust client to these network-based applications.

Cooperative processing distributes the computer logic and processing load to the most appropriate computers on the network. The user sees one consistent, "virtually integrated" application and human interface even though a portion of the application is executing on the desktop computer and a number of servers simultaneously. Partitioning and logic control are done by means of messages exchanged among the "cooperating" applications.

The increasing power of personal computers and improvements in communications

technology make it possible to distribute the computing workload between the personal computer, the host mainframe computer, and local server systems. Cooperative processing leverages the power of personal computers and provides a modular, flexible, scalable, and cost-effective approach to multi-user applications. Such cooperation provides the productivity, ease of use, and human interface intimacy commonly associated with personal computer applications as well as the processing power normally associated with mainframe systems.

Typically, cooperative processing applications are partitioned into component parts. These components can be distributed onto computer platforms that are optimized for each specific purpose. This approach can provide four key benefits:

- Ease of use, productivity, and quality improve when the advantages of personal computers—powerful built-in human interface capabilities and programming tools—are combined with the robust "back-end" computing environment associated with enterprise-quality servers.
- Data access is improved through graphic and multimedia user interfaces and data integration.
- Overall completion (turn-around) time of business tasks is reduced. Because various portions of the application operate concurrently on separate systems, the turn-around time is only as long as the longest single task rather than the cumulative time of all the tasks.
- Overall costs per business task are lower, and adaptability to

changing system workload requirements increases: Due to adroit design and configuration management, each system component handles the part of the processing that it's best suited for.

The rest of this article provides some background on cooperative processing approaches, focusing on client/server computing models and client/server development scenarios. First off, a word on the client/server model for computing, and why it's become so prevalent.

Organizational Climate Demands Client/Server Model

Every large company has a diverse range of computers in locations around the globe. Companies are acquired and sold, centralized and decentralized; entire divisions, after many years of independent operation, may join the enterprise, bringing with them their own established computing environments. Such dynamics create a complex and diverse IS infrastructure.

Additionally, every group in the enterprise is compelled to do more with less. Leaner IS organizations are expected to create new systems in less time, with fewer resources. The percentage of time that IS professionals spend maintaining existing systems has become too high to meet new business demands. Increasingly, IS developers have difficulty meeting the rapidly changing needs of their businesses.

Enterprise computing has evolved from two distinct paths that are now converging: host-centered mainframe processing

and desktop-centered personal computers. In the 1950s and 1960s, most systems existed in a batch environment; a user had no direct interaction with the system. The 1970s and 1980s saw the evolution of systems designed to support online video display terminals, which led to extensive changes in systems structure, application design, and systems management.

Personal computers evolved independently and concurrently, starting as stand-alone systems used to improve individual productivity. Then, they were joined together in wide area networks along with “dumb” terminals to allow the personal productivity computer to replace the redundant terminal for host-centered mainstream systems. Personal computers were connected to local area networks to improve workgroup productivity, primarily for electronic mail and for print and file sharing.

Today, the host and personal computer paths have merged, creating a complex multiplatform environment. The challenge is to integrate this diverse collection of computing platforms and communications technologies into a cohesive, manageable enterprise system. The most appropriate in-house development strategy is (usually) an architectural approach that preserves the investment in legacy systems while allowing evolution from legacy systems to new systems.

Virtual integration is an emerging concept in enterprise systems design. It means that the user “sees” existing diverse applications and hosts through a single common interface. A virtually integrated system has two key advantages: First, virtual integration lets IS staff continually improve existing systems without seriously disrupting operations. Second, to the end-user, the information system appears to be an

integrated whole, even when it is actually composed of incompatible applications and data that reside on a variety of platforms and networks.

Using virtual integration, corporate IS departments can deliver systems that benefit the whole enterprise by

- exploiting existing systems, applications, data, and networks
- providing scalable systems
- minimizing system development and deployment time
- reducing training time for IS staff and users
- reducing maintenance and support efforts
- providing distributable application services
- increasing users’ overall effectiveness

If you develop user-centered applications with graphical user interfaces, you further empower end-users by providing them with a consistent, easy-to-use interface for their business systems. Because users can concentrate on the business problem at hand, the IS system becomes a competitive tool, rather than an obstacle.

Graphical user interfaces and virtual integration are significant steps toward the ideal IS world. When you combine them with network services to local servers and large-capacity host systems, you can create a powerful enterprise computing environment—one that’s capable of efficiently adjusting to the rapidly changing demands of today’s business environment.

Client/Server Processing Defined

Because of the changing business and computing climate, client/server processing has become the most prevalent form of cooperative processing. Client/server applications consist of a mainline control program that manages a portfolio of electronic requests

delivered as “invocation messages.” Cooperating programs called *services* receive requests and perform tasks as specified in the requesting messages. The platform that issues the request is called the *client*, and the platform that accepts requests and performs the requested services is called the *server*.

Client/server processing is thus a set of programmed and prescribed cooperative processing roles. It is not a hardware platform concept; rather, client/server processing is a state of transaction pairing. Each request is a message with control parameters and transaction data, and is delivered to the target server. The server accepts the request, interprets the preferences contained in the message, and operates on the transaction data contained in it. The server can itself act as a client when it sends requests to other cooperating programs running on other computer systems while it processes the original request.

Four Cooperative Processing Models

Cooperative processing can be categorized into four computing models:

- host-driven client/server processing
- desktop-driven client/server processing
- collaborative processing: peer-to-peer logic sharing
- distributed object computing (DOC)

These models are listed in order of increasing sophistication and increasing independence from a “master” application. That is, a true distributed object is far more self-contained and operationally independent than a host-driven client/server application. The models can, of course, be mixed. A virtually integrated application may actually use a

combination of models to mask the apparent complexity of the underlying legacy and independent systems.

Host-Driven Client/Server

In a host-driven environment, the host application contains the main control logic and manages the network session. Under the host-driven client/server model, also known as the *remote presentation model*, the host computer receives user input from the personal computer and determines how the application responds to that input. The host computer is then free to perform centralized business tasks, which are better suited to its design. The personal computer’s role is little more than that of a sophisticated terminal.

In the host-driven client/server scenario, the personal computer runs an application that turns the computer into a server capable of receiving well-defined messages from the host. The host sends messages to the personal computer, which interprets the messages and performs specific tasks, such as displaying a menu, window, or dialog box. In addition, the personal computer may send information back to the host, usually at the prompting of the host or a user response.

In advanced host-driven client/server applications, the host can alternatively download “applets” or request that the personal computer perform a routine that’s part of an application residing on the personal computer. In such situations, the server computer is actually acting as the “client,” and the desktop computer is performing the role of “server.”

Host-driven client/server computing reduces part of the host’s processing load—some of the processing takes place on the personal computer—but it requires changes to the legacy application’s human interface

code. Because the host program must be aware of and manage the interaction with the cooperating desktop application, host-driven client/server computing is considered “intrusive” to legacy systems, and is therefore not recommended for legacy code that can’t be maintained. However, if your legacy code is easy to maintain, this client/server model is the one that’s easiest to implement and support, because it requires limited skills in developing for personal computers.

Desktop-Driven Client/Server

The desktop-driven client/server model allows for more efficient use of the graphic interface code and processing power resident on the personal computer. A larger portion of the application function moves from the host to the desktop computer, significantly reducing the processing load on the host system. In the desktop-driven approach, the personal computer has dominant control of the application’s behavior, including what it displays and when and how it receives input from the user.

Borrowing terminology originated by the Gartner Group, an electronics industry consulting firm, the desktop-driven client/server model can use a “fat client” (also called *remote data access*) or “thin client” (also called *distributed function*) approach. The “fat client” presumes that the bulk of the application and control logic resides on the desktop computer, and that volatile data validation, transaction contention management, and database access/update take place on the host computer as requested by the desktop application. Generally speaking, the “thin client” still presumes that the application control logic remains on the desktop computer, while network session management and most of the

application business logic reside on the host/server.

