

AppleDirections

CONTENTS

NEWS	
New PowerBook 3400 Is Fastest Ever	1
Strategy Mosaic: Delivering Container Support With ALOE	1
What's New?	2
Apple Announces Further Details of New Organizational Structure	8
New Power Macs Pack More Punch	8
New Spring Line-up of Educational Products	9
The Apple QuickTake 200 Camera	9
PowerPC Chip KO's the Pentium—Even With MMX	10
Proof That the Mac OS Platform Is Alive and Well	10
Apple Demos DVD Prototype	
TECHNOLOGY	
CD Highlights: System Software/SDK Editions, April 1997	11
Human Interface: Users? What Users?	12
MacApp Release 13: A Major Update	14
BUSINESS	
Running Direct Mail Campaigns by the Numbers	16
Internet Resources for This Issue	20

APPLE NEWS

New PowerBook 3400 Is Fastest Ever

It's been clocked at speeds of up to 240 MHz. Its web site beckons you to take it for a test drive (http://powerbook.apple.com/phase_2/show.qry). If you decide to accept the offer, better fasten your seat belt—it looks like the PowerBook 3400 promises to deliver one heck of a ride.

Sporting such features as a 240 MHz 603e PowerPC™ processor, a hot-swappable expansion bay that can contain a CD-ROM drive or a third-party Zip drive, a 12.1-inch color screen, a four-speaker sound system, and built-in Ethernet and modem connections, the new PowerBook 3400 has got what it takes to meet the needs of Macintosh customers in business, higher education, scientific and engineering markets, and design who are looking for the best in performance, multimedia, and communications from a notebook computer.

“The introduction of the PowerBook 3400 represents a significant milestone in the revitalization of the PowerBook line,” said Dale Fuller, vice president of PowerBooks at Apple Computer. “Apple is out to regain leadership in the notebook market. With the PowerBook 3400, we do just that by fulfilling the needs of performance-hungry mobile professionals.”

“With up to 240 MHz of processing power and a 12.1-inch screen, the PowerBook 3400 is the first true portable to market that can be considered a viable desktop replacement,” said Tim Bajarin of Creative Strategies.

Performance

By combining a number of key technologies such as industry-standard PCI architecture, PowerPC processors, DMA (direct memory access), faster EDO (extended data output)

please turn to page 7

STRATEGY MOSAIC

Delivering Container Support With ALOE

*By Gregg Williams,
Apple Directions staff writer*

OpenDoc has come a long way. It's now a part of system software in Mac OS 7.6 (along with a set of OpenDoc parts called the Apple OpenDoc Essentials). In addition, leading-edge developers are selling compelling OpenDoc parts; see the box “Noteworthy OpenDoc Parts” on page 4 for a partial list. Several applications—including Nisus Software's Nisus Writer 5.0 and B&E Software's Ragtime 4—are container applications (meaning that their documents can contain OpenDoc parts), and future versions of Corel's CorelDRAW and Corel WordPerfect are also expected to be container applications.

What's the next step for you? You should consider turning your application into a container application (that is, one that supports the embedding of OpenDoc parts). Apple has developed a new technology called ALOE (Apple Library for Object Embedding), which works with OpenDoc 1.2, a new version of OpenDoc, to deliver a robust solution for container support. ALOE hides the complex OpenDoc API (application programming interface) and gives you the simplest possible API for converting your application into a container application. It is an Apple-supplied technology that is tightly integrated with OpenDoc, and Apple will continue to support and enhance it as the recommended way to implement container support.

ALOE puts container support within the

please turn to page 3

AppleDirections

Volume 5, 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.

Editor

Patty Bing-You (bingyou@apple.com)

Technical Editor

Gregg Williams (greggw@apple.com)

Business Editor

Kris Dalebout Newby (kris@newbytech.com)

Associate Editor

Anne Szabla (szabla@applelink.apple.com)

Barbara Milligan

Production Editor/Graphic Designer

Lisa Ferdinandsen (ferdinan@apple.com)

Contributors

Laurie Beasley, Meredith Best, Peter Bickford, Michael D. Rosetti

Production Manager

Diane Wilcox

Prep and Print

Consolidated Publications, Inc., Sunnyvale, CA

© 1997 Apple Computer, Inc., 1 Infinite Loop, Cupertino, CA 95014, 408-996-1010. All rights reserved.

Apple, the Apple logo, APDA, AppleLink, AppleScript, AppleShare, AppleTalk, A/UX, Bento, ColorSync, DigiSign, Firewire, GeoPort, HyperCard, ImageWriter, LaserWriter, Mac, MacApp, Macintosh, MacTCP, MessagePad, MPW, Newton, OpenDoc, Performa, Pippin, PlainTalk, PowerBook, Power Macintosh, PowerShare, QuickDraw, QuickTake, QuickTime, StyleWriter, and TrueType are trademarks of Apple Computer, Inc., registered in the U.S. and other countries. All great software wears this face, AppleNet, Cocoa, Cyberdog, develop, eMate, Finder, Macintosh PC Exchange, Power Mac, and StartingLine are trademarks of Apple Computer, Inc. Adobe, Acrobat, Illustrator, Photoshop, and PostScript are trademarks of Adobe Systems Incorporated, which may be registered in certain jurisdictions. FaceSpan is a trademark of Software Designs Unlimited, Inc. Java is a trademark of Sun Microsystems, Inc. Netscape Navigator is a trademark of Netscape Communications Corporation. Pantone is a registered trademark of Pantone, Inc. PowerPC is a trademark of International Business Machines Corporation, used under license therefrom. RealAudio is a trademark of Progressive Networks, Inc. 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

What's New?

Apple Developer Relations recently completed a worldwide research study to evaluate the design and delivery of the programs and services it provides to Apple developers. Based on recommendations stemming from this research, ADR is making changes over the next few months to improve the level of service and support you now receive. We will be reporting all changes to you in future issues of Apple Directions and at the Developer World web site (<http://devworld.apple.com/>).

*Patty Bing-You
Editor*

• • •

This month's "What's New?" focuses on the exciting changes coming your way from Developer University (DU), including everything from three new in-depth classroom courses to training in more affordable formats like self-paced classes and free online training, to make sure you can get the training you need, when you need it, at a price you can afford.

Results from the recent worldwide developer research study indicate that the majority of you prefer to learn new technical skills using self-paced training (48%) or online training methods (14%) rather than formal classroom instruction (6%). Based on this feedback, DU has made some changes in their program to expand the types of training they offer. Depending on your needs, you can choose from three different delivery methods: instructor-led courses, self-paced courses, and free online training. Each method is designed to deliver maximum training in minimum time, so you can start using the technology immediately.

The March 1997 DU class schedule features seven instructor-led courses, three of which are new for this session: "Multimedia Development with QuickTime VR 2.0," "Programming Java™ on the Macintosh," and "Scripting with AppleScript."

"Multimedia Development with QuickTime VR 2.0," Developer University's new instructor-led class, and its companion self-paced

training course, "Virtual Tutor for QuickTime VR," take you deep into the next generation of multimedia. Now, with QTVR 2.0—Apple's award-winning, nonlinear, panoramic movie format—creating QTVR panoramas, objects, and scenes has never been easier. You'll use the tools, techniques, and production processes involved in creating QuickTime VR scenes. Even if you're already familiar with QTVR, the course will give you the tips and tricks you'll need to master all the features of the new 2.0 version. The QuickTime VR course is also available online.

The "Multimedia Development with QuickTime VR 2.0" course is offered in Cupertino, California, March 17–20, April 14–17, May 19–22, July 7–10, and August 25–28, and in Boston, Massachusetts, May 6–9 and August 12–15. Cost for the four-day class is U.S. \$1200.00.

The complete "Virtual Tutor for QuickTime VR" self-paced course includes a workbook and companion CD-ROM (*Apple Developer Catalog*, #R0717Z/A, U.S. \$79.95). A special bundle version of QuickTime VR 2.0, which includes the Virtual Tutor self-paced training package, is also available from the *Apple Developer Catalog*, #B3003Z/A, U.S. \$439.00. Other self-paced courses include "Object-Oriented Fundamentals" and "Multimedia Development with Apple Media Tool."

In "Programming Java on the Macintosh," you'll learn all about the world of Java. The new four-day course will teach you everything you need to know about the language, libraries, and tools you need to create Java applets and applications. It will also help you identify appropriate uses for Java, chart the general direction of future changes in the Java platform, and compare the tools available for creating Java experiences. The course is ideal for programmers with experience in high-level programming languages like C, C++, or Pascal. While the course isn't aimed at web publishers, it can still teach you how to use simple Java applets to enrich your web site.

May Apple Directions Online

The May issue of *Apple Directions* will be available by April 15 on the web at <http://www.devworld.apple.com>.

“Programming Java on the Macintosh” is offered only in Cupertino March 24–27, April 21–24, May 19–22, June 30–July 3, and August 4–7. Cost for the four-day class is U.S. \$1200.00.

“Scripting with AppleScript” is a new two-day class that will teach you how to automate

and customize your work on a Macintosh computer using the AppleScript scripting language. In this class, you’ll learn how to use AppleScript to control and integrate multiple applications, including the Finder, and gain experience in using advanced scripting techniques, such as subroutines and networking.

You’ll also have an opportunity to use FaceSpan to add graphical interfaces to create complete, stand-alone applications, and you’ll learn to use AppleScript to create CGI applications for your Mac OS web server.

“Scripting with AppleScript” is offered in Cupertino April 21–22, June 23–24, and August 25–26 and in Boston on April 23–24 and July 23–24. Cost for the two-day class is U.S. \$400.00.

For more information on any of the courses, including course descriptions, course outlines, target audiences, and prerequisites, or to learn more about self-paced training or online courses, visit the Developer University web site at <http://devworld.apple.com/du> or <http://devworld.euro.apple.com/du> or you can contact Developer University by phone at 408-974-4897, e-mail <devuniv@apple.com>, or fax at 408-974-0544.

To register for classes, call 1-800-282-2732 (U.S.), 1-800-637-0029 (Canada), 1-716-871-6555 (outside U.S. and Canada), or send an e-mail to <order.adc@apple.com>. ♣

DevWorld— Members Only



The only place on the WWW for Apple installed-base and market intelligence information, Mac technology seeding, country-specific purchase program information, and price lists *plus* Discussion Boards and Live Chats with the development community in mind.

Registration is fast and easy. All you need is a computer with access to the Internet and your Apple Customer Membership Number.

Point your browser to <http://www.devworld.apple.com> and click on “Register Now.”

Note: If you are a developer in the United States who has been accessing the Hardware Purchase Program Price List from the DevWorld Members Only web site, the current user name and password will remain in effect for several more weeks—you will need a private ID to enter after that date.

STRATEGY MOSAIC

ALOE

continued from page 1

reach of almost every developer. You can implement container support using as few as 12 ALOE routines (though the number of routines needed will be proportional to the complexity of your application). You can call ALOE routines from C, C++, Pascal, or 680x0 assembly-language code—meaning that most applications, not just those written in C or C++, can use ALOE. All in all, you can expect to add ALOE container support with only a modest development effort—somewhere between a few weeks and a few months of work.

