



AppleDirections

Inside This Issue

Editor's Note: Real News	2
IndustryWatch	3
MacApp 3.1 Goes PowerPC	9
MacinTalk II and MacinTalk Pro	9
PowerPC Processor Wins Award	9
Macintosh LC 520	10
Apple Products Meet EPA Standards	10
Third-Party Compatibility Test Labs	11
CD Highlights	11
Human Interface	12
Making the Transition to PowerPC	14
Foreign Language Kit Family: Global Opportunity	15
Shop Talk: AppleLink Avenues to Developer Resources	17
Market Research Monthly: PowerPC to Flood Market	20
Marketing Feature: Guide to Getting Venture Capital	22
It Shipped	26
APDA Top Ten	26
Hot Products of the Month	26
Now Available from Apple	27
GetNextEvent	28

Apple News

OpenDoc To Be Developed For Windows and OS/2

It's unusual for an *Apple Directions* lead news story to be about companies other than Apple Computer, Inc., but then again, Apple has never before announced a technology like OpenDoc, the open, cross-platform, compound document architecture we've been talking about since the Worldwide Developers Conference in May. See this month's Strategy Mosaic, right next door on this page, for details about OpenDoc, which, by the way, is the new name for the compound document architecture July's *Apple Directions* reported about on page 8.

At June's PC Expo, WordPerfect Corp. and Novell Inc. announced plans to deliver OpenDoc for the Microsoft Windows platform by summer 1994, at the same time as Apple's planned release of the Macintosh version. At the same time, IBM and Borland also announced their support of OpenDoc technology.

These announcements mean that if you develop an OpenDoc "part," the potential market for the product will be substantially larger than the Macintosh computer installed base:

Strategy Mosaic

Why 1994 Will Be Like 1984

OpenDoc Will Change the Macintosh . . . and More

By Gregg Williams, Apple Directions Staff

If you ever saw Apple's famous "1984" ad, you'll remember its assertion that, because of the newly introduced Macintosh, "1984 [the year] won't be like 1984 [the book]."

It's almost 1994, and the year *will* be like 1984. Apple will introduce a new technology that will be as revolutionary as the Macintosh was in 1984.

"What could be that fundamental?" you may ask. "Is Apple Computer, Inc. hyping us again?" Well, maybe a little, but not much. OpenDoc is important because it is an open, cross-platform architecture for compound documents (documents with text, graphics, sounds, QuickTime content, and so on). In fact, because OpenDoc is a cross-platform technology, its influence will extend beyond the Macintosh itself. (If you're not familiar with OpenDoc, the same compound document architecture Apple announced at this year's Worldwide Developers Conference but with a new name, see "What Is OpenDoc?" on page 6 for an overview.)

What does this mean for you? If your company doesn't start planning for it now, you'll be left behind.

AppleDirections

Volume 1, Number 3

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

Editor

Paul Dreyfus (AppleLink: DREYFUS.P)

Technical Editor

Gregg Williams (GREGGW)

Business & Marketing Editor

Dee Kiamy (KIAMY)

Designer

Robert Stone

Production Editor

Lisa Ferdinandsen (LISAFERD)

Contributors

Juan Bettaglio, Cassi Carpenter, Suzanne Dills, Sharon Flowers, Chloe Freeland, Andie Galt, Mark Johnson, Stacy Moore, Rilla Reynolds, Caroline Rose, Anne Szabla, Wendy Tajima, Jessa Vartanian, Liz Westover

Manager, Developer Press

Dennis Matthews

Manager, Developer Support Information

Greg Joswiak

Production Manager

Diane Wilcox

PrePress/Film

Aptos Post

Printer

Wolfer Press Co., Inc., Los Angeles, CA

© 1993 Apple Computer, Inc., 20525 Mariani Ave., Cupertino, CA 95014, (408) 996-1010. All rights reserved.