The desktop-driven client/server approach is the ideal model for applications that involve significant human interaction, data filtering, and analysis incorporating graphics, video, and sound. You might consider the “fat client” style for end-users in your organization who typically make unpredictable, irregular use of the system—for example, users in research and scientific undertakings—and for creators of intellectual capital. The “thin client” style is preferable for end-users who make predictable, repetitive use of the system—for example, if many individuals ask for frequent, even simultaneous updates of the same data or perform the same tasks over and over again. If you use the “thin client” approach, you can also administer software distribution and version control more easily, because you can distribute relatively few copies of the client to selective site servers rather than having a great many copies sent directly to every desktop.

Collaborative Processing: Peer-to-Peer Logic Sharing

Collaborative processing is a highly flexible message-based computing style in which personal computers and host computers become functional peers. In this advanced form of cooperative processing, software programs interact with one another independently and transparently to both the application and the user.

In collaborative processing the roles of the personal computer and the host shift; instead of having the static roles of client and server, the cooperating platforms are allocated tasks dynamically based on a set of priority rules and on their individual processing capacities. In response to an event trigger, services determine the availability of computers on

the network and the priority of the event, and allocate tasks among the available computing resources.

Under the collaborative processing paradigm, software services may reside on a single platform or on different computers communicating over a local area network (LAN). In either case, the services operate identically. The services may be requested synchronously or asynchronously, and they may operate in a serial or parallel manner to complete a collaborative task. Various hardware platforms can be used, each acting as a “node” in the network to the others.

Cooperative processing systems can be scaled incrementally, from small trial systems to huge, enterprise-wide organizations. Services can be moved around, subdivided, and replicated—all transparently to the clients. Cooperative processing is especially well suited to the development of vertical market and custom cooperative systems.

Collaborative peer-to-peer integration dramatically improves processing quality because it allocates computing tasks among available systems on the network more appropriately than other models do. It also results in flexible, highly optimized IS solutions that don’t depend on a single platform for their success.

Distributed Object Computing

As client/server architectures mature, new and more reliable methods are being invented to enable richer functionality. As significant computing tasks become more widely distributed throughout an enterprise, the underlying computing architecture must keep pace. The next frontier in client/server computing is termed *distributed objects*. The distributed objects architecture will be based on component

software and more sophisticated methods for requesting, reusing, and integrating components in custom solutions.

Distributed objects have the same collaborative processing characteristics as their predecessors. Distributed objects, however, are more self-contained and independent. Objects encapsulate the metadata (information about information), the methods that can operate upon that data, and the triggers and messages that may be sent to and from other objects. Objects are grouped into “object frameworks” that perform major business tasks with a single originating invocation command. The component objects may be distributed and replicated on as many platforms in the network as necessary to assure that they’re available when needed, and to assure that they can complete tasks in an optimal fashion.

The emerging standard for distributed object computing (DOC) is called CORBA (Common Object Request Broker Architecture), which is being drawn up by the Object Management Group (OMG), a large association made up of the industry’s major software and operating system vendors. Many vendors—including IBM, Sun, and Hewlett-Packard—now offer CORBA-compliant products. Besides the communications backbone and interface definition language of CORBA, OMG is creating specifications for general services that should be available in any DOC system. (OMG will also extend CORBA to include a compound document model, and CI Labs has submitted OpenDoc for consideration.) CORBA will be a widely accepted standard for a language-neutral and platform-neutral environment that provides a rich collection of tools for the creation and management of distributed applications. You can find more information about OMG and

CORBA at the OMG Web site. (For the location, see "Internet Resources for This Issue" on page 36.)

Component Software: The OpenDoc Standard

For those of you not already familiar with it, OpenDoc is Apple's cross-platform software architecture that enables applications to be replaced by a number of smaller components, each performing a specific set of tasks. OpenDoc components can interoperate on a single platform, or they can be executed on remote platforms in the network by means of collaborative processing and distributed object models. *Component software* is a term used to describe reusable, compatible building blocks of code that work seamlessly with applications—called *part editors*—from different vendors.

The result is more efficient software development, more freedom to mix and match part editors from a variety of vendors, and more time to innovate. Users get the ability to work in a task-centered (rather than application-centered) manner. OpenDoc will make it easier for users to create and edit content-rich documents.

OpenDoc consists of five functional layers:

- compound document services
- component services
- automation services
- object management services
- interoperability services

A technology called ComponentGlue being developed by IBM provides interoperability between OpenDoc and OLE 2.0 objects. So, you shouldn't view OLE 2.0 and OpenDoc support as mutually exclusive. In fact, supporting OpenDoc is the most efficient way to make applications OLE 2.0-compliant, because, in the experience of many developers,

OpenDoc provides an easier way of developing component software. Also, supporting OpenDoc makes it easier to develop for multiple platforms, since OpenDoc, already available for the Mac OS and OS/2, will also be available for Windows and UNIX.

OpenDoc Enables DOC Model in Large-Scale Systems

OpenDoc enables the rapid development and evolution of network software systems, using

OpenDoc enables the rapid development and evolution of network software systems, using existing network infrastructures.

existing network infrastructures. Just as OpenDoc simplifies development of software components by reducing the number of issues a programmer must deal with, distributed objects simplify client/server software development by allowing clients to treat servers as local objects with an application-specific interface. This means that programmers are free of the task of coding to network and "middleware" APIs, and can concentrate on higher-level application issues.

Large organizations need to integrate the wide variety of computing resources now found at their sites. Personal computers, workgroup systems, legacy servers, and new services must be seamlessly integrated for organizations to effectively use the data and logic wrapped up in those systems. OpenDoc clients that provide access to new and existing services are an excellent

solution because they are easy to develop and are readily integrated with each other and with productivity software.

Imagine, for example, a document showing live figures from a departmental sales database. The document is used to form a query for a company-wide personnel database that returns contact information about an employee from a file on the local workstation. Salary data from the personnel database is plugged into a spreadsheet part that the

should be made of standardized, reusable components that are assured to interoperate. Distributed object computing itself can most effectively be managed through object-oriented development techniques.

Object-Oriented Programming . . . Why Bother?

The traditional procedural development process focuses on functions that manipulate data, thereby reducing tasks to the language statement level. The statements are then coded and assembled into a complete system that performs to specifications defined at the beginning of the process.

Object-oriented development is a more abstract process that uses object-oriented programming (OOP) techniques to create a model of the work to be performed using the application. Each object in the model has a defined behavior—for example, a printer object may serve as an interface between a desktop system and a specific type of printer.

Objects created with OOP can be easily reused in other systems, allowing developers to create complex applications without being required to redevelop the detailed routines that make up the application. Existing routines are catalogued and represented by an object. OOP is based on the following key concepts:

- *Encapsulation.* Related items are grouped into a single conceptual capsule that includes both data and functions.
- *Hiding.* An object-oriented program reduces the visible complexity of encapsulated information by hiding portions of the data that needn't be seen.
- *Classes.* Similar objects are grouped based on their behavior specifications.
- *Inheritance.* Methods are defined and then shared among member objects and classes.

• *Polymorphism.* Two or more objects respond to a message from another object in unique ways.

To include an object's functions in the application under development, you can use OOP concepts to place the object in a process flow. The object brings its functions to the application transparently. Once the application is in the production environment, changing the object allows quick upgrades to many applications that use it.

An example of an object is the printer driver mentioned earlier. First, you create the printer object. Then, by adding the object to an application during the development process, you give the application the ability to send documents to that type of printer. If a new version of the printer requires updated software, simply updating the printer object allows all applications that use the object to send documents to the new printer.