Why Have a Container Application?

The reasons for turning your application into a container application are the same as they have always been: to add value to your application and to make it more useful and attractive to your customers.

Perhaps the most important reason for having a container application is that, through the ability to contain OpenDoc parts, your application can support new data types—and when Apple (or some other company)

provides a viewer or editor part for that data type, your application will automatically support it with no extra work from you. For example, the Apple OpenDoc Essentials product includes viewer parts for audio, QuickDraw 3D objects, and 2D graphics (in PICT, GIF, TIFF, or JPEG format). (OpenDoc Essentials does not currently include a QuickTime viewer part, but Apple expects to add one this May.) If your application is a container application, it automatically supports these data types through these Apple-supplied viewers. In addition, Apple’s Cyberdog parts automatically give your applications Internet access.

Because of OpenDoc’s widespread acceptance as a well-recognized architecture for component software, your application can gain access to other types of component software. Through the Apple Applet Viewer (an OpenDoc viewer part that is included in Mac OS Runtime for Java), your container application can embed Java applets. As it turns out, Microsoft has acknowledged the importance of OpenDoc by announcing that it will create an adapter that will allow ActiveX objects to be embedded in OpenDoc containers. Once that becomes available, your container application

will be able to embed ActiveX objects.

Another reason for having a container application is that it allows you to add new functions to the application itself by adding OpenDoc parts instead of by adding extra code to the application itself. Because OpenDoc part editors are often small programs that perform only one function, they are easier to write, debug, maintain, and distribute—and you don’t have to worry about dependencies existing between the part’s code and the code in your main application.

What About Rhapsody?

You may be thinking that, because of Apple’s upcoming Rhapsody operating system, putting more development effort into Mac OS 7.x applications might not be a good idea. Believe me, that’s not a valid concern. Apple is pursuing a dual operating system strategy, which means that the current Mac OS will be around for several years to come. Also, Rhapsody will include a Mac OS compatibility feature (also known as the “blue box”) that will enable customers to run most of their Mac OS software under Rhapsody—and that includes OpenDoc and ALOE-based container applications.

Because of these factors, the market for Mac OS applications—including your ALOE-based Mac OS container applications—will remain strong for the foreseeable future. Adding container support makes good sense: With a modest development effort, you can add significant value to your application and sell it to more customers.

ALOE: A Solution for Container Support

The purpose of ALOE (which, in an earlier form, was known as ODIA) is to allow you to convert your application to a container application, a conversion that is extremely difficult to do using the OpenDoc API. ALOE acts as a software layer between your application and OpenDoc, so you don't have to learn OpenDoc or make any calls to OpenDoc routines when you use ALOE to convert your application to a container application.

ALOE has a scalable API; that is, it's only as complex as it needs to be to fully support your application. If your application doesn't

support certain Mac OS features—for example, floating windows or the Edition Manager—then you don't have to deal with the ALOE routines that support those features.

ALOE Services

The ALOE API provides services that manage various aspects of container support for OpenDoc parts, thus making it feasible for you to add container support to your application.

Among these services are the following:

- Initialization and shutdown of OpenDoc
- Handling of OpenDoc-related events
- Support for the Clipboard, drag-and-drop, scripting, and linking between embedded parts
- Storage of embedded-part data when the container document is saved or opened
- Menu handling
- Rendering of part content embedded in a container document
- Reconciliation of the Undo mechanisms of OpenDoc and the container application

- Support for the application's floating window scheme
- Scripting of embedded OpenDoc parts (even if the application is not scriptable)

How ALOE Works

ALOE enables the embedding of OpenDoc parts into a document while minimizing the number of changes you must make to the application.

For an ALOE application (that is, an application that uses ALOE to implement container support), when a part is embedded in a window, ALOE creates an invisible *proxy root part*—an OpenDoc container part—to contain the embedded part. Any OpenDoc part that is embedded in the container window actually “lives” inside the proxy root part, which acts as a “bridge” between the application's process and the OpenDoc process.

The container window “sees” things differently. The ALOE application (and here I'm simplifying a bit) deals with embedded OpenDoc parts through three objects called an *adapter*, an *adapter view*, and an *adapter manager*:

The adapter represents the part's data, and the adapter view handles the layout and rendering of the adapter; in other words, the adapter view is the visible representation of the OpenDoc part that gets drawn inside the container window. (Usually, an adapter has only one adapter view. However, ALOE allows a document to contain multiple views of a single adapter—for example, when a split-pane word processor window shows two different sections of the same document.) The adapter manager, one of which exists for each application document, manages the document's adapters and is involved in ensuring that each adapter's content gets stored as part of the document's data.

ALOE's implementation minimizes the amount of work you need to do to add container support to your existing application, and it allows you to do so without having to make major structural modifications to your application. ALOE gives you a handful of routines for dealing with adapters, adapter views, and adapter managers—and these routines hide the complexity of maintaining OpenDoc parts and of allowing them to exist inside a non-OpenDoc environment.

Adding ALOE container support will be easy if you've ever modified your application to allow it to display some kind of foreign content—a QuickTime movie or a QuickDraw 3D

Noteworthy OpenDoc Parts

OpenDoc parts are here! 1996 saw the emergence of a variety of full-featured, commercial-quality OpenDoc parts. Here is a list of companies and parts that I am aware of. (My apologies to any vendors whom I have inadvertently left out.)

- Adréline Software: Adréline Numbers and Adréline Charts (spreadsheet and graphing parts)
- Apple Computer: Apple OpenDoc Essentials—includes Apple Draw (a container draw part), Apple 3DMF Viewer (for viewing QuickDraw 3D shapes), Apple Audio (for recording and playing audio), Apple Button (for triggering actions), Apple Image Viewer (for displaying pictures saved in PICT, GIF, TIFF, or JPEG format); to be added in May: Apple QuickTime Viewer (for viewing QuickTime movies)
- Bare Bones Software: BBEEdit Lite for OpenDoc (a text editor part)
- Corda Technologies: C-Graph, C-Table, and C-TextBox (graphing, table, and text parts)
- Digital Harbor: WAV (a word processor part)
- Eclipse Services: Canopy Outliner (an outliner part)
- Hutchings Software: Rapid-I Button (a button part for triggering actions)
- Kantara Development: Kantara Internet Search Service (a Cyberdog service that allows users to use Internet search engines within Cyberdog)
- Metamind Software: Dock'Em (a part for constructing media-rich documents and presentations)
- OnBase Technology: DragNet for Cyberdog (a replacement for the Cyberdog notebook)
- SoftLinc: LEXI (a spelling checker/thesaurus part)
- theta group, GbR: LiveAccess (parts for accessing external databases)
- Totally Hip Software: WebPainter for OpenDoc (a web animation and GIF design tool)

Here are two Apple web pages that list interesting OpenDoc and Cyberdog parts:

<http://www.opendoc.apple.com/users/odproducts.html>

<http://www.cyberdog.apple.com/cyberparts.html>

object, for example. Your modified application treats this foreign content as a “black box.” All your application needs to know is the location and shape of each piece of foreign content; it then calls the appropriate routines to deal with the content—one routine to draw the content, for example, and another to handle user events that occur within the content’s boundary. You will use this same technique—that of modifying your application to handle foreign content—when you use ALOE to provide container support.

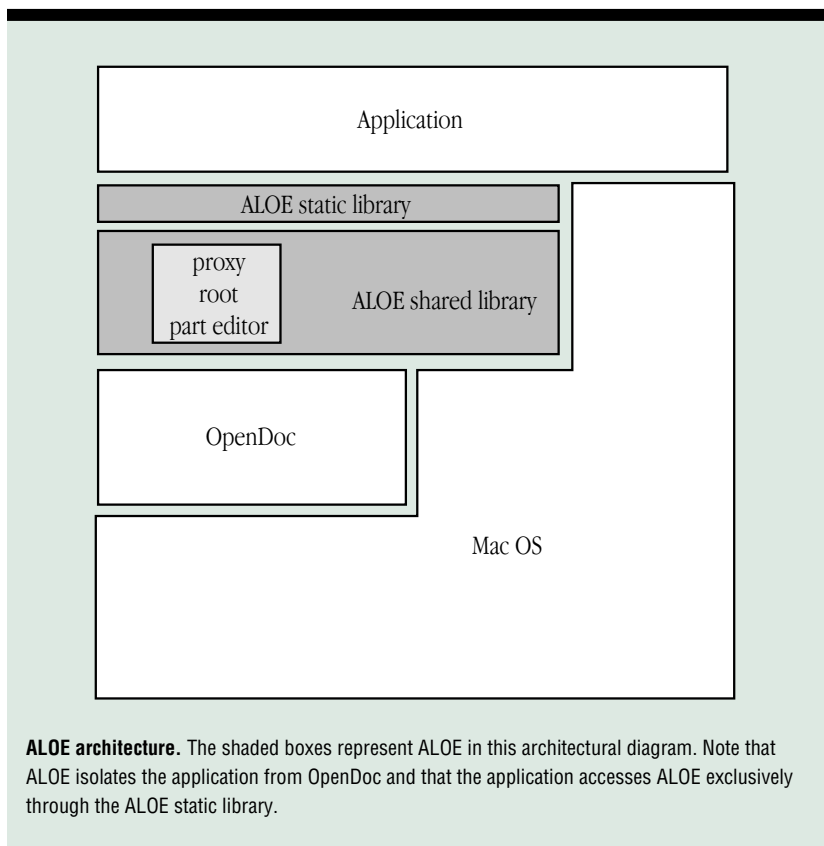
ALOE Architecture

Now that you have some idea of how ALOE works, I can talk about how ALOE fits into the Mac OS architecture. (See the figure on this page.) The two libraries that implement ALOE are the *ALOE static library* and the *ALOE shared library*.

ALOE static library. This is code that is linked with your application, and it does two things. First, it enables you to dynamically load and unload ALOE and OpenDoc under program control. This makes it possible for you to minimize the amount of memory your application uses under certain circumstances (for example, when none of the open application windows contain embedded parts) by unloading ALOE and OpenDoc.

Second, the ALOE static library provides some default behavior when your application is running on a computer where either ALOE or OpenDoc isn’t present. For example, ALOE doesn’t complicate your code by forcing you to check for its presence before calling one of two alternate versions of a routine. A number of ALOE routines are equivalent versions of Toolbox routines that you would normally call. For example, using ALOE, you would call the ALOEGetScrap routine instead of the GetScrap routine. Such ALOE routines are written so that if ALOE isn’t present, their Toolbox equivalents are called instead.

ALOE shared library. This shared library, which must reside (along with OpenDoc) on the computer that is running an ALOE application, implements most of ALOE’s functions. One component of the ALOE shared library is the proxy root part editor. As I mentioned earlier, each window that contains embedded parts has an associated proxy root part. The proxy root part editor is the OpenDoc part editor that governs the behavior of each proxy root part.