Apple, the Apple logo, APDA, AppleLink, AppleShare, AppleTalk, A/UX, EtherTalk, LocalTalk, MacApp, Macintosh, MacTCP, MPW, MultiFinder, Newton, and TokenTalk are trademarks of Apple Computer, Inc, registered in the U.S. and other countries. AppleScript, develop, DocViewer, Finder, KanjiTalk, MacinTalk, MacSNMP, PlainTalk, QuickDraw, QuickTime, System 7, ToolServer, TrueType, WorldScript, and Zeal are trademarks of Apple Computer, Inc. Classic is a registered trademark licensed to Apple Computer, Inc. DEC is a trademark of Digital Equipment Corporation. HyperCard is a registered trademark of Claris Corporation. ITC Zapf Dingbats is a registered trademark of International Typeface Corporation. UNIX is a registered trademark of UNIX System Laboratories, Inc. 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

Real News

First off, no, I don't have any inside scoop on why John Sculley "really" stepped down in favor of Michael Spindler. I guess you'll have to wait for the *New York Times*, the *Wall Street Journal*, *Business Week*, or somebody else to tell you what's "really" happening at Apple.

It's one of those amazing times when people outside Apple seem more interested in Apple than the people who work here, and that's saying a lot, because people here at Apple really care about this place. It's always astounded me how much print Apple gossip gets, when there's so much of genuine substance to talk about (you know, wars, the economy, President Clinton's haircuts).

As I write this on July 2, sure, people here are talking about reorg and layoff rumors. But as you'll see from this issue, we're also spending our time making some real news and getting new products and technologies out to our developers and customers: There's OpenDoc, the exciting cross-platform compound document architecture we're creating in conjunction with WordPerfect, Novell, IBM, and the thousands of you developers who'll be working on OpenDoc components (see page 1).

Then, as you've been hearing about for some time, there's the PowerPC-based RISC Macintosh. Our engineers are churning full-speed ahead to have it ready by early next year. Yours should be too, getting applications ready for it when it ships so that, together, we can take the marketplace by storm. This month, on page 14 we tell you

more about how you can get your applications ready for PowerPC, and about tools Apple is preparing to help you do so.

You'll also be interested to know about global market opportunities created for you by System 7.1 and our language kit family, which will soon include specific kits for Japanese, Chinese, Korean, and Arabic (see page 15) and will one day support all the world's major languages on the Macintosh.

As you've probably come to expect from Apple, there's also a bunch of new hardware and software products this summer, including the ones covered in our News section as well as the Personal LaserWriter 300, which puts full LaserWriter functionality inside a 15-pound box, and the Portable StyleWriter, a 1.9 x 12.2-inch, 4.5-pound wonder that delivers laser-quality printing especially for use with PowerBooks.

Soon, *Apple Directions* will be providing details about Newton, QuickDraw GX, AOCE, and a couple of hot new Macintosh computers, all of which will be available in the very near future.

So, if you want to (and I know it's fun—I do it myself, and I also tape *All My Children* every day so I can watch it when I get home), you can follow all the ink about Apple gossip. I just hope you'll give the real news equal time, maybe even more time. It's an opportunity time to be an Apple developer, and we're continuing to pump out the technology you need to be successful.

Paul Dreyfus
Editor

Apple Directions Available On-Line Two Weeks Before Printed Copy

Log on to AppleLink August 2, and find out what's going on at Apple as it happens!

The next (September) issue of *Apple Directions* will be available on AppleLink on August 2 (path—Developer Support:Developer Services:Apple Directions:Apple Directions September).

Remember, this means you can read an unformatted, draft version of most of *Apple Directions* two weeks before the printed version is available. Final files for the complete newsletter will be posted August 16.

IndustryWatch: News & Perspective

Hot Chips, Cool Margins, and Pedagogical Polyester

By Amanda Hixson, Consultant, Instant Insight

The Pit and the Pentium

At one point, Apple cofounder Steve Jobs yearned to make computers common household appliances, as ubiquitous as telephones and toasters. Well, it looks like we might be halfway there.

With the arrival of Intel's new Pentium processors, it appears that Pentium-based systems can double as toasters. Heck, with power consumption in the 16-watt range, there might be a developer opportunity to create combination computer/microwave oven systems using these chips.

Most of you are familiar with the jet intake sound created by a 7-watt microwave as it fires up. Just wait until you turn on a Pentium-based system running a fan and heat sinks powerful enough to keep it cool. I can't say for sure, but I imagine experiencing the same audio effects produced by sorties leaving the flight deck of an aircraft carrier during the Gulf conflict.