Although not required for developing cooperative applications, OOP concepts improve the ability of the IS developer to meet the business needs of the user. OOP is also particularly well suited to the development needs of systems that employ graphical user interfaces.

Conclusion

We hope this overview of Apple's way of looking at client/server computing has given you a few ideas for your IS development

schemes. If you'd like more specific information about developing IS solutions that integrate the Macintosh computer into an in-house computing environment, we urge you to consult the *Guide to Client/Server Development Tools and Technologies*. The guide includes specifics about a wide variety of client/server development tools that let you integrate the Macintosh computer into virtually all of today's significant IS environments. ♣

CD Highlights

continued from page 15

in the OpenDoc International folder are OpenDoc files, Code Fragment Manager (CFM) files, extensions, and a control panel for OpenDoc J. Also included are recipes to support Text Services Manager (TSM) TextEdit and sharing a TSM document data structure among multiple parts in an OpenDoc document, a header file for the OpenDoc TSM Addition (which is a shared library installed with OpenDoc J), sample parts with TSM TextEdit support, other sample parts, and an OpenDoc glossary in Japanese.

OT Code Resource

This piece of sample code shows how to call Open Transport from a 680x0 code resource written in either Metrowerks CodeWarrior C or Symantec C for MPW. The sample creates a HyperCard XCMD that returns the default Ethernet address for the computer using Open Transport calls.

PlainTalk 1.4.1

PlainTalk is a collection of software that enables your Macintosh computer to speak written text and respond to spoken commands. PlainTalk 1.4.1 includes three components, each with its own installer: English Text-to-Speech, Mexican Spanish Text-to-Speech, and English Speech Recognition.

The document About PlainTalk 1.4.1 describes each of these software packages, and discusses the differences between PlainTalk 1.4.1 and earlier versions of PlainTalk.

This folder also contains version 1.5a4 of Apple's new Speech Recognition Manager. This is alpha software and documentation—not ready for wide distribution or release in products yet. When finished, version 1.5 of the Speech Recognition extension will embody the first officially supported developer release of the Speech Recognition Manager.

ScriptX 1.5

With version 1.5 of the Apple ScriptX Interactive Multimedia Platform, Apple is taking the first steps toward establishing a premier platform for the development and delivery of the next generation of multimedia applications and tools.

The platform consists of three elements: the ScriptX Media Player, the ScriptX Language and Class Library, and the ScriptX application development and authoring tools.

The ScriptX Language Kit version 1.5 implements the ScriptX 1.5 platform. It contains over 250 "core classes" for multimedia development. They include title management facilities for creating and managing ScriptX applications, composition facilities, graphics, timed behavior facilities for handling time-based streams and sequences, interaction facilities for creating user interface elements, data management facilities, and language support facilities.

See the document About ScriptX for details.

Worldwide System Software

This folder contains U.S. and Worldwide versions of System Software 7.5.2 for the Power Macintosh 7200, 7500, 8500, and 9500 computers, and for PowerBook models 190, 2300, and 5300.

In the More System Software folder on the Dev.CD Apr 96 SDK2 disc, you'll find Kanji versions of SSW 7.5.2 for PowerBook 190, 2300, and 5300 computers, and Greek and Hebrew versions of System 7.5.

Here's how to work with the material in the Worldwide System Software folder:

- To create floppy disks from the disk images, use the Disk Copy 4.2 or ShrinkWrap 1.4.2 application in the Utilities folder. To create network installation folders on your desktop, use the DropDisk 1.0b5 or ShrinkWrap 1.4.2 application.

- The installer program for System Software 7.5.2 must determine at run time the type of the computer it's installing the software for; therefore, you must run the installer on the appropriate computer, even if you're installing on a disk that is not the current startup disk.

- The "•common disks" folder contains disk images common to the PowerPC 7200/9500 and PowerPC 7500/8500 folders. Do not install from this folder (you can't anyway, as it contains no Install 1 disks); install from the images in the PowerPC 7200/9500 or PowerPC 7500/8500 folders.

Alex Dosher
Developer CD Leader

Business

Business Feature

Six Assumptions That Can Kill Your Multimedia Title

By Dee Kiamy,
Open Door Communications

During a recent dinner party, a friend presented a brain teaser he'd recently heard. It went something like this:

Romeo and Juliet lie dead on the living room floor. The carpet on which they lie is wet, and they are surrounded by broken glass. What happened?

We were allowed to ask as many "yes or no" questions as it took to solve the mystery. The point of this exercise was to stretch our thinking and explore alternative approaches to problem solving (not my usual fare with pasta and caesar salad, I assure you).

But loving a good challenge, we dove in. After 10 minutes of animated questioning (OK—15 minutes, really), we were all beginning to scratch our heads until someone's inner light bulb flashed. This person hit upon the breakthrough question: Are Romeo and Juliet humans?

The answer was *no*. After that, the rest of the puzzle pieces began falling quickly and neatly into place. The solution: Romeo and Juliet are goldfish whose bowl was inadvertently broken. Simple enough—once you shed the natural assumptions that Romeo and Juliet are the

Shakespeare characters and, further, that they are human.

Before that "breakthrough," what impeded our solving the puzzle were the natural assumptions we all tend to use as ground zero when faced with a challenge or problem. We automatically presumed that Romeo and Juliet were Shakespeare's ill-fated couple and started from there, not consciously aware that we'd taken this for granted.

Afterward, the more I thought about it, the more I realized that this was a valuable reminder. Especially in the fast-paced PC industry, we routinely (and very naturally) tend to make some assumptions so that we can quickly move forward to the business at hand: solve the problem, build the product, get it on the shelves in time for Christmas—whatever the business at hand is. Sometimes this approach works in our favor, and sometimes it doesn't.

In this evolving and highly volatile arena in which pioneers are breaking ground and, in many ways, inventing "how things are done," you can't take much for granted. Making the wrong assumptions—or being unaware of those you've inadvertently made—can be particularly deadly.

As the Romeo and Juliet puzzle demonstrated, making incorrect assumptions can, at the very

least, prolong or preempt a healthy problem-solving process; worst case, the right path may be totally elusive, because in business you usually don't have the luxury of getting the immediate "yes or no" answers to questions that could prevent you from heading down the wrong path.

However, by consciously recognizing and then setting assumptions aside, by challenging how you think about your products, markets, customers, and even yourself, and by exploring alternate approaches, you clear out the cobwebs and put yourself on the right track.

All in all, this dinner table experience made me wonder what assumptions I've been making while on the job, and how they might be affecting my work. Then I started thinking about which assumptions are the easiest, most common, and deadliest ones to make in the multimedia business. Which assumptions get in the way of successfully creating and marketing truly excellent multimedia titles? Here are a few that not only may get in your way, but also could kill your product.

Killer Assumption #1

Before buying a product (or even after), customers carefully read product packaging and understand the system requirements for running a title. How we all wish

that were true! This assumption can be a real killer. (I could cite many recent high-profile examples, but we all know them. . . .)

Given the incredible complexity of today's hardware and software configuration possibilities, it's all too easy to presume too much about the average customer's understanding of what it takes to install and run your title—this system, that driver, whose-it's sound card, and so on.

Customers become frustrated when they can't easily install and run a title, regardless of whether it's their fault for not reading the package before they bought it or your fault for being incomplete or inaccurate. Novice users often don't even realize that different products have different requirements. Often, people don't fully realize that buying a CD-ROM disc for a computer isn't like buying one for a stereo system—one size doesn't fit all—and that it isn't like buying the content in another medium, such as a book, because of the extra effort needed to use it.