ALOE architecture. The shaded boxes represent ALOE in this architectural diagram. Note that ALOE isolates the application from OpenDoc and that the application accesses ALOE exclusively through the ALOE static library.

How Much Work Will It Take?

Of course, a key question to ask when considering implementing any new technology is, How much work will it take? In the case of ALOE, the answer depends on two factors: the structure of your application and the number of ALOE features you implement. Depending on your situation, the time needed to add ALOE support, according to one Apple engineer, is anywhere from a few weeks to a few months.

Your application’s structure. As I described earlier, ALOE integrates OpenDoc parts into a container document by allowing them to be treated as foreign content; the application doesn’t handle them directly but rather calls specific routines that know how to handle them. If you have previously modified your application to handle some other kind of foreign content (QuickTime movies, for example), you’ve already made the major structural modifications you’ll need to implement ALOE. After all, an ALOE adapter view is just another kind of foreign content.

A few applications—ClarisWorks comes to mind—already have their own private

component architecture. Since these applications allow their own components to be embedded into their documents, it should be fairly easy to use ALOE to embed OpenDoc parts.

If your application is based on MacApp, you’ll be interested to know that the next version of MacApp, Release 13, integrates ALOE into the MacApp framework code. I’m told that this integration makes it very easy for you to add container support to your MacApp application. (For details, see the box on page 6.) You can also benefit if your application is written in C++; MacApp Release 13 encapsulates most of the ALOE functionality in a set of C++ classes that you can call from your C++ program.

The worst-case scenario for using ALOE is the one where your application doesn’t know how to handle foreign content. In this case, you’ll need to make the structural modifications to your application to do so. The good news is that you only have to do it once to give your application access to many different data types—as many different data types as there are kinds of components (OpenDoc parts, Java applets, and ActiveX objects).

The number of features implemented.

As I mentioned earlier, the ALOE API is scalable. Different sets of routines provide support for various behaviors—for example, embedding, persistent storage, Undo, Clipboard, dragging and dropping, Edition Manager support, and scripting.

Apple recommends that you support those behaviors that are consistent with the way your application already behaves. For example, if your application accepts content dragged from outside itself, you should make OpenDoc parts just another kind of draggable content by supporting the ALOE drag and drop functions.

Apple recommends that ALOE applications support one particular behavior: the caching of each embedded part's static image when the enclosing document is saved. Your application may be asked to open a document containing embedded parts while running on a computer where ALOE or OpenDoc is absent. If this happens, your application can use the cached image to display the document to the best of the computer's ability.

ALOE Future Directions

The ALOE API was designed to make it as easy as possible for you to add container support to your applications. Apple believes that ALOE delivers the capabilities that most developers need for implementing container support.

Apple engineers plan on improving ALOE based on feedback from developers, and they are investigating adding features in future versions of ALOE. One such feature is linking (that is, transferring data) between the application and an embedded OpenDoc part. Apple may also allow access to *containing part properties*, another way of sharing settings between the application and a contained part.

Another possible addition to ALOE is giving applications access to the extensions of the embedded parts. This aspect of the OpenDoc design allows part developers to specify their own customized, low-level mechanisms for allowing parts to communicate with each other.

OpenDoc 1.2

OpenDoc 1.2 is a new release of OpenDoc that provides the necessary infrastructure for container support; ALOE requires OpenDoc

1.2 to work properly. For example, an improvement in OpenDoc 1.2's window management makes it possible for OpenDoc to notify an application when the OpenDoc window state changes.

In addition to changes related to container support, OpenDoc 1.2 includes the following improvements:

- OpenDoc 1.2 is more stable in low-memory conditions. In addition, if a low-memory situation occurs, OpenDoc 1.2 makes it easier for users to change a document's memory partition.

- OpenDoc looks for needed files to be stored in certain folders (for example, part editors need to be in a certain folder). Before OpenDoc 1.2, OpenDoc looked for the folder by its English name—for example, the Editors folder. This caused problems on systems with localized versions of OpenDoc, where the needed files might be stored in a folder with a different name. OpenDoc 1.2 fixes this problem by using a non-language-specific method of locating files.

- OpenDoc 1.2 provides support for Apple Guide 2.1.

- OpenDoc 1.2 includes a number of bug fixes and optimizations.

ALOE and MacApp Release 13

If your application is written using the MacApp development framework (or if you're contemplating creating a new product using MacApp), the next release of MacApp will make it very easy for you to add container support to your application.

Apple engineers are incorporating ALOE into MacApp Release 13. Much of the source code in a framework implements an application's standard behavior, so you don't have to. MacApp Release 13 integrates ALOE into the standard MacApp classes that you use to build your application. This means that you need to make fewer changes to a MacApp application to provide ALOE container support than you would to provide the same support for a conventional application.

Many details of using ALOE within MacApp—support for menus, the Clipboard, AppleScript, and event handling, for example—are already provided in the appropriate MacApp classes, thus decreasing the number of ALOE routines that you have to deal with directly. As an example of how MacApp simplifies ALOE container support, you need do only three things to enable your MacApp application to support the dragging and dropping of OpenDoc parts into your application's documents: add two functions that enable dragging and change the parent class that your document class inherits from.

If your application is written in C++, you may still benefit from MacApp Release 13. The MacApp engineers wrote a series of C++ classes that encapsulate many of the ALOE routines along with "recipe" code that you would normally have to write yourself. These C++ classes give you most of the automatic support of ALOE that MacApp Release 13 delivers, and you can call them from within your C++ application to add ALOE container support.

A developer release of MacApp Release 13 will be the E.T.O. 23 CD, due in April 1997. The final version will be released on the web in mid-1997 and will be on E.T.O. 24, due in August 1997.

Availability

Apple posted its first developer release of ALOE onto the World Wide Web in February and has since posted later developer releases to the web. You can download the current version from the web page located at <http://www.opendoc.apple.com/dev/ALOE/aloemain.html>. Apple plans for ALOE to be of finished ("golden master") quality in time for the Apple Worldwide Developer Conference in mid-May. This means that ALOE will be close to its final version by the time you read this, and you should start adding container support to your application now.

Conclusions

It's important for you to add container support to your application because doing so gets you "free" support for any content or data type that can be represented as an OpenDoc part, thereby making your application more valuable to customers and increasing its sales.

ALOE is the technology that Apple recommends you use to implement container support, for the following reasons.

- ALOE presents the simplest possible interface for delivering robust container support.

- ALOE is the result of extensive Apple research on how to provide container support, so it represents what Apple believes to be a sound and robust solution. It was also designed to support a wide range of Mac OS applications.

- You can add ALOE container support without making radical structural changes to your application, and you can do it in a reasonable amount of time—somewhere between a few weeks and, at most, a few months.

- Most existing applications can use ALOE, since it can be called from C, C++, Pascal, and 680x0 assembly-language programs.

- ALOE is an Apple-supplied technology that Apple will continue to support and improve.

A number of software companies are using ALOE to add container support to upcoming versions of existing applications. One prominent company that is doing so is Corel, which is using ALOE to implement container support for future versions of both CorelDRAW and Corel WordPerfect. According to Jon Nichols, a member of the Corel Macintosh software development team, "After looking at other alternatives, it quickly became evident that ALOE was the best solution for adding container support. ALOE is extremely clean and straightforward, and it provides all of the

functionality we'll need for complete [container] support. With the availability of ALOE, the number of OpenDoc parts available will most certainly grow."

The combination of ALOE and OpenDoc 1.2 represents a milestone in the evolution of OpenDoc: Apple has now delivered all the technologies necessary for almost every Mac OS developer to be able to actively participate in the world of OpenDoc software. OpenDoc presents you with new development opportunities and your customers with a richer, more interesting computing experience. I hope you decide to make your applications part of this exciting new world. ♣

NEWS

PowerBook 3400

continued from page 1

RAM, and lithium-ion batteries, Apple has created a high-performance PowerBook that allows you to run processor- and graphics-intensive applications on a mobile platform.

Multimedia

The PowerBook 3400 comes with a 12.1-inch active-matrix SVGA display, unique four-speaker sound system, PCI video controller with hardware acceleration, industry-standard VGA connector for 16-bit video out, 16-bit stereo in/out, zoomed video support, and software-based MPEG-1 decompression. All of these features combined make the PowerBook 3400 one of the most robust systems available today and give you the key elements you need to create and present multimedia-rich information.

Communications

The PowerBook 3400 series features a number of communications capabilities, including built-in 10 Mbps Ethernet and 33.6 Kbps modem. The innovative RJ-45 connector located off the system's backplane is designed to accept either Ethernet or telephone cable for access to network or modem lines. Asante Corporation designed the combination Ethernet and modem PCI card specifically for the PowerBook 3400 series and expects to sell it as a stand-alone product for upgrading the base configuration. Another communications capability includes IRTalk, an infrared standard for

easy, wireless file sharing that allows for communications between PowerBooks and access to a network by means of a one-megabit-per-second IrDA infrared connection. In addition, you'll now have access through the serial port to GeoPort technology for ISDN and analog modems.

Expansion

A flexible expansion bay on the PowerBook 3400 accepts a variety of modules, including a floppy drive, CD-ROM drive, additional hard drive, magneto-optical drive, internal AC adapter, and Zip drive (expected to be available in the second quarter of 1997 from VST Technologies). In addition, expansion bay modules from the PowerBook 5300 and 190 series can be used with the PowerBook 3400. For more expansion options, two PC Card slots can support two Type II or one Type III card. The PowerBook 3400 can "hot swap" modules while the system is running, unlike notebook systems that require you to shut your computer down or put it into sleep mode before you can remove an expansion bay module.

Software

To provide a complete, mobile computing solution, all PowerBook 3400 models come configured with Mac OS 7.6 and a variety of productivity, communications, and compatibility software. This software includes Claris Organizer for tracking schedules and contacts, Apple Internet Connection Kit software (including Netscape Navigator™ 3.0, Claris EMailer Lite, Fetch, NCSA Telnet, and

NewsWatcher), Apple Remote Access for easily connecting to a desktop or network from a remote location, Macintosh PC Exchange for reading and writing PC-formatted floppy disks, Macintosh Easy Open and MacLinkPlus from Dataviz for reading and translating PC-formatted files, ForeFront Group's WebWhacker 2.0 for browsing the Internet off-line, and Wynd-Mail wireless e-mail for wireless mobile communications. In addition, the PowerBook 3400 also comes with Virtual Pool from Macplay, a 3-D pool game.

The PowerBook 3400 is also the first PowerBook to be shipped with Apple Location Manager, a software utility that allows you to save system settings that are location dependent, including settings related to networking, printer configurations, time zones, file sharing, sound, auto-open items, and the Extensions Manager. Once a defined location is set up, you simply select that location and the system automatically reconfigures itself to reflect the chosen environment.