Assume an overly optimistic level of user knowledge, and before you know it, your return rate is skyrocketing and your reputation plummeting. (Granted, this is less of a problem for Mac OS products, but it's a good reminder, which is why it's included here.)

Inside This Section

The Changing Landscape of Software Distribution 26

The first thing to do is to fling this assumption aside, and then use the appropriate amount of diligence in determining system requirements and communicating them to buyers. The burden is on *you* to thoroughly test your product to clearly determine *exactly* what system configuration is needed to successfully run your product. But you must also take great pains to communicate this crucial information clearly on the package. On most packages, requirements are in tiny type and aren't nearly prominent enough to accomplish what is needed. The easier they are to spot, the better.

Also, to simplify the installation process for users, consider building in such things as intelligent agents—"helpers" that coach users through the installation process and help earmark system components that they may be missing. For example, build in an Apple Guide "top ten" list of troubleshooting tips that automatically pops up when the user encounters an installation problem. Another helpful mechanism would be to include your customer support telephone number in any installation error messages that the user may encounter.

Which leads to the next killer assumption. . . .

**Killer Assumption #2
(Corollary to #1, Actually)**
It's a simple, straightforward title; it won't need much customer support. Support is one of the biggest issues facing title developers today. No matter how straightforward the title or elegant the interface, customers invariably need some sort of assistance.

Assuming otherwise can devastate your business. Especially in light of what I discussed earlier, throw this assumption to the wind and expect that you'll hear from customers who need help installing and running your

product. If you don't plan for the time and resources needed to take care of customer problems, inquiries, and returns, you'll find

Furthermore, if your product suffers from a poor interface, if it is low on the end-user delight scale, or if it is just a digital rehash

If you just dump a book, movie, or TV show onto CD-ROM, tout it as the interactive version, and assume it will have the same sort of user appeal, that assumption may bite you.

that your time and operating budget are consumed by these important tasks, and no amount of advertising, product reviews, or the like will save you from it. Your best bet is to be aware of the possibilities and plan ahead for them.

Killer Assumption #3
The original book/movie/TV show/video was successful, so the multimedia title will probably also be a hit. If your product is based on a work already published in another medium, you're likely to start out a step ahead, because you may already have at least some degree of name recognition with which to attract attention.

But name recognition alone ensures nothing. If a multimedia product that uses content originally prepared for another medium is to be truly successful, it may have to perform better than average—maybe even better than the content fared in its original form.

There are several reasons for this. For one, some content just doesn't translate well to a computer-based CD-ROM product. Also, the product's reputation in its original format (say, a book) sets customers' expectations and creates a standard, often a high one, for comparison.

of existing content without your having added that "special something," be prepared for yawns—and stacks of product returns. It's one thing to dump a book, recording, or piece of video onto a CD-ROM, add a spoonful of new material, and tie it all together with an interface. Unless this is done exceptionally well, the product will never be a distinct "whole"—it won't be more than a collection of parts.

The question I have to ask is, why build this multimedia title unless users can get something from it that they can't otherwise get from the originally published content? If you just dump a book, movie, or TV show onto CD-ROM, tout it as the interactive version, and assume it will have the same sort of user appeal, that assumption may bite you.

The truly exciting possibilities of multimedia are realized, and the real magic unfolds (and money is made), when a product is not just a digital rehash of content from a variety of media, but is instead an amalgam of content that allows users to see, hear, learn, understand, and interact with the material in new ways.

So, killer assumption #3 holds water only if you do a lot of things right—starting with putting this assumption aside.

Killer Assumption #4
"Enhancing" my product with extra features and capabilities will make it better and more appealing to users. In some ways, this assumption can result from the overzealous pursuit of avoiding killer assumption #3, combined with a fear of "product inadequacy." For whatever reason, some of us feel like we really have to include all the features our competitors do. Other people suffer from "blank sector syndrome," feeling that a product isn't complete if there's any space left on a CD-ROM.

In many cases, overdevelopment (in American slang, commonly referred to as *featuritis* or *creeping featurism*) results from creativity running amuck, spurred by the all-too-seductive belief that the product would be that much better if you added "just one more cool thing." Granted, sometimes you absolutely must make mid-course adjustments and add features or capabilities you hadn't originally specified. Staying open to unforeseen possibilities can help make a product better if you're judicious about how and when to take action.

But adding "just one more thing" again and again can be dangerous. Not knowing when you've "hit the wall"—when you can't develop any further because it's in your best interest to wrap it up and ship it—can hurt you in a variety of ways.

Because overdevelopment usually entails a larger number of design iterations, you risk overrunning the development budget. Furthermore, overall productivity decreases because the resulting benefit incurred from the extra iterations usually is disproportionately low; that is, project spending and company resource consumption increase, while the incremental profit margin on the product may very well stay the same or even decrease.

This is especially true if cluttering the product with unneeded features has made it less elegant, useful, or appealing; the resulting negative word-of-mouth publicity and less-than-glowing reviews may mean that fewer people buy your title.

Increasing the number of design or development iterations may also mean missing your announced ship date, sometimes more than once. (Of course, shipping a product before it's complete is also dangerous.) Delays sometimes result in missed opportunities: You may lose sales during a critical selling season, such as Christmas or late-summer (before school starts), or you may get to market later than a competitor does.

Delays also erode credibility with customers, the press, opinion leaders, and distributors, all of whom influence your long-term viability. And, of course, if you're a startup company, your investors will get restless.

The bottom line on delays is that if you miss a critical selling opportunity, it just doesn't matter whether you added some obscure feature or capability.

To avoid these traps, you've got to know when you've made this assumption—and acknowledge that this assumption is acted on frequently, even in your company. You can and should do something about it.

Killer Assumption #5

If my multimedia title is a real ground breaker, getting venture capital money should be relatively easy. First, as we all know, getting venture capital (VC) money is never easy. But in particular, I've recently been seeing that many venture capital firms already have made some level of initial investment in the multimedia area, and they are waiting to see what happens with it before taking on more.

While this doesn't mean you *can't* get VC money (and, of course, there are always exceptions), it does mean that it can be much tougher than you realize to get it from a VC firm, even for a really solid multimedia title idea. Assuming otherwise and not having an alternative funding plan

contains such information as what venture capital firms are and aren't; what attracts a VC firm; assumptions to avoid when working with a VC firm; criteria for selecting a VC partner; preparing your business plan and presentation for a VC firm; and more.

In many cases, overdevelopment results from creativity running amuck, spurred by the all-too-seductive belief that the product would be that much better if you added "just one more cool thing."

will kill your product while it's still just a gleam in your eye.

So when you're looking for seed money for a multimedia title or series of titles, it's absolutely critical that your business plan be exceptionally well thought out and complete, and that the numbers be well researched *and credible*.

Also, as with any other business, understanding the art and craft of approaching and working with VC firms is a must. The August 1993 issue of *Apple Directions* featured a good primer on the subject, titled "The Software Entrepreneur's Guide to Getting Venture Capital." If you don't happen to have that back issue in your archive, you can find it on AppleLink (path—Developer Support:Developer Services:Periodicals:Apple Directions:Apple Directions August 1993:Venture Capital) or at the *Apple Directions* Web site. (See "Internet Resources for This Issue" on page 36 for details.)

Written by a partner in a prominent VC firm, the article

Killer Assumption #6

My product "does" everything right, so it should sell itself. If your product is solid—it has a good interface, high user stimulation factor, good content, runs on common platforms/configurations, and so forth—then you've got the best possible start.

But interestingly enough, there's a catch: The market contains so much multimedia noise and so many mediocre products, that some buyers—especially the more experienced ones who have been burned in the early days of this medium—are becoming somewhat skeptical.

If your product really is that good, don't expect it to sell itself on its merits alone. (Actually, that goes for any product, any time.) You should be prepared to overcome the noise and preconceived market perceptions and sell, sell, sell. *Now* is the time to put away all your assumptions and use all the marketing know-how at your disposal. Then, your great product has a better chance of becoming a big hit.

Romeo and Juliet Revisited

The Romeo and Juliet puzzle inspired me to do my own reality check; I hope you, too, will take a few minutes to think about the assumptions you may be making and how they affect your company and products.

To get your thinking going, I'll pose another brain teaser, whose solution is on page 36. Take the challenge with a colleague; that way, one of you can look up the solution and answer the "yes or no" questions. Here goes, and good luck:

A man lives on the twentieth floor of a high-rise building. On sunny days, to get to his apartment, he rides the elevator up to the fifteenth floor and then takes the stairs the rest of the way up. On rainy days, however, he rides the elevator all the way to the twentieth floor. Why does he do this? ♣

© 1996 Dolores A. Kiamy, *Open Door Communications*. Reprinted by permission. All rights reserved.

Dee Kiamy is president of Open Door Communications, a marketing and communications consulting firm located in San Jose, California, that serves high-tech and entertainment industry companies. She is coauthor of the book The High-Tech Marketing Companion: Expert Advice on Marketing to Macintosh and Other PC Users, available from the Apple Developer Catalog. She can be reached at 408-266-9699 or at kiamy@applelink.apple.com.

Business Feature

The Changing Landscape of Software Distribution

By Kris Newby,
Apple Directions staff

Though software developers and distributors have the same objective—to sell lots of software—they see the business from fundamentally different points of view. Assuming you're a developer, your view of the software market is heavily influenced by *your* sales reports and *your* market niche. The value of talking to distributors is that they see the market from a higher point of ground. Distributors spend their days analyzing sales figures from hundreds of product lines and vendors, and in so doing, they often have a better grasp of the competitive landscape and emerging trends of software distribution.

To help you evaluate your own company's distribution plans, we interviewed two experienced software distributors: Steve DeWindt, CEO of AmeriQuest, and Tracy Robbins, director of software at Merisel. The benefit of these interviews are twofold: They offer you valuable software channel insights and advice on how to work more effectively with distributors.

A Global Perspective on Distribution

*Steve DeWindt, AmeriQuest
Santa Ana, California*

*Steve DeWindt, CEO of
AmeriQuest, offers a global view
of software distribution. DeWindt
recently came to the United
States from Germany, where he*

served as copresident of Computer 2000 (C2000), a software distributor that currently ranks first in Europe and third worldwide. After Computer 2000 bought a controlling interest in AmeriQuest, (a value-added wholesale hardware distributor based in Santa Ana, California), DeWindt was placed at AmeriQuest's helm. As the "new kid on the block" in a U.S. company, DeWindt has spent a great deal of time thinking about how to effectively compete against the "big three" American distributors: Ingram Micro, Merisel, and Tech Data. Here are his thoughts on software distribution today and his predictions for the future.

A State-of-the-Market Report

Within the software channel, profit margins have been eroding steadily. This has been frustrating, and it has some long-range effects on both developers and distributors.

This margin erosion is due to two things. First, the nature of software being sold through direct channels is changing. Several years ago business-oriented packages, such as spreadsheets and word processors, accounted for the largest volume of software sold. Today, most of these types of products are sold in bundles or suites—and many of them are pre-loaded on computers. Second, more business software is being sold through site licenses or as upgrades. So the ability for someone in the channel to sell a stand-alone package at a decent price is very difficult.

With the channel receiving less

money on individual software products, distributors can't afford to provide as much developer support—such as reseller training and additional demand generation—for these products. The funding for it isn't in the margin—it's too tight.

Many of the same trends within the U.S. distribution channel are also being seen in Europe: falling margins, tighter credit, and under-capitalized resellers. Of course there are differences as well. One of the most notable is the absence of aggregators in Europe. Another key differentiator is the lack of homogeneity: Each European country has unique differences and challenges.

More Site Licensing

On the business side of the software market, there are some other implications of this changing model. I mentioned that more of the revenue is coming from site or multiuser licenses. For a while, industry people were having trouble figuring out the logistics of this licensing, but for the most part they've solved the problem: The actual software can be transferred with a "golden disk" or electronically, and the credit, support, and administrative details are all handled by the channel. As the business has matured, the channel has tended to treat upgrades like any other SKU (shelf-keeping unit). Most vendors have worked out some sort of pricing scheme that allows all channel players to participate at an appropriate level. The amount of money that people make isn't quite as great, but frankly, the costs associated with this arrangement are not as great—

it doesn't take a lot of energy to warehouse a piece of paper.

SOHO Users, Kmart Shoppers

The small office/home office (SOHO) and "edutainment" side of the market presents a different set of dynamics.

First, there's been significant growth in this category, as the number of home and small business computers has grown dramatically, along with the demand for associated software.

We've also found that a lot of users in these markets don't like to download products electronically. They like to touch the products they're interested in, so merchandising is still very important. What we're finding, however, is that *where* these people are buying software is changing dramatically. They don't always go to a general computer reseller, such as Egghead or Computer City; they're buying software from places like Walmart or Sears.

To serve these users, a new distribution channel needs to evolve—as well as a whole new set of business terms and conditions. Support complexities also need to be worked out for this base of users. (One of the best examples of support issues occurred when Disney shipped their *Lion King* CD without sufficient configuration testing: They quickly discovered that they were completely unprepared for the support requirements of this product.)

Distributor Survival Strategies

In this era of shrinking profit margins, we all have different survival

strategies. Some distributors who were specialists have had to find new areas of specialization: Either they try to become a broad-line distributor or they focus on a certain customer or product set. For example, a number of software-only distributors have become education or retail specialists. I don't know of any company that's remained a general software-only distributor.

If you look at the broad-line distributors such as C2000, Ingram, or Merisel, there hasn't been a huge change in the way things are done—there's just been an acceptance that distributors are going to have to live with slimmer margins. The gross product margins of these three distributors have come down significantly over the last five years, and all of them have survived by reducing their cost structures. In C2000's case, software went from 25 to 19 percent of total revenues in the last two years, with the absolute revenue remaining fairly steady. (The rest of the business's revenue grew at around 30 or 40 percent.)

Because of this slowdown on the software side, there's a little less focus on providing new services or pushing software over other types of computer-related products. And what does this mean to software developers? The giants of the software industry, such as Microsoft and Novell, are having to manage more support and market-building activities on their own, since there isn't enough margin in the channel for distributors to do it.

Advice for Small Developers

Small software vendors, on the other hand, can still entice a distributor to support and focus on their products if they're willing to provide that distributor with an incentive, such as an exclusive distribution agreement for a specific region. This particularly

makes sense for developers who want to expand into markets outside their native country, but who don't have an abundance of capital for this expansion. An exclusivity arrangement makes it an easier justification for distributors to provide you with support, marketing, and other resources, because they know they'll be able to set their own prices and recoup their up-front investment.

The other avenue that's open to smaller vendors, particularly in the international arena, is to work with a subpublisher or republisher. One of the better examples of this type of arrangement is when Symantec worked with European Software Publishing (ESP) to expand into Europe. In this case, Symantec agreed to let ESP take care of everything: ESP received Symantec's code, translated it, duplicated it, created local language marketing materials and packaging, stocked it, sold it, and then paid royalties back to Symantec in the United States. This strategy worked extremely well for Symantec for a number of years. Obviously, this European expansion didn't hit Symantec's cash flow at all—it just provided them with additional profits. It wasn't until about three years ago that Symantec bought back their European distribution rights.

Most people want to focus on their home market and improving their products, so why not turn international distribution over to a republisher or distributor?