Configurations and Availability

The 180 MHz and 200 MHz PowerBook 3400 configurations are available immediately in the United States. Additionally, these configurations are expected to be available in Europe in late March, and in Japan sometime in April. The PowerBook 3400 240 MHz configuration is expected to be available worldwide in April. Apple expects demand for the PowerBook 3400 to be high and availability to be constrained through the end of March.



Apple Announces Further Details of New Organizational Structure

Apple recently announced further details of its streamlined organizational structure that consolidates the company's product development, marketing, sales, support, and operations efforts into groups sharply focused on Apple's strategy.

As part of this announcement, Apple unveiled the structure of the new Worldwide Sales and Service organization, which comprises all the geography sales organizations, as well as Claris and AppleAssist. The organization—responsible for selling and bringing service and support to customers in the education, business, and consumer markets—will focus on fulfilling customer demands and will be organized geographically, as follows:

- The North America Business Division will focus on three key markets: PEN (Publishing/Entertainment/New Media), Enterprise, and SED (Scientific/Engineering/Design), and will include NeXT Sales and Professional Services. Mitch Manditch will be acting as head of the North America Business Division.

- The North America Education Division will be focused mainly on two key markets, K–12 and higher education, and will be led by Mike Lorion.

- The North America Consumer Division, led by H. L. Cheung, will focus on two key markets: the consumer and the Macintosh installed base.

- Other geographical divisions include EMEA (Europe, Middle East, Africa), led by Jan Larsen; Japan, led by Tetsuya Shiga; Asia-Pacific, led by Steve Vamos; and Latin America, led by Luis Rubio.

- Claris, led by Dominique Goupil, maintains an unchanged charter and is responsible for the sales of Claris and Apple software worldwide.

"We are shaping our sales organization to put the responsibility for sales and support closer to our customer," said Gil Amelio, Apple's chairman and CEO. "A number of activities were previously joined with the sales organization—engineering, manufacturing, operations, marketing functional groups.

Apple's sales organization now focuses on one thing and one thing only—sales and support to serve our customers and fulfill their demand worldwide."

Apple also announced Marco Landi's resignation to pursue other interests. Landi was most recently executive vice president of Worldwide Sales and Service and was formerly the executive vice president and chief operating officer.

David Manovich, former vice president of Apple's U.S. Consumer/Retail Sales division, has returned to the company as executive vice president of Worldwide Sales and Service.



New Power Macs Pack More Punch

Apple recently introduced a new line of Power Macintosh computers that offers greater speed, performance, expandability, and affordability. This new line includes a high-performance Power Macintosh running at 233 MHz, as well as fast 200-MHz systems at prices ranging from approximately U.S. \$1,700 for a general-productivity system to approximately U.S. \$4,700 for a top-of-the-line multiprocessor system for professional graphics.

The high-end models in the new line of Power Macintosh computers feature a dramatically reengineered "tower" that makes it easy to add new components. This new, compartmentalized design allows direct access to the main logic board, making upgrades and PCI card insertions a cinch.

On the inside, you'll find faster 200 MHz processors with larger on-chip caches, more memory (up to 32 megabytes of RAM for the high-end models), high-speed CD-ROM drives (up to 12x-speed) for better multimedia performance, and an operating system that's forward-compatible with future versions of the Mac OS.

Most of the new computers are designed with the processor on an easily removable card that can be replaced with a faster processor card when Apple or a third-party manufacturer makes one available. In addition, the new Power Macintosh computers are compatible with two Apple PC-compatibility PCI cards, a 7-inch 100 MHz PC compatibility card, and a 12-inch 100 MHz Pentium PC compatibility card. With these cards, you

can run Mac OS, DOS, and Windows applications on one computer.

Power Macintosh 9600 Series

The PowerPC 9600/200, 9600/200MP, and 9600/233 computers include PowerPC 604e processors running at 200, 200, and 233 MHz, respectively, and should be popular with graphics and multimedia professionals. These computers include a new video card—the IMS Twin Turbo 128 M4A—for improved graphics performance. The PowerPC 9600/200MP offers the same capabilities as the PowerPC 9600/200 but with the addition of a second 200 MHz PowerPC 604e processor. This second processor makes it a multiprocessor (MP) computer that, when used with specially adapted software, can perform up to 50 percent faster than its single-processor equivalent.

Power Macintosh 8600 Series

The Power Macintosh 8600/200 computer includes a 200 MHz PowerPC 604e processor and is popular for media authoring. It includes such features as a 12x-speed CD-ROM drive, high-quality video input and output, a built-in Zip cartridge drive, and built-in graphics acceleration.

Power Macintosh 7300 Series

The Power Macintosh 7300/180 and 7300/200 computers include PowerPC 604e processors running at 180 and 200 MHz, respectively. These mid-range computers are suitable for small- and medium-sized businesses.

Note: All the previously mentioned computers have their processors mounted on a removable processor card and are capable of accepting faster processor cards when such cards become available from Apple or third-party companies.

Power Macintosh 4400 Series

At the low end of Apple's current offerings, the Power Macintosh 4400/200 includes a 200 MHz 603e PowerPC processor and is meant to be a general-productivity business computer.

You can find more details on each computer, suggested pricing, and worldwide availability at <http://product.info.apple.com/pr/press.releases/1997/q2/970217.pr.rel.pm9600.html>.



New Spring Line-up of Educational Products

Apple recently introduced its most comprehensive line-up of new education products in 20 years. Timed to coincide with the key purchase season for schools, Apple's new product portfolio will offer K-12 and higher-education customers a broader selection of products, with enhanced multimedia and Internet functionality at extremely competitive price and performance levels.

Apple's new education products include a new class of mobile computers, a new Power Macintosh line with higher performance and richer features, and an expanded offering of custom software solutions that not only help students learn but also prepare them for work in the real world.

eMate 300

Apple's eMate 300—an affordable mobile computer that can serve as a companion to Mac OS and Windows-based computers in the classroom, school, or home—makes computing technology available to more students than ever before.

The eMate 300 weighs less than four pounds, features a rugged "clamshell" design with steel chassis, and has a battery life of up to 24 continuous hours before recharging. It features the award-winning, easy-to-use Newton operating system and comes with Newton Works, a set of built-in applications that include a word processor, drawing program, spreadsheet, and graphing calculator. The bundled Newton Internet Enabler offers TCP/IP capabilities, giving students access to the World Wide Web and the ability to send and receive e-mail by using third-party software and an optional modem with dial-up access.

Other key features include an ARM 710a 32-bit RISC processor, Cirrus Voyager chip set, 8 MB ROM, and 3 MB RAM (1 MB DRAM, 2 MB Flash). Expansion capabilities include a PC Card slot that accommodates Type I, II, or III PC Cards (PCMCIA standard), as well as an internal RAM/Flash upgrade slot. It is compatible with a variety of serial, parallel, and network printers—both Macintosh and Windows-based—including Apple StyleWriter and PostScript printers. Available in an English-only version, the eMate 300 is scheduled to be shipped in March.

Expanded Power Macintosh Line

Apple's new Power Macintosh line is the most affordable, most comprehensive suite of desktop multimedia products that Apple has ever offered. (For more information on Apple's new line of Power Macintosh computers, see "New Power Macs Pack More Punch" on page 8.) This expanded line features three integrated all-in-one products: the Power Macintosh 5400/180, 5400/200, and 5500/225, along with two new tower systems, the Power Macintosh 6500/225 and 6500/250. (All these Power Macintosh models include PowerPC 603e processors.)

With the new Power Macintosh models, customers who want more built-in features and higher performance now have a range of options to choose from. All of the new models are multimedia- and Internet-ready and are designed to meet the needs of customers on a budget. Those who demand significant power at a competitive price can choose the high-end models. For example, the Power Macintosh 5500/225, 6500/225, and 6500/250 include PCI slots and built-in graphics acceleration. These computers are for sale to educational markets only.

Apple Education Series (AES) Bundles

To complement its new Power Macintosh line-up, Apple announced three new Apple Education Series (AES) bundles designed for secondary schools. With the addition of these new titles—Productivity Tools, Science Tools, and Math Tools—the AES product line now includes more than 20 solutions designed for specific subject areas.

You can find more details on these offerings at <http://product.info.apple.com/pr/press.releases/1997/q2/970217.pr.rel.education.html>.



The Apple QuickTake 200 Camera

Apple's Compact, Light-Weight, Easy-to-Use Digital Camera

Apple recently introduced the QuickTake 200 digital color camera, the new compact and easy-to-use camera that lets you capture, edit,

and present digital images and information—whenever and wherever the mood hits you.

With the QuickTake 200, you have everything you need to easily capture, edit, and share digital images on computers, television, or videotape. The QuickTake 200 includes all software and cables required for full operation, as well as Apple Camera Access, Adobe PhotoDeluxe, Adobe PageMill, and PictureWorks NetCard software, which collectively allow you to edit, manipulate, and electronically share images as well as create web pages.

With a look and feel similar to that of a traditional camera, the Apple QuickTake 200 is very simple to use, but its digital technology offers more functionality than any other similarly priced product. To start, the QuickTake 200 replaces the traditional viewfinder with a vivid 1.8-inch liquid crystal display (LCD) panel, which can also be used to display stored images. The Apple QuickTake 200 lets you preview a scene before taking a picture. You can also review all stored images, using the camera's "selective delete" feature to immediately delete pictures you don't want to save, eliminating the frustration of "wasting film" on unwanted images.

QuickTake 200 images are recorded on a removable solid-state storage card. The included 2 MB storage card can store 30 standard-quality or 20 high-quality color images (or a combination of both); an optional 4 MB storage card can hold 60 standard-quality or 40 high-quality color images. Both standard-quality and high-quality images use 24-bit color and are stored at a resolution of 640 by 480 pixels. The only difference is that standard-quality images are compressed more, which can result in some degradation of the decompressed image. Images are captured in JPEG format, and you can save them in TIFF, PICT, and EXIF formats.

The QuickTake 200 has an automatic white balance adjustment feature that gives you the best possible image quality for both indoor and outdoor photography. The camera has a focus-free lens with three focal ranges: close-up (3.5 to 5 inches), portrait (17 to 35 inches), and standard (3 feet to infinity).

With the Apple QuickTake 200, you get a number of flexible display options and ways to share images with other people. After you have finished capturing images, Apple's Camera Access software lets you easily download images into the Power Macintosh, Performa, or PowerBook computer. The QuickTake 200 features NTSC video-out capability, allowing

you to view your images on a television set or connect to a VCR for taping for easy distribution. The QuickTake 200 can also function as a video conferencing camera over the Internet.

The price for the QuickTake 200 for Macintosh is estimated at approximately \$600 U.S. and includes QuickTake and third-party software, a serial cable, four AA lithium batteries, a 2 MB storage card, and a neck strap. A QuickTake 200 4 MB storage card is available for an estimated U.S. \$130. The QuickTake 200 for Macintosh and additional accessories were expected to be available in North America starting March 1, 1997 and in Japan, Europe, and South America starting in April 1997.

You can find more information on the QuickTake 200 camera at Apple's digital-camera web page at <http://www.imaging.apple.com/cameras/cam-main.html>.



PowerPC Chip KO's the Pentium— Even with MMX

BYTE magazine has just documented a win for the PowerPC microprocessor over Pentium chips, even with Intel's Pro and MMX enhancements. In this benchmark study, a 200 MHz 604e PowerPC-based Macintosh computer is compared with the 200 MHz Pentium, Pentium Pro, and Pentium with MMX systems, plus one Cyrix 166+ system. *BYTE* tested five basic functions in Adobe Photoshop: Arbitrary Rotate, Unsharp Mask (default), Unsharp Mask (custom), Gaussian Blur, and RGB to CMYK.

In three of the tests—Arbitrary Rotate, Unsharp Mask (custom), and RGB to CMYK—the Power Macintosh blew away the Pentium-with-MMX system by about a factor of two. Power Macintosh was within a few percentage points of Pentium-with-MMX performance in the other two tests. It also soundly beat the Pentium products (non-MMX) on all five tests, often by factors of three, four, and even more. A great graphical representation of this data is available at <http://www.byte.com/art/9702/img/027bita2.htm>.

Developers who pride themselves on having an in-depth understanding of the technical differences between these chips shouldn't miss the article in the November issue of *c't*, a German technical journal, at <http://www.heise>

[.de/ct/Service/English.htm/9611270/](http://www.heise.de/ct/Service/English.htm/9611270/). While the article concludes with the observation that there isn't a chip that's good at everything, the author states right up-front, "The PowerPCs obviously have the better core."



Proof That the Mac OS Platform Is Alive and Well

In last month's issue of *Apple Directions*, we ran an article on Mac OS hybrids ("Hybrids—How to Tell an Apple from an Orange," *Apple Directions*, March 1997), which discussed the impact that the increase of Mac OS hybrids—and the decline of Mac OS-only software titles—is having on Mac OS presence in the retail software marketplace and the steps Apple is taking to dispel that perception. This month we've included some facts and figures that should convince even the most cynical journalist or enthusiastic Windows developer that there's still plenty of software being developed for the Mac OS platform.

- There are more than 14,000 Mac OS-compatible products listed within Apple's *Mac OS Software & Hardware Guide*.

(Source: Apple Computer, January 1997.) The list is available at <http://www.macsoftware.apple.com/>.

- There are 1,899 software titles available exclusively on the Mac OS platform. (Source: Apple Computer, January 1997.) The list is available at <http://www2.apple.com/whymac/onlymac/default.html>.

- There are 2,449 available titles that were first shipped on the Mac OS platform. (Source: Apple Computer, January 1997.) The list is available at <http://www2.apple.com/whymac/macfirst/default.html>.

- There are 2,749 Power Macintosh native software products currently being shipped. (Source: Apple Computer, January 1997.) The list is available at <http://www2.apple.com/whymac/products/default.html>.

- Among PC Data's list of the 100 best-selling software titles of 1996, 48 percent can run on a Macintosh computer or have a Mac OS version available. (Source: PC Data unit sales data, November 1996.)

- Forty percent of the titles on PC Data's top 10 games list are also available in a Mac OS

version. (Source: PC Data, November 1996.)

- Seven of PC Data's listed "Top 10 Creativity Titles" run on the Mac OS platform. (Source: PC Data, December 1996, year-to-date units sold.)

- Five out of ten best-selling titles in the desktop publishing category are Mac OS-compatible. (Source: PC Data, December 1996, year-to-date units sold.)

- The number-one best-selling title in the painting and drawing category (Adobe™ Illustrator) is Mac OS-compatible. (Source: PC Data, December 1996, year-to-date units sold.)

- The number-one best-selling e-mail software application (Claris EMailer) is Mac OS-compatible. (Source: PC Data, December 1996, year-to-date units sold.)

- Apple recently shipped its 26-millionth Macintosh computer, which will be used by one of the 60 million Macintosh users worldwide. (Source: Apple Computer, January 1997.) (For more details, see the news item "26 Million and Counting!" *Apple Directions*, February 1997, page 6.)



Apple Demos DVD Prototype

At the recent MILIA 97 conference in Cannes, France, Apple demonstrated the first DVD-ROM Macintosh prototype—a modified Performa 5400 computer—and announced support for the DVD-ROM format throughout most of the Apple Macintosh product line. DVD is a new industry standard for consumer electronics and computers that delivers vastly superior audio and video compared to current CD-ROM technology. DVD-ROMs—which look identical to CD-ROMs—can hold up to 18 GB (gigabytes) of multimedia data, including any mixture of CD-quality sound, AC-3 surround-sound information, MPEG-2 video, and computer binary data. Apple noted that the first DVD-enabled Macintosh system could appear as soon as late 1997, with several new PowerBook and desktop models expected to be shipped with DVD-ROM drives by early 1998.

For further information, see the Apple press release at <http://product.info.apple.com/pr/press.releases/1997/q2/970212.pr.rel.milia97.html>. ♣

Technology

CD Highlights System Software/SDK Editions

Human Interface Users? What Users?

Feature MacApp Release 13: A Major Update

CD HIGHLIGHTS

System Software/SDK Editions, April 1997

While we continue to reexamine the most timely and usable way to get your system software (hope you're enjoying that Mac OS 7.6 CD, by the way) for this particular month, we are shipping you the regularly scheduled System Software edition so that you have a number of localized versions of 7.5.3 and 7.5.5. (We'll get you the localized versions of Mac OS 7.6 as soon as they become available.)

On this month's edition of the System Software CD, you will find U.S. and Worldwide versions of System Software 7.5.3 for Power Macintosh 4400/200 and PowerBook 1400 computers, along with 7.5.5 for Power Macintosh 7300, 7600, 8600, and 9600 computers, and versions of the Mac OS 7.6 universal installer for international English and U.S. English.

In addition to the new system software, you can find the new and revised SDKs described in this article in the "What's New" folder on SDK discs 1 and 2. All other new and revised content (like the Sample Code and Snippets updates) are in the "What's New" folder of the System Software CD. (We've divided things up in this manner to best utilize space.) Check it out!

Apple Location Manager

The Apple Location Manager (ALM) software makes it easy for users to move from one location to another without having to spend a lot of time manually reconfiguring their computer with every move.

The Apple Location Manager software is developer extensible, so if your software has settings or preferences that might change as the user moves from place to place, the ALM SDK gives you the tools you need to write an ALM module for your software. The simplest modules can be created using nothing more than ResEdit.

The ALM software also provides APIs that a developer can use to set the current location

or to register to be notified when the location changes.

The ALM SDK contains everything you need to write a Location Manager module, including a working sample, CodeWarrior and Symantec projects, and an MPW Makefile. Also included is ALM 1.0.1b1, both debug and non-debug builds.

ColorSync

The ColorSync 2.1.2 software is a second-generation color management system that offers prepress-quality color matching and separations. ColorSync 2.1.2 offers substantial benefits for developers. Now there's a standard architecture and profile format for color matching on the desktop that delivers the solution that users have asked for: No more decisions about which application programming interface (API) and profile format to use, and no more time wasted on writing custom routines. The ColorSync 2.1.2 API provides the most powerful "plumbing" for color management.

The features of the ColorSync 2.1.2 API are as follows:

- High-quality, high-performance transformation engine
- Expanded, compatible architecture
- ICC profile compatibility
- Support for PostScript Level 2 devices
- Support for more than four color devices
- Increased application support

Developer Notes

Developer notes give descriptions of new hardware and software features, comparisons with existing CPUs, and expansion card design information. This month's System Software CD features a new developer note that describes the Power Macintosh 4400 computer, which is a new Macintosh model that uses a logic board based on the Apple Logic Board Design LPX-40.

Disk Copy 6.1 GM

The Disk Copy 6.1 GM application will mount disk images on your desktop, make exact copies of floppy disks from a disk image, convert disk images from one format to another, and create a disk image from a mounted volume or individual folder. Using DiskScripts and AppleScript scripts, you can use Disk Copy 6.1 GM to automate software installations and disk image manipulation.

Disk Copy 6.1 GM features extensive Balloon Help and supports Drag and Drop (which requires System Software 7.5 or later), AppleScript, PlainTalk Text-to-Speech, and DigiSign digital signature software.

Disk Copy 6.1 GM requires System Software 7.0.1 or later and is a "fat binary" for both PowerPC and 68K-based Mac OS-compatible computers; A/UX and Macintosh Application Environment (MAE) are not supported.

For more information, see the *Disk Copy 6.1 Manual* or Quick Reference documents included with this package.

Installer SDK Cornucopia Version 1.0.2

As before, the goal of the Installer SDK Cornucopia software is to provide the latest and greatest Installer tools and helpful utilities, as well as to provide a look at where we're headed with the tools and the Installer itself. The new SDK update, version 1.0.2, contains some tool updates and bug fixes. In addition, it includes all the regular great stuff—updated contents from the previous SDKs, in addition to tools and tidbits, a prerelease version of the Installer Engine 4.1 software, and the final version of the Installer 4.0.6 software.

The features of this SDK update are:

- Installer application: released and prerelease versions
- Installer tools: compression and Installer scripting tools

please turn to page 15

Users? What Users?

By Peter Bickford

I've recently received several letters like the following one:

Dear Doc,

Our company has been developing a product for <name of technology>. We believe it will open up new vistas in the world of personal computing and change the world as we know it. I've enclosed screen shots to give you a better idea of our product. Please treat this information as extremely confidential, etc. etc. etc.

What we wondered was, what sort of interface should a product like this have? We figured you're the right person to ask since you seem to be Apple's resident font of Human Interface Knowledge. Any help you could give us would be greatly appreciated.

Sincerely,

<name of company>

Yes, I really do get letters like this, and I really did get called Apple's "resident font of Human Interface Knowledge." To be honest, though, I usually find some excuse to put off answering letters like these. There are three reasons for this:

1. I don't consider myself Apple's resident font of Human Interface Knowledge. (I'm more of a Zapf Dingbat, or maybe a Zeal.)
2. Everyone knows that the only Apple resident fonts are Chicago, Geneva, and Monaco.
3. I have no clue what the answer is.

The problem is that I know nothing at all about the people who are going to use this product. I don't know what they're like, how they do their job, or how they'll want to use this wonder of technology to do their job better. Without this knowledge, it's impossible to design a good interface.

It sounds like a cop-out, but even the simplest design decisions become a problem if you don't know much about the product's users. Take the issue of whether to use a pop-up menu or a set of radio buttons to let the user choose from a list of choices. For example, users of a kiosk system I designed had no idea how to use a pop-up menu, because they were unfamiliar with the concept of "clicking and dragging." When I replaced the pop-up with a set of radio buttons, the problem went away. However, had I been able to assume basic Macintosh knowledge from my users, I would have stayed with the pop-up menu. It all depends on who your users are, and what they can be expected to know.