Electronic Software Delivery

After about three years of talk and exploration, electronic publishing is finally becoming a reality—primarily because of acceptance of the Internet. I think the real opportunity for electronic publishing today is in the professional software market. Once a corporation has a product like ClarisWorks installed throughout the

organization, it makes sense to receive upgrades electronically, distribute them from servers, and bill for the updates electronically—in essence, cutting out the middleman. That makes sense. The problem is that it's always more difficult to implement that kind of approach, because companies still like paper audit trails. But I still can foresee a time when more business software upgrades will be done electronically.

The Shoppers and the Shop-Nots

In the SOHO and edutainment markets, there's a raging debate about electronic distribution. Of course, I have my own theory, which is that electronic commerce will make up a certain percentage of the business—somewhere between 10 and 60 percent—but a large percentage of software will always be shipped in packages. My intuition is based on a couple of observations.

One observation is that if mail order has done well in an industry, there are always a lot of people who say, "This is the demise of stores—mail order is going to take this over." But in most mature markets, you'll find that direct marketing rarely occupies more than 20 percent of a market's total volume.

My other observation is very personal: It seems to me that shoppers inherently fall into two different categories—and they're pretty much typified by my wife and me. My wife is an engineer by trade, an absolute hacker, and a Macintosh fanatic—and she loves to buy things electronically or by catalog. She always prefers this mode over actually going out and shopping.

I, on the other hand, *really* don't like to shop electronically. It's too ethereal. Online I can only look at one thing at a time, and when I want to do comparisons, it's very difficult. It's not enough of a sensory experience. I like to

look at things on store shelves, and if something strikes my fancy, I like to buy it on the spot.

I think the world is divided that way, and whenever I present this theory to crowds, there are always lots of nods of acknowledgment. Some people will always want to buy boxes in stores, and online marketing will always miss that part of the population.

On Working With Distributors

My advice to developers who are approaching distributors is to formulate a clear distribution strategy. A software vendor should understand and articulate what role they expect a distributor to fulfill.

Often a developer will come to us and say, "All I want in a distributor is a time-and-place utility—someone to fulfill product, offer credit, and ship it." Then, in subsequent meetings we'll be surprised to hear, "We're getting calls that you folks aren't providing much technical support." We can only remind them, "Excuse me, but that's not what we were hired to do."

You can get distributors to provide you with a variety of services, but please state it clearly up front, and then make sure that the profitability of your line is sufficient to cover whatever expenses it requires.

Compare, for example, the distribution requirements of Microsoft Office, with margins in the low single-digits, to Informix, a product with considerably higher margins. The support required for each of those products could be about the same, but Microsoft clearly doesn't expect their distributor to provide much support. In selecting distributors, Microsoft focuses on a distributor's financial position, stock capacity, and asset terms. Informix, with higher margins, can afford to ask for a lot more service and support from their distributors. They

focus on staff training, phone and electronic response capabilities, and user training facilities. Each of these vendors has a very cohesive strategy, and it fits their expectations.

Worldly Advice

The other important thing that I wish developers would understand is how different terms and conditions are in different markets around the world. Often, developers have told me, "Our worldwide payment terms are net 30 days, period."

That's fine, I say, but what are you going to do in Spain, where no one gets paid in fewer than 60 days? Or in Italy, where you can expect payments to arrive in 90 to 120 days, unless you deal strictly in cash? And did you know that in Germany, Austria, and Switzerland, it's been a tradition for over a hundred years to offer resellers discounts if they pay with cash? Even if you don't acknowledge the cash discount in writing, people will take it anyway. This expectation has precipitated huge arguments with vendors, because they didn't realize that this discount structure needs to be built into their pricing.

Some Model Software Vendors

In thinking about model software vendors, I believe Symantec does a brilliant job on a number of fronts. First, they have a very clearly defined target market. They say, "This is the market we're going after; therefore, this is the channel we need, and this is what we expect from our channel." They have different product sets, and within each set they have different requirements of their channel partners. One product set needs distributors that just provide "stack 'em high, and let 'em fly" services. Another product set requires a lot of demand generation and customer support, and for that line, they have a different set of distributor requirements.

The other thing Symantec does that is very interesting is that they don't record revenue until it is *sold out* of reseller locations. They have a very deep understanding of what's sitting in inventory. And as a result, they're able to watch their financials in an incredibly uncanny fashion. The weekly sales-out reports that they get from their distributors is an extra expense, but I think it's worth it.

Finally, I appreciate Apple Computer as a vendor in Europe: Apple Europe's new president, Marco Landi, seems to be doing a great job. One of the good things he's done is to visit all of Apple's European channel partners and explain in great detail what he wants from the channel. In essence, he's told these partners, "Here's our strategy, here's the role I'd like the channel to assume, does this make sense, and more important, how can we help you?" As a result, there's a nice open dialog, and Apple's sales are increasing accordingly in this region.

A Commodity Perspective on Distribution

*Tracy Robbins, Merisel
El Segundo, California*

The second distribution expert we interviewed, Tracy Robbins, is the director of software at Merisel, Inc. Merisel, with projected 1995 revenues of \$5.8 billion, is currently the largest publicly-held software distributor in the world today. This company is best known for its broad product lines, streamlined logistics capabilities, marketing savvy, and some of the toughest product return policies in the business. With seven years of experience at Merisel, Robbins brings a commodity-focused perspective to our look at software distribution.

1995: A Competitive Year

Last year was a very competitive year for a lot of software manufacturers—especially if you look at the increase in the number of software titles, and the growing requirement for developers to more fully support their resellers.

Windows 95 was one of the few major product launches this year, and there was a slowdown in sales pending the release of this operating system, followed by a sales surge when it was finally released. It took a little longer than we thought for customers to get up and running with new Windows 95 products. After this initial period, we went into a fairly strong fourth quarter, though I think it was a little softer worldwide than everyone had hoped.

This year, Merisel's Macintosh software sell-through rates have been fairly stable, and we've met most of our objectives. But it's a challenging time right now for Apple, because a lot of resellers are wondering what Apple's new product line strategy will be.

In the consumer software arena, the large number of titles on the market right now is confusing resellers and consumers. It's hard for resellers to decide what products should be on the retail shelves and how long they should stay there.

There are also a lot of newcomers to the channel: This year we saw a 20- to 30-percent increase in the number of mass merchants carrying software. This can create a challenge for software vendors, because mass merchants are limiting their relationships to the software publishers who can provide sophisticated distribution capabilities and a steady flow of products. Mass merchants are also looking for developers with effective merchandising and marketing programs that can help them minimize their inventory and maximize their return-on-investment.

Merisel's Challenges

At Merisel, we are focused on maximizing the profit in our business. Making the most of your inventory is very crucial in distribution. We're dedicated to making sure that we're carrying the right inventory levels and that we don't incur excessive inventory holding costs. And we're monitoring our sell-through rates and revenues to ensure that we're doing business effectively and profitably.

With intense SKU proliferation, it becomes very expensive and inefficient for us to carry a broad range of products from every software manufacturer. It becomes more important to select the products that our customers really want. We're continuing our efforts to reduce operating costs and product returns, so that we can return more profit and less product to our software manufacturers. We are also working with our manufacturers to help them understand that because we're a demand-driven business, we cannot carry inventory beyond our immediate needs.

Advice to Developers

I think it helps for developers to understand the function of a distributor and what we're here to do—to buy, inventory, sell, and ship products.

One of our major concerns today is limited shelf space, and along those lines, my advice to developers is to try and combine as many products into single SKUs as possible. Understanding packaging is also critical: The box not only has to look good, but it must also fit the space requirements of all your resellers—an odd form factor could result in more shipping damage and product returns. And use the universal product code (UPC) bar-code standard on your packages.