Certainly there are some things you can guess, simply by knowing that the users are human beings. The core principles that underlie the Apple Human Interface Guidelines are grounded in basic human

psychology. So when we say that it's good for your application to be "consistent" or to have "aesthetic integrity," it doesn't really matter whether you're programming for Macintosh or DOS, or whether the audience consists of Irish accountants or Indonesian construction foremen. Simply because you know that the user is human, you know that some things will tend to make the application easier to learn and use.

On the other hand, it's hard to know which tradeoffs to make without knowing your user's needs. Is it OK for the product to be a bit harder to learn if, once learned, it's more efficient to use? Will warning users of dangerous situations anger those who don't want the computer to "second-guess" them? Where do the ideas of user control and perceived stability fit in when a central data processing department wants to exercise control over what software people have installed on their machines?

Determining What Your Target User Is Like

Although many different types of people may eventually use your product, you need to focus on one, or a few, "target users." These are the people whose needs you'll design your product around. A good product will be usable by more than these target users, but the target users are the ones you'll want to concentrate your efforts on.

In general, the more targeted your product is, the more successful it will be. You should resist the temptation to say that "my product is for everyone." By taking that approach, you'll either be fooling yourself, or you'll end up designing a product with such a hodgepodge of features that it will wind up serving the needs of no one. So whether your audience is "mid-level managers who need to schedule time more efficiently" or "comic book collectors with medium to large collections," you need to pick your audience and design with them in mind.

Once you've determined who your target users are, the next step is to draw up a user profile. A user profile is a collection of facts you know about your users, including answers to the following questions:

- What do your users know about their area of expertise?
- What terms and concepts from that area of expertise are the users familiar with?
 - What do the users know about Macintosh computers?
 - What computer terms and concepts are they familiar with?
 - What sort of environment do they work in?
 - What are their jobs?
 - What factors are important in those users being successful at their jobs?
- Are there special physical, social, cultural, or other considerations that I need to be careful of when designing for those users?

The user profile for a purchasing program might read (in part):

- Users are extremely proficient with purchasing tasks (dealing with stock levels, purchase orders, and so on).
- Users are comfortable with and expect to use specialized terms like PO (purchase order), FOB (freight on board), MRP (manufacturer's retail price), and so on.

- Users have some Macintosh experience, but this program may be one of the few they use. However, basic Macintosh skills can be assumed.

- Users are familiar with very simple Macintosh terms (*click, system*), but not more technical terms (*file system, partition*).

- Users work in an office environment with open cubicles and phones. They're frequently diverted by interruptions.

- Users' main concern is the successful placement and tracking of purchase orders.

- Accuracy and speed are both important. Information on the state of purchase orders must also be readily available.

- Several social levels are associated with different types of purchasing work. Higher-ups are also expected to keep tabs on the work of those below them in the organization.

The more you know about your target users, the better you'll be able to make your design decisions. And when the inevitable tradeoffs come, you'll have the information you need to know where the balance should lie.

Gathering the Information

You can begin to gather the information for your user profile by visiting with your marketing department. You might have noticed that a user profile looks suspiciously like a marketer's customer profile. The difference is that the marketer wants to know how the product should be sold, while you want to know how the product will be used. You'll still gain a lot of useful information by learning the customer profile inside and out.

You can also gather information about your customer's world by reading trade magazines and attending industry seminars and trade shows. Be wary, however, that what these really give you is other people's views of the needs and concerns of your target users. Often, these coincide with the products that the same people are selling into the industry.

By far, the best way to gather information is to spend time with representative users, watching them do their jobs. It's then that you'll

learn the culture and the subtle quirks of the jobs that make all the difference. For instance, a developer of a meeting-scheduling program should realize that in many companies, managers are free to schedule their employees' time, but the reverse is taboo. Or any retail system should allow for flexible, even sloppy ways of handling inventory tracking in a situation like the local comic book shop. There, the time and energy required to practice rigorous inventory control might far outweigh the benefits.

You Are Not the User

If there is one thing you must not do, it's to assume that you don't need to worry about all this because you look at the target user as just a slightly less clever version of yourself. To paraphrase Larry Tesler, former head of Apple's Advanced Technology Group and now the company's chief scientist, we may not know who the user is, but we know it isn't us. Clip this out and tape it to your monitor, staple it in the family Bible, or scan it in and use it as a start-up screen. Just don't ever forget it when you design a human interface. Nobody who wants to have a successful product can afford to forget this. We've known for a long time that programmers

who design for themselves usually wind up selling into a market of one. But even veteran human interface designers need to remember that the mere fact that we have spent so much of our lives around computers makes us see things very differently than do most people.

Although we see ourselves as the representatives of the user, we are in constant, mortal danger of losing our ability to see things through the user's eyes. Without that perspective, we stand as much chance of designing a good interface as we do of winning the lottery.

*Till next time,
Doc*

Peter Bickford, a human interface senior scientist in Apple's Developer Consulting Group, is temporarily away pursuing other duties. This month's article is a reprint of his August 1993 column. You can send him your interface questions and comments at bickford@apple.com.

**We may not
know who
the user is,
but we know
it isn't us.**

MacApp Release 13: A Major Update

By *Michael D. Rossetti*
MacApp Engineering Manager

The MacApp engineering team has been pretty busy these last few months preparing for the upcoming MacApp Release 13. This tremendous effort will shortly result in what could be considered the most significant enhancement of the MacApp framework since it was first written. The only previous change that perhaps comes close would be the switch from Object Pascal to C++ as the language in which MacApp was written.

The final version of MacApp Release 13 will be a well-designed, modern, mature application framework that uses the most appropriate and modern programming technologies available. Here is a list of some of the major enhancements implemented in MacApp for delivery in Release 13:

- Increased usage of “modern” C++ language features, such as native exceptions, better use of constructors and destructors, templates, and the C++ Standard Template Library
- More modular design, removal of TObject, and creation of the Apple Class Suites (ACS)
- Network support classes
- Thread support classes
- New and improved view classes
- Implementation of OpenDoc container application support using the ALOE library
- Improved view editor (shipped earlier as Ad Lib version 2.6)
- New and improved debugging aids

“Modern” C++ Usage

“Modern” C++ usage provides many benefits to developers who use the MacApp application framework. Primarily, it allows far greater compatibility with third-party class libraries. It also takes advantage of the fact that programming skills have become more common in the industry. And further, it eases the transition to new and emerging OS technologies and to other popular operating systems.

Implementing “modern” C++ language features for Release 13 will include the following:

- Native exceptions
- Replacement of I-methods with constructors
- Implementation of destructors

- Use of templates, where appropriate
- Use of the C++ Standard Template Library
- Use of the C++ Standard Library, where appropriate

Modularization

The new modular version of MacApp enables developers to customize their applications, because they can have tighter control over what components of MacApp will be included in their applications. Previous versions of MacApp essentially forced the developer to use almost all of the framework, even when much of what was used was not necessary in the resulting application.

The modularization of MacApp consists of two basic activities: the removal of the TObject class, a base class from which all MacApp classes formerly descended; and the creation of a set of class suites called the Apple Class Suites, separate from MacApp, that provide a number of classes that can essentially stand on their own (that is, they can be used with or without the rest of MacApp).

Network Support Classes

One of Apple’s stated objectives is to provide ubiquitous Internet and networking capabilities in its products. The implementation of network support classes in MacApp provides a solution to that objective that is easy and compelling for developers to use.

The new networking support is a provider-independent, asynchronous event and networking subsystem, complete with timers, transaction queues, state machines, and parsing, all with a scalable and simple, powerful, object-oriented API. The current release is aimed at making Internet-aware applications and will be shipped with support for TCP connections and Domain Name Resolution services, using either Open Transport or MacTCP. Future plans include high-level protocol support—such as support for FTP (File Transfer Protocol), SMTP (Simple Mail Transfer Protocol), POP (Post Office Protocol), and HTTP (Hypertext Transfer Protocol)—as well as support for other types of networking, such as AppleTalk.

The new suite of classes, called the ACS Network Suite, is one of the new Apple Class Suites.

Thread-Support Classes

Built-in support for multithreaded processes is one of the MacApp features most frequently requested by developers. This new feature gives you support for cooperative and multiprocessor threading, allowing you to bring the power of the Thread Manager and multiprocessor support to your application with greater ease than ever before.

In MacApp Release 13, the thread-support features straightforward and consistent classes for creating Thread Manager threads and multiprocessor (MP) tasks, as well as an efficient, intuitive synchronization abstraction based on the monitors and conditions model.

Synchronization classes enable you to transparently synchronize between threads and MP tasks. In addition, these classes automatically handle many of the other challenging aspects of the thread and MP application programming interfaces (APIs). The basic abstractions of this suite were designed to be cross-platform and have already been implemented for the Mach microkernel. The ACS Thread Suite is one of the new Apple Class Suites, but basic support for providing time to threads is also built into MacApp, further simplifying development.

New and Improved View Classes

MacApp Release 13 adds several new view and view-related classes and improves existing view and view-related classes. The new Apple Grayscale Appearance specification is expected to become the standard for Tempo (the next major release of the Mac OS, due in July 1997); a large number of newer Macintosh applications already attempt to follow this specification. Apple has licensed Trygve Isaacson’s Gray Council classes and incorporated them into the Apple Class Suites and MacApp Release 13. MacApp applications can seamlessly use the grayscale appearance in its standard view controls as well as in a number of new views (sliders, tab panels, and so on).

A number of core suites borrow code previously available only inside the MacApp framework and put it into a form that can now be used with any other framework. For example, the ACS Imaging Suite provides an initial subset of classes related to imaging. These include C++ abstractions for the

following: short and long point and rect types, a region type, a temporary region type, an RGB color type, a graphics device iterator class, and temporary drawing environment classes. The ACS Imaging Suite defines QuickDraw-derived types but does not attempt to replace QuickDraw or any other imaging model.

Also, the ACS Files Suite provides a file and FSSpec class, along with some standard file type definitions and resource utilities. The ACS Toolbox Suite provides miscellaneous C++ abstractions for Toolbox types and APIs.

OpenDoc Container Application

MacApp Release 13 enables a MacApp-based application to embed OpenDoc parts by implementing various specialty classes that use the ALOE library. For more information on ALOE, see the article “Delivering Container Support With ALOE” on page 1 of this issue.

Debugging Aids

The new Apple Class Suites now includes TidyHeap, a tool for tracking and verifying dynamically allocated memory in C++ applications. TidyHeap started out as a quick-and-dirty way to keep track of undeleted objects in the

MacApp framework and has evolved into a tool that you can easily use when writing application software in C++, regardless of what framework you’re using. TidyHeap itself is a small library of C++ classes that intercept and track all allocations and deallocations that go through the “new” and

While MacApp Release 13 will require you to modify existing MacApp applications, it goes a very long way to providing many of the new features that MacApp developers have requested.

“delete” global operators. It does this invisibly; you don’t need to do anything but call the “new” and “delete” operators in the way you usually do.

Summary

MacApp Release 13 will first appear on the web at <http://devtools.apple.com/macapp/> in an early developer release form. Apple will continue posting versions of Release 13 until it becomes final in mid-1997. MacApp will also continue to be included on the E.T.O. series of development tools on CD-ROM. A developer release of MacApp Release 13 will be on the E.T.O. 23 CD, due in April 1997.

While MacApp Release 13 will require you to modify existing MacApp applications, it goes a very long way to providing many of the new features that MacApp developers have requested. We hope you will use it and give us feedback on how it meets your needs; the best way to contact us is to send e-mail to mabugs@devtools.apple.com. Also, let us know what features you would like to see in future versions of MacApp. ♣

CD HIGHLIGHTS

System Software/SDK Editions

continued from page 11

- Tidbits: nifty stuff for Installer script development
- Up-to-date Installer developer interfaces
- Many Installer script examples

Sample Code Update 2/97

The Sample Code Update 2/97 package includes the following sample code examples, which will be rolled into the Sample Code folder on the May 1997 Tool Chest edition.

- CPUgestalt—sample code that illustrates the way to determine the processor type of the Macintosh you’re running on.
- EditTextCDev—a sample Control Panel device (cdev) that demonstrates how to use the edit-related messages and how to implement an editText item in a cdev.

• OTDumpPortRegistry—a very simple sample that dumps a text version of the Open Transport port registry to stdout. The compiled executable also makes a handy debugging tool.

• SillyBalls—a very simple sample program that demonstrates how to use the Color QuickDraw software. Consisting of about two pages of code, it does nothing more than open a color window and draw randomly colored ovals in the window.

Word Services SDK 1.0.8

The Word Services SDK 1.0.8 allows any application to link to a spelling checker, grammar checker, or other text service as if it were a built-in menu item. This SDK is a public protocol; no license fee or nondisclosure is required for using it.

The Word Services SDK includes the complete protocol specification, tutorial documents, and the source code to Writeswell Jr., a simple Word Services word processor. Also

available from the Word Services web page <http://www.wordservices.org/Developers/> is the source to WSI NewsWatcher, a version of NewsWatcher that allows you to spellcheck your posts.

The HyperCard application supports the Word Services SDK, so you can spellcheck stacks as you develop them, and allow your users to spellcheck text that they enter themselves. A simple XCMD, which is provided by the web page, is needed to initiate the spellchecking session.

The updated SDK includes bug fixes to Writeswell Jr.’s scrolling, plus the Spellswell Plus 2.1 Demo.

A list of client (word processor) and server (spelling and grammar checker) applications is available at <http://www.wordservices.org>.

*Meredith Best
Developer CD Leader and Online Content
Librarian*

Business

Marketing Feature

Running Direct Mail Campaigns by the Numbers:

An analytical model for helping software developers predict direct mail response rates, costs, and revenues.

Running Direct Mail Campaigns by the Numbers

By Laurie Beasley, Beasley Direct Marketing

Most software companies live or die by how well they run their direct mail campaigns. Direct mail is the way they notify their installed base of new products. And for many software companies, direct mail is a primary marketing vehicle for getting new customers.

One challenge to selling a new software product through direct mail is timing. Just when a marketing team should be spending “quality time” on a direct mail plan, members are typically juggling a dozen other critical product launch activities. There are press tours to go on, product reviewers to contact, and marketing literature to produce.

This article eases the burden of this hectic time by providing you with analytical tools to help you more quickly and accurately formulate direct mail plans. Based on 10 years of experience in selling software through the mail, I've developed charts and spreadsheets that can help you

- Predict response rates and order arrivals
- Estimate marketing and fulfillment costs
- Price direct mail products and incentives
- Forecast revenues

By adapting these numbers to your own situation, you'll hopefully have a more reliable basis on which to run a profitable direct mail campaign.

[Editor's Note: Though the response rates and cost estimates presented in this article are derived from United States-based direct mail campaigns, the direct mail principles and spreadsheets can be easily adjusted to work in most any country.]

Predicting Response Rates

The most important metric for measuring the effectiveness of a direct mail campaign is the *response rate*—the percentage of people who respond to a given product offer. Anticipating a response rate is key to strategizing a

profitable direct mail campaign, but this can be somewhat daunting if you've never done it before. Luckily, the behavior of people who purchase products by mail is surprisingly predictable. By making a few informed assumptions and conducting a few test mailings, you can estimate how many responses you'll receive from a mailed promotion with a reasonable degree of accuracy.

In my experience, software companies can expect 5 to 25 responses for every 100 upgrade direct mail pieces sent out (in other words, a 5 to 25 percent response rate). European response rates are equal to or higher than U.S. rates. And for some mysterious reason, Canadian mailings often yield responses 10 to 15 percent higher than mailings in the United States. I've also found that Mac OS customers are more likely than Windows users to purchase software through the mail.

Many of the factors that influence response rates are out of your control. For instance, your responses can be lower if your company has spaced product revisions too closely. On the flip side, your responses can be higher if an operating systems vendor releases a major upgrade during the same time you're shipping a new product release for that platform.

There are, fortunately, a few “tricks of the trade” that you can use to increase your response rate; and since the cost of producing and mailing a direct mail piece is pretty much a fixed cost, every additional response that you can squeeze out of a campaign increases its profitability.

Improving Response Rates

Through many years of managing direct mail campaigns, I've learned a lot about what works well and what doesn't. Here are some tried-and-true ideas that consistently seem to increase direct mail response rates.

- *Use a mailpack, not a self-mailer.* You'll cut your response rate in half if you send out your offer in a *self-mailer*. (Typically, a self-mailer is a folded, three-panel brochure with a stamp or postal mark applied directly to the outside.) A *mailpack*, which is an offer mailed inside an envelope, works better for several reasons. It gives you more surface area for compelling copy and screen shots. It's easier to add a business reply envelope and *upsell* product offers inside the envelope. (Upsell offers are products pitched along with your primary product offer.) Also, people tend to read a mailpack with a well-written letter before they'll read a self-mailer.

Rule-of-Thumb Upgrade Response Rates for Multiple Mailings

By mailing to a single group of installed-base customers three times, software vendors can almost double the response rate for a given upgrade offer.

Mailings to one list of customers	Response equation	Example
Mailing #1	$X = \text{initial response rate}$	10.0%
Mailing #2	$X * 0.5$	5.0%
Mailing #3	$X * 0.25$	2.5%
Total for offer	$X + (X * 0.5) + (X * 0.25)$	17.5%

- *Send out multiple mailings.* One of the most effective ways to maximize direct mail revenues during a software upgrade campaign is to *mail to a list of customers multiple times*. All of the big software companies do it—from Intuit to Claris Corporation—because this strategy really works. It keeps your company’s name at the forefront of customers’ minds, and often the second or third hit is enough to win a sale. To see just how effective this tactic can be, look at the table on page 16. By mailing to your installed base three times, you can almost double your response rate. This is important if you’re trying to gain market share. And it’s a way to enhance the profitability of a mailing, since you can significantly reduce your per-piece mailpack costs if you print all of your mailpack materials at the same time. (Any mailing-specific messages, such as “Last chance to upgrade,” can be overprinted in black on preprinted mailers, using a less expensive one-color press.)

- *Add a purchase incentive or premium.* You can easily increase your response rate by adding a compelling incentive or bonus to your offer. Some premium ideas that have

worked well for me in the past include computer games, product accessories, template packs, and complimentary products. Here are some other ideas:

- Price discounts for recent customers.*

Most software companies recognize the long-term value of installed-base upgrade sales, and they reward loyal customers with discounts. Offering these discounts encourages your users to upgrade early and often.

- Money-back guarantee trial periods.*

For software products, I’ve found that offering customers a money-back guarantee provides them with a comfort factor that boosts sales. (Ninety-day trials seem to work best.) The rewards of making this type of guarantee far outweigh the risks: Software companies usually see product returns of only about 2 percent.

- *Offer a giveaway to generate more leads.*

For products with retail prices above \$500, you may opt for a two-step direct mail program: An initial mailing that yields a qualified list of leads, and then a follow-on mailing of more detailed product information. Here are a few

incentives for generating more of these qualified leads:

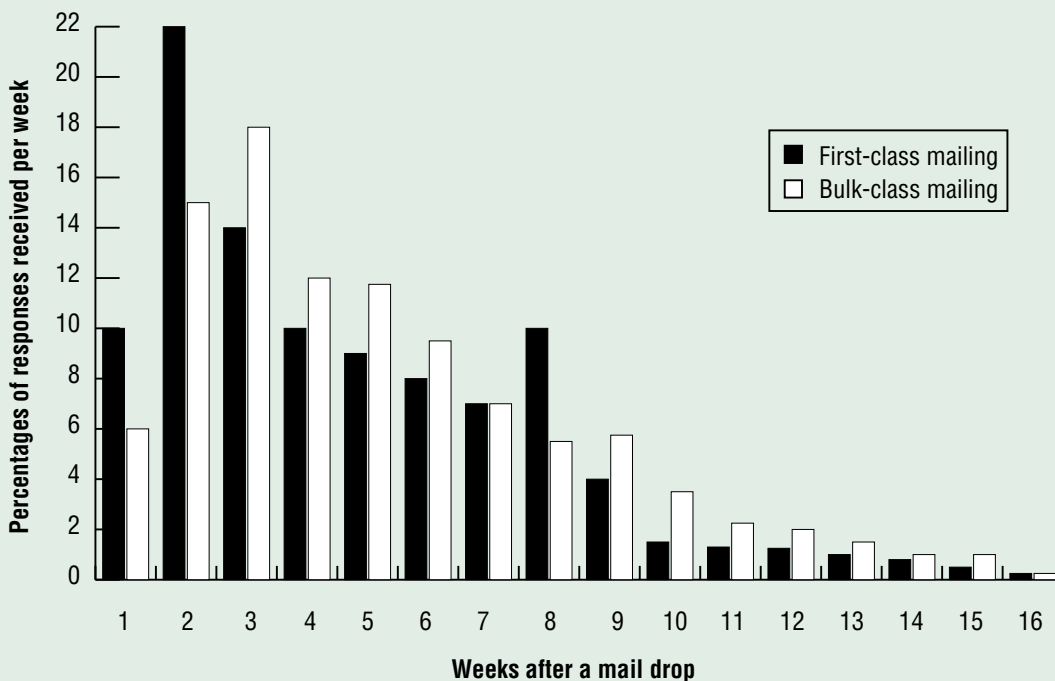
- Free white papers on hot industry topics.* This approach works particularly well for complex, high-end products. One example: An intranet database software vendor looking for qualified prospects could offer a free research report on something like “Setting Up Intranet Security Firewalls.” This type of offer can open the door to future product sales too, if the report recommends one of the developer’s other products.

- Sweepstakes or drawings.* For lead generation, contests can potentially boost response rates by several points, and they can also be used to draw prospects to your web site. You should, however, check on the legal ramifications of running these contests internationally, since each country has different laws regarding contests.

- Demo disks and videos.* Creating time- or feature-limited demos of other products in your line is a great way to boost the long-term effectiveness of a direct mail campaign.