Other advice is to always keep your "middlemen"—your distributors and resellers—in the loop if

you make any dramatic changes in your terms and conditions, credit terms, or return policies. For example, if you change the way you accept returns, you need to communicate that change to your distributors and resellers. Each channel partner needs to make sure that it's taking ownership of the inventory that's purchased, and that it's the right amount of inventory.

Alternate Channels

Merisel has a department that's working internally and with manufacturers to figure out how to make electronic distribution part of our business. We think that this channel will definitely be an important component of software distribution, and we'd like to be involved in the leading edge of it. D.A.T.A. Bank, our data ware-

house and fulfillment center, is a first step. Through D.A.T.A. Bank, resellers can quickly access information such as product descriptions and SKU details from our SELLine online order entry service *D.A.T.A. Bank on CD* or fax-back system.

I don't, however, think that traditional software stores will ever go away. There's a real need for some consumers to look at and hold a tangible product.

Developers With the Right Stuff

In looking at developers from a distributor's perspective, I think Claris is doing a lot of the right things in the market. They certainly understand how to carry the proper amount of inventory and deliver it on time. They build a superior product, and they've

been able to maximize the awareness of their product to continue building their revenue stream. They're also fairly flexible in meeting our inventory and return needs. They have a stable product, they let the demand drive the product, and they're good at reacting to what the channel wants.

I also appreciate Intuit's recent experiments in getting products through the channel faster. They've spent a lot of time looking at "just-in-time" techniques for shipping and working with distributors.

Microsoft, of course, has repeatedly put together marketing programs that seem to work in the channel. They offer a variety of good marketing and support programs, such as support for Windows NT, road shows, and

"authorized NT reseller programs." They're also pretty good at reading the needs and wants of the channel.

A Vote of Confidence

In looking at the Macintosh software business as a whole, a lot of people feel that Macintosh software products are very much needed by the creative people in this world. There's also been a great deal of innovative work done in areas such as Web page development—for example, 35 percent of Web pages are developed on Macintosh computers. There's a whole lot of opportunity for software vendors and resellers in the areas of the Internet and multimedia. ♣

Listings

It Shipped!

The following 59 Macintosh products were entered into the It Shipped! database between January 15 and February 15, 1996. Congratulations to all of you with new shipping products!

More than 200 have been entered into the database since Apple reinstated the program in late 1995.

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 at the following location: <http://dev.info.apple.com/itshipped.html>

To enter your Macintosh product in the database, use the form located at the following Web 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
A.D. Software	OOFILE	1.1
Adobe Systems, Inc.	Adobe SiteMill	1
AKS Acoustics	Bass Pump	2
Artel Software, Inc.	Boris Effects—3D Effects for Media 100, Premiere	1.01
Atlantic Records	Spew+	1
BadgerCom	MARS: Prince of Destruction	1.11
Bradley S. Pliner	FormSaver	1.2
BrainScan Software	BrainHex	1
Cordant Imaging Systems	Portable Safekeep	4.04
Cordant Imaging Systems	Safekeep	4.03
Crystal Structure Design AS	CrystalDesigner	DR4

Company	Product	Version
Davidson & Associates	Kid Works Deluxe	
Davidson & Associates	Magic Tales: Baba Yaga and The Magic Geese	
Davidson & Associates	Magic Tales: Imo and The King	
Davidson & Associates	Magic Tales: The Little Samurai	
Davidson & Associates	Fisher-Price Sing-Alongs	
Davidson & Associates	Your Personal Trainer for the ACT	
Davidson/FUNNYBONE Interactive	Puddle Books: A Day at the Beach with the Fuzzooly Family	
Davidson/FUNNYBONE Interactive	Puddle Books: After the Beanstalk	
Davidson/FUNNYBONE Interactive	Puddle Books: The Lie	
DB:Solutions, Inc.	DB:\$ Retailer's Advantage	8.05
Diehl Graphsoft, Inc.	MiniCad 6.0	6
Dynamic Engineering	MacPortable Fax/Modem	1
ExtraWare International	Scan!Plot	1.2
FORE Systems, Inc.	ForeRunner ASX-1000 ATM LAN Backbone Switch	
FORE Systems, Inc.	ForeRunner ES-3810 ATM-Ready Ethernet Switch	
FORE Systems, Inc.	ForeRunner PCA-200EMAC PCI Bus ATM Adapter for Power Macintosh Computers	
guideWorks, LLC	guideWorks Translator	1
Inedit Software S.L.	QuickBar	1.7
Interactive Media Corporation	Interactive Multimedia—Getting Started	2.1
Mesa Graphics	Mesa Graphics Plotter Utility	
Metrowerks Corporation	CodeManager DR1	DR1
Metrowerks Corporation	CodeWarrior Academic	8
Metrowerks Corporation	CodeWarrior Bronze	8
Metrowerks Corporation	CodeWarrior Gold	8
Metrowerks Corporation	Discover Programming for Macintosh	
NSC Software Construction	ScreenLight, ScreenSaver especially for Power Macintosh	1.0
OpenLink Software, Inc.	OpenLink High-Performance ODBC Drivers	1.1
Pascal Van Londersele	SmartPage	2
PAX Analytics, Inc.	ACCESS Environmental Monitoring System	1.6.1
PeterSoft Multimedia	Launch O Matic	2.3
Phase One	Phase One digital photography solutions for the professional photographer	2.5
RES Software	World of Words	2
Research Systems, Inc.	IDL 4.0.1 for the Macintosh/Power Macintosh	4.0.1
Ronald P. Vullo, Ph.D.	Traci (Temporally Real Automatic Chat for the Internet)	1.0
Rubi Inc.	HemaTool 1.1	1.1
Salford Systems	CART V.2.0—Classification and Regression Trees	2
SAS Institute Inc.	JMP Statistical Discovery Software Version 3.1.5	3.1.5
Softing	EasyBeam	1
Square Dance Action	Square Dance Animation S/W for Dancers & Callers	1
Stairways Software Pty Ltd	NetPresenz 4.0	4
Storm Software	EasyPhoto	1.5
T/Maker Company	VroomBooks Holiday Activity Fun Pak	
T/Maker Company	VroomBooks The Escape of Marvin the Ape	
Tropical Penguin Software, Inc.	Cool Coach for swimmers	1.5
WIN-SYSTEM EUROPE	FORTERA SUBSYSTEMS	
WorldWired, Inc.	cT Professional	2.5
YARC Systems Corporation	YARC-RIP for EnCAD NovaJet Large Format Printers	2

Developer University Schedule

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

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

Classroom Courses

Apple Events/AppleScript Programming

5 days/\$1,200

May 6–10, Cupertino

Creating Apple Guide Help Systems

4 days/\$1,200

May 6–9, Cupertino

Creating OpenDoc Parts

5 days/\$1,500

April 15–19, Cupertino

May 20–24, Cupertino

June 17–21, Cupertino

Macintosh Debugging Strategies & Techniques

3 days/\$900

April 8–10, Cupertino

Multimedia Development with QuickTime VR

3 days/\$900

March 19–21, Cupertino

April 16–18, Cupertino

May 21–23, Cupertino

June 18–20, Cupertino

Newton Programming: Essentials 2.0

5 days/\$1,500

April 15–19, Cupertino

May 20–24, Cupertino

June 17–21, Cupertino

Newton Programming: Converting to 2.0

5 days/\$1,500

April 1–5, Cupertino

May 6–10, Cupertino

June 3–7, Cupertino

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

May 20–22, Cupertino

QuickStart Mac OS Programming

5 days/\$1,500

April 15–19, Cupertino

June 3–7, Cupertino

Scripting with AppleScript

2 days/\$600

April 8–9, Cupertino

June 10–11, Cupertino

Self-Paced and Online Courses

Intermediate Programming: 7.5 Topics

Self-paced

Introduction to PowerPC

Online

Introduction to PowerTalk

Online

Introduction to RISC Technology

Online

Programmer's Introduction to PowerPC

Online

Programmer's Introduction to RISC and PowerPC

Self Paced, Online

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

Apple Sites

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

Apple Developer Services and Products

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

The main page for Apple Computer's developer services, including back issues of *Apple Directions*.