When Orders Will Arrive

Since you’ll receive almost half of your anticipated orders in the first three weeks, consider staggering your mailings to ease the load on order takers.



Source: Beasley Direct, 1997

Platinum Software, a developer of high-end financial software, found that offering a free video of customer testimonials increased its lead response rate by 0.5 percent over previous mailings.

Estimating Costs of Direct Mail Campaigns

Once you've "guesstimated" a realistic response rate for your offer, it's time to build a break-

even model of your promotion by rolling in some intelligent estimates on costs. In the two tables "Direct Mail Cost Assumptions" and "Cost-Per-Piece Estimates" (at right) you'll find some cost assumptions that I use in my profitability models, as well as per-piece estimates for mailpacks. Because of the economies of scale inherent in printing four-color materials, per-piece costs drop dramatically as you increase the quantity of your mailings.

Assumptions for Direct Mail Analysis Spreadsheet

The sample "Direct Mail Analysis Spreadsheet" (in the box below) can provide you with a starting point for designing your own direct mail program. A Microsoft Excel version of this spreadsheet accompanies this article on the *Apple Directions* web page, at <http://www.devworld.apple.com/mkt/adtop.shtml>.

Suggested Retail Price (SRP)	\$99.95
Shipping and Handling Costs.....	\$10.00
Product Returns to Manufacturer	2.00%
Product Cost of Goods (COG).....	\$10.00
Premium/Incentive COG	\$5.00
Upsell Product SRP	\$29.95
Upsell Product COG	\$7.50
Order Taking.....	\$10.00
Credit Card Processing	3.00%
Upsell Rate (% of Total Orders).....	15.00%
Per-Piece Marketing Cost	\$1.20

Direct Mail Cost Assumptions

Costs, return rates, and upsell percentages observed in typical software direct mail campaigns:

Shipping and handling fulfillment	\$5-\$10
Product returns to manufacturer	2%-5%
Premium/incentive offer costs	Under \$5
Upsell response rates, per buying customer	10%-50%
Credit card processing (American Express is higher)	3%
Postage, per piece	\$0.32
Lists, per name	\$0.15

Cost-Per-Piece Estimates

Here's what you can expect to spend for a typical four-color brochure and mailpack, including the cost of hiring an outside creative agency, list rental, printing, and postage.

Quantity (in 1,000s)	Cost per piece
50 to 75	\$2.50-1.50
75 to 150	\$1.50-1.15
150 to 200	\$1.00-0.80
200 to 500	\$0.80-0.60

Direct Mail Analysis Spreadsheet

Description	Mail Quantity	Proposed Orders	Proposed Response Rate	Projected Base Revenue	Projected Weighted Revenue	Revenue Less Returns	Base COG	Weighted COG	Order-taking Costs
Scenario 1	100000	5000	5.00%	\$499,750	\$522,213	\$512,218	\$75,000	\$80,625	\$50,000
Scenario 2	100000	10000	10.00%	\$999,500	\$1,044,425	\$1,024,435	\$150,000	\$161,250	\$100,000
Scenario 3	100000	15000	15.00%	\$1,499,250	\$1,566,638	\$1,536,653	\$225,000	\$241,875	\$150,000
Scenario 4	100000	20000	20.00%	\$1,999,000	\$2,088,850	\$2,048,870	\$300,000	\$322,500	\$200,000

Spreadsheet Key

Term	Definition
Mail quantity	QTY = The number of mailers to be sent
Proposed orders	ORD = An estimate of the number of orders that will come in
Proposed response rate	RESP = The estimated percentage of people who will respond to a mailer with orders
Projected base revenue	SRP * ORD = Gross revenues from the mailing
Projected weighted revenue	(Projected base revenue) + (ORD * upsell rate * upsell SRP) = main product revenues + upsell revenues
Revenue less returns	(Projected weighted revenues) - (ORD * %returns * SRP)
Base COG	ORD * (product COG + incentive COG) = base cost of goods
Weighted COG	(Base COG) + (ORD * %upsell * upsell COG) = main product cost of goods + upsell cost of goods

If you want to be on the conservative side when estimating costs, figure that each order that you take costs about \$10 in overhead and labor. This factors in the cost of setting up an 800 number, using an outside order-taker, and ordering reports. Since outside order-takers typically charge you by the call—even if a prospect just wants a question answered—one cost-saving strategy is to direct first-time callers to your in-house product information line. Customers who are ready to purchase your product can then be forwarded to the more expensive outside telemarketers.

In pricing your upsell offers, there's an interesting buying behavior that I've observed: Once a customer has a credit card or checkbook out, that customer typically has very little price sensitivity in buying a second item. Customers are just as willing to pay \$99.95 as \$29.95 for an upsell product. The only limitation that I've found is that your upsell product has to cost less than the primary product you're selling.

Preparing for Order Arrivals

The timing of order arrivals is practically a constant in the world of direct mail. This information is extremely useful in forecasting when revenues will flow into the company and in figuring out staffing requirements for order taking. These forecasts should also be passed along to manufacturing so that they'll build enough product to keep the supply lines full.

Looking at the order response curve on

page 17, you can see that if you mail your mail-packs at first-class rates, about half of all orders will arrive within the first three weeks. Response rates will be lower if you send out your offers by means of bulk mail or during holiday periods (November 15 through December 31 or May 15 through August 30). To even out the load on your order takers (and to keep customers from being put on hold too long), I encourage clients to consider mailing promotions in three separate batches about three weeks apart.

I should also mention something important about taking orders from your web page: You may depress your response rate and frustrate customers if you direct prospects to your home page URL and then bury an order form several levels beneath the main page. I strongly recommend that you set up a separate URL for product promotions for two reasons: it improves customer satisfaction and enables you to more accurately track your web responses.

A Spreadsheet for Direct Mail Decisions

One of the best ways to analyze your direct mail options is to build a worksheet such as the "Direct Mail Analysis Spreadsheet," which begins on page 18. (You can download this sample Excel spreadsheet, which accompanies this article on the *Apple Directions* web page, at <http://www.devworld.apple.com/mkt/adttop.shtml>, and then modify it for your own use.)

To begin your analysis, go to the "Assumptions for Direct Mail Analysis" chart on page 18, plug in your target suggested retail price (SRP), enter zeros in the upsell and incentive cells, and overwrite any overhead costs that are different from my assumptions. Enter mailing quantity estimates into the first cell of the spreadsheet, and then plug in different response rates until you get a near-zero number in the profit column to the far right. This is your *break-even response point*, and it provides you with a baseline for analysis.

For example, if your mailing is profitable only with a 75 percent response rate, then something about your promotion needs to change. You should raise your SRP, offer an incentive, or have a telemarketer upsell other products to upgrading customers. Use the other rows to test and compare these types of options.

Though the data presented in this article should provide you with a pretty good starting point for a direct mail promotion analysis, I always recommend that clients test assumptions with limited-quantity test mailings. My rule of thumb for sizing a statistically valid test mailing is this: Send out a quantity of mailers that will yield you 100 orders. For example, if you think that your offer response rate will be about 10 percent, send out a test mailing to 1,000 customers. If you run a couple of these trials, you can also test other variables in your offer letter, such as incentives and various pricing levels.

The other bit of advice that I always give is to *keep good statistical records of all of your mailings*. Historical data on the behavior of your specific customer base can greatly improve your ability to run profitable mailings.

This is an exciting time for direct mail.

Database technology has dramatically improved the ability of direct marketers to target their mailings. And the power of corporate web pages to collect qualified leads is opening up a whole new range of direct mail possibilities. Build your direct mail spreadsheet, test it, and then run your profitable direct mail campaign by the numbers. ♣

Credit Card Charges	Marketing Costs	Total Costs	Total Cost Per Piece	Profit
\$15,666	\$120,000	\$266,291	\$2.66	\$245,926
\$31,333	\$120,000	\$412,583	\$4.13	\$611,852
\$46,999	\$120,000	\$558,874	\$5.59	\$977,778
\$62,666	\$120,000	\$705,166	\$7.05	\$1,343,705

Spreadsheet Key (continued)

Term	Definition
Order-taking costs	ORD * order-taking costs
Credit card charges	(Projected weighted revenue) * credit card %
Marketing costs	QTY * (per-piece marketing cost)
Total costs	Weighted costs + order-taking costs + credit card costs + marketing costs
Total cost per piece	(Total costs)/QTY
Profit	(Revenues less returns) – (total costs)

Laurie Beasley (lbeasley@ix.netcom.com) is the president of Beasley Direct Marketing, based in the San Francisco Bay Area. Her firm specializes in providing technology clients with big-agency expertise in direct marketing and advertising at a small-agency price. Beasley Direct Marketing has won Pioneer and Caples awards for direct marketing campaigns, and her client list includes Apple, Claris, Bay Networks, Platinum Software, Software Publishing, and more.

Internet Resources for This Issue

News

- PowerBook 3400—http://powerbook.apple.com/phase_2/show.qry
- New Power Macintosh computers—<http://product.info.apple.com/pr/press.releases/1997/q2/970217.pr.rel.pm9600.html>
- New education products—<http://product.info.apple.com/pr/press.releases/1997/q2/970217.pr.rel.education.html>
- QuickTake 200 camera—<http://www.imaging.apple.com/cameras/cam-main.html>
- Graphical representation of *BYTE* magazine test results on PowerPC processor—<http://www.byte.com/art/9702/img/027bita2.htm>
- Article in *c't* on PowerPC processor—<http://www.heise.de/ct/Service/English.htm/9611270/>
- Mac OS-compatible products—<http://www.macsoftware.apple.com/>
- Software titles on the Mac OS platform—<http://www2.apple.com/whymac/onlymac/default.html>
- Available titles first shipped on the Mac OS—<http://www2.apple.com/whymac/macfirst/default.html>
- PowerPC native software products currently being shipped—<http://www2.apple.com/whymac/products/default.html>

- Press release on DVD-ROM prototype—http://product.info.apple.com/pr/press_releases/1997/q2/970212.pr.rel.milia97.html
- Developer release of ALOE—<http://www.opendoc.apple.com/dev/ALOE/aloemain.html>
- Vendors shipping OpenDoc and Cyberdog parts—<http://www.opendoc.apple.com/users/odproducts.html> or <http://www.cyberdog.apple.com/cyberparts.html>
- Apple Developer World—<http://devworld.apple.com/>
- Developer University—<http://devworld.apple.com/du> or <http://devworld.euro.apple.com/du>

Technology

- MacApp Release 13—<http://devtools.apple.com/macapp/>
- WSI NewsWatcher (Word Services enabled version)—<http://www.wordservices.org/Developers/>
- Client and server applications that support Word Services protocol—<http://www.wordservices.org>

Business

- Direct mail break-even spreadsheet—<http://www.devworld.apple.com/mkt/adtop.shtml>

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; from Canada, call 800-637-0029; from other countries call 716-871-6555. Or, send an e-mail message to <order.adc@apple.com>. The *Apple Developer Catalog* is also available online on the web at <http://www.devcatalog.apple.com/>.