Apple Directions Express List Server

<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>".

Guy Kawasaki's EvangeList List Server

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.)

Key Apple Developer Relations Contacts

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

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

Apple FTP Sites

<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.

Development Tools

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

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

Apple Computer

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

The Apple Computer home page.

Mac OS

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

Go here for the latest information on the Mac OS.

Technotes

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

Contains all Technotes—new and old—as well as author's guidelines in case you want to contribute your own technical notes.

Apple Pacific

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

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

Apple Europe

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

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

Apple International Developer Services and Products

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

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

Apple Software Licensing

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

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

Apple and the Internet

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

Information about Apple's Internet strategy and products.

Newton

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

Includes information about Newton 2.0 and Newton Toolkit 1.6.

OpenDoc

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

The place to go for the OpenDoc 1.0 SDK and OpenDoc sample parts.

Third-Party Products

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

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

It Shipped!

<http://dev.info.apple.com/itshipped.html>
<http://dev.info.apple.com/thirdparty/submission.html>

The first site provides information about the It Shipped! program. The second site contains a form that you can use to get your product into the It Shipped! database.

Apple Multimedia Program

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

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

The Macintosh Advantage

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

The source for official Apple ammunition to fight the war against Windows 95.

Macintosh PowerBook and Mobile Computing

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

Complete information about PowerBook computers.

PowerTalk

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

Resources for PowerTalk programmers, including the StarNine gateways recently licensed by Apple. As of this writing, the site doesn't yet contain further information about Apple's replacing PowerTalk with the Internet and OpenDoc as the core of its communications architecture.

QuickDraw 3D

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

Everything you need to know about QuickDraw 3D, Apple's new 3D graphics architecture.

Power Macintosh

<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.

QuickTime

<http://quicktime.apple.com>

News and technical and marketing information about QuickTime.

QuickTime VR

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

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

QuickDraw GX

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

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

Apple Education

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

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

Macintosh Application Environment

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

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

Pippin

<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.

eWorld on the Web

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

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

QuickTime Live!!

<http://live.apple.com>

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

Apple Internet Servers

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

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

Non-Apple Sites

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

Internet Search

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

Provides access to InfoSeek, Lycos, WebCrawler, and other Web search engines.

Macintosh Vendor Directory

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

A directory of companies with products for the Macintosh computer.

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.

Macintosh Programming Tools

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

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

OpenDoc Part Ideas

http://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.

CI Labs

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

Provides a great deal of OpenDoc content.

MacTech Magazine

<http://www.mactech.com>

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

Nathan's Everything Macintosh Page

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

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

Digitool (Macintosh Common Lisp)

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

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

The Ultimate Macintosh Page

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

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

guideWorks

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

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

The 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

New This Month/From Our Readers

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

Guy Kawasaki's Semper.fi List Server

Guy set up this two-way list server to encourage communication between developers and Apple. When you subscribe, e-mail messages you send to the server can be read by every other subscriber, and you can read everyone else's e-mail messages, as well. Many Apple employees—including executives—also read Semper.fi and answer your comments, suggestions, and questions. Subscribers post dozens of memos to the list every day, so we suggest you get the digest version, which combines several postings into a single e-mail; to do so, send a message to listproc@solutions.apple.com. In the body of the message, type "set semper.fi mail digest".

Apple Forever

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

Apple uses this new Web site to post regular updates about the company and special communications from Apple executives.

Dr. Gil Amelio

<http://www.natsemi.com:80/profit/gil.html>

Apple's new CEO and chairman has his own Web page, primarily with information from and about his recently published book, *Profit From Experience*.

Macintosh Speech Recognition

http://www.vannevar.com/Mac_SR

A developer has been good enough to use this server to provide software

he developed using Apple's PlainTalk speech recognition system. You can find out more about PlainTalk and post your own speech recognition software at this site; its owner says he hopes it "will become a showcase for Mac speech recognition tips, tricks, and software."

Ambrosia Cafe

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

Yet another "everything you want to know about the Macintosh" page, this one with an archive of technical resources for Macintosh programmers, three "e-zines," an index of Macintosh game resources available on the Internet, a page with info about how to use the Macintosh computer to access the 'net, and more.

Apple Developers Listing

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

A very useful site containing links to the Web pages of literally hundreds of Apple platform developers.

The Hartsook Letter

<http://pserver.hartsook.com/root/The%20Hartsook%20Letter%20homepage>

This site contains excerpts from *The Hartsook Letter*, written by long-time Macintosh market analyst Pieter Hartsook; the newsletter contains market data and Mr. Hartsook's analysis of trends affecting the Macintosh market.

Natural Intelligence

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

This site provides information about Natural Intelligence's tools and solutions—specifically, Roaster, the first Java development environment for the Macintosh, and Roaster Professional, the first Java development environment designed specifically for application development.

development of QuickDraw GX; the second provides addresses of two e-mail lists for receiving updates about QuickDraw GX.

PC Fairy Tales

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

Information to help debunk common Macintosh myths.

Metrowerks

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

Go here to find out about Metrowerks' CodeWarrior PowerPC development environment.

Pictorius

<http://www.pictorius.com>

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

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.

The 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.

MacSciTech

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

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

MacHack

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

Go here to find out about the annual MacHack hackers conference.

Macintosh Help Wanted

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

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

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

Cult of Macintosh

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

This is the site of another "everything Macintosh" compendium of information for Macintosh lovers.

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.

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.

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.

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.

Pages That Employ QuickTime VR

<http://www.bmwusa.com/ultimate/roadster/z3downloads.html>

Take the new BMW Z3 roadster for a Quicktime VR test drive.

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

Shows the 1995 World Series in QuickTime VR.

<http://sfasian.apple.com>

View a QuickTime VR version of an exhibit of Mongolian art.

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

See QuickTime VR interiors of Honda Motor Company's new car models.

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

Real estate tours of actual, for-sale properties.

Top 100 Computer Magazines

<http://www.internetvalley.com/top100mag.html>

A list of the 100 top online computer magazines—including our sister publication, *develo*p—as determined by the site's owners.

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?

Nisus Software

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

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

Internet Resources for This Issue

News

- mFactory Web site—<http://www.mfactory.com>
- Apple Worldwide Developers Conference information—<http://www.info.wwdc.carlson.com/>
- Motorola Web site—<http://www.mot.com/GSS/MCG/>
- Club OpenDoc : <http://www.software.ibm.com/clubopendoc/>

Technology

- Apple's guides to client/server computing—<http://devtools.apple.com/>
- Object Management Group (OMG) Web site—<http://www.omg.org>

Business

- *Apple Directions* Web site—<http://dev.info.apple.com/>

Brain teaser solution from page 25 of the “Six Assumptions” article:

The man is a midget and, unassisted, he can only reach the fifteenth-floor elevator button—except on rainy days, when he carries his umbrella and can use it to reach up and push the twentieth-floor button. The elevator buttons are arranged in a vertical column, with the first-floor elevator button on the bottom and twentieth-floor button on the top.

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. You can also reach us by AppleLink at APDA or by e-mail at APDA@applelink.apple.com. More detailed Apple Developer Catalog ordering information is available at the following locations:

- Internet—order.adc@applelink.apple.com
- AppleLink—Developer Support:Developer Services: ORDER.ADC