I know, a little heat and noise are small prices to pay for machines that run up to 66 MHz. According to one recent report, driving 16-bit Windows 3.1 (on top of DOS) using a Pentium processor is wicked-fast. And, when they start shipping, Windows NT-specific applications are expected to blaze as well. Unfortunately, you probably won't see many 32-bit NT applications anytime soon.

If running Windows NT is a primary need, why is everyone so ga-ga over the steroid-laden 486-like Pentium anyway? There are some other new, exotic RISC chips that currently run NT (and, in some cases, were designed for NT) that deserve more consideration.

Before diving into RISC processing options, I have to be fair and point out that my understanding is that all of the non-Intel processors currently use emulators to run 16-bit Windows 3.1 with NT (which is a key reason many folks are ignoring some of the faster RISC alternatives). Emulation causes NT systems to do a lot of thinking when running in emulation mode. That's right, "thinking."

Thinking is the name Microsoft has given the process of converting 32-bit to 16-bit and 16-bit to 32-bit information (or any data structure not directly accessible by NT) as it passes between Windows NT and an emulated system. Just as it sounds, thinking under emulation is slow, even on speed demons like the RISC processors mentioned below.

Silicon Graphics, for example, has a nice little RISC model named the WS/EISA (WS is short for workstation) Mips R4400 that zooms along at 67 MHz and competes quite nicely, thank you, in 32-bit mode operations. Actually, the processor is clock-doubled (performing two instructions every clock cycle) and runs at an internal speed of 134 MHz that is gated by bus transfers and what-not, so it actually outperforms Pentium in many instances.

The real cheetah of new processors is DEC's Alpha AXP. It reaches 300 MIPS (millions of instructions per second) in some situations, because it also munches two instructions per clock cycle, and every instruction executes in a single cycle.

According to one informed individual I spoke with, DEC had to slow the clock speed of some Alpha chips used in developer machines being exported to Australia because the chip exceeded limits allowed by U.S. government export laws (the machines ranked as supercomputers). And you don't need asbestos gloves to touch the AXP, either.

And what about PowerPC? Motorola is anticipating equal integer performance and up to 40 percent greater floating-point performance than Pentium from the PowerPC 601. At an estimated street price of \$374 per chip in quantities of 20,000 (approximately a third of the cost of Pentium), the 66 MHz, PowerPC 601 processor is a steal. PowerPC is also designed for a broad range of systems, ranging from low- to high-cost, and you won't need oven mitts to handle systems running it, either.

I'm not saying that Apple must support Windows NT (although it could); I'm just saying PowerPC is a contender in the processor battle, no matter what operating system you decide to run. Besides, with costly Pentium upgrades required for 486 users, PowerPC could be a cost-effective alternative. Few existing 486 machines are really upgradable, and most require a new logic board or the upcoming Pentium Overdrive chip, code-named P24T, due late in 1994; of those that are upgradable, most don't have a 64-bit bus.

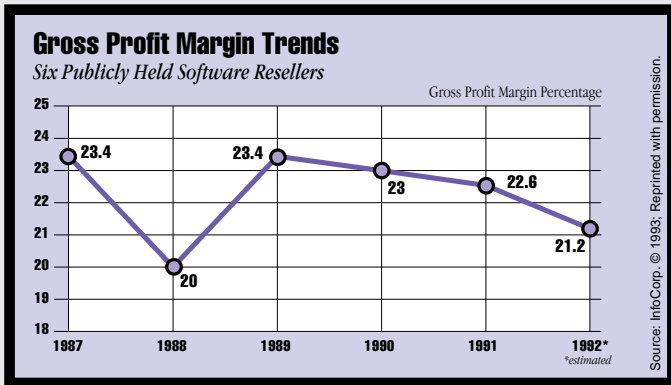
Declining Margins of Safety

Price erosion and margin compression are forcing resellers to become even more business-savvy despite increased unit sales. A variety of market conditions are making things pretty grim for traditional resellers. Resellers find themselves confronted with improved competition from established mail order houses and new, nontraditional distribution vehicles such as electronic distribution, CD-ROM bundling and software sales, OEM bundling, direct mail from manufacturers, superstores, and even proposed on-demand kiosks for various media, including software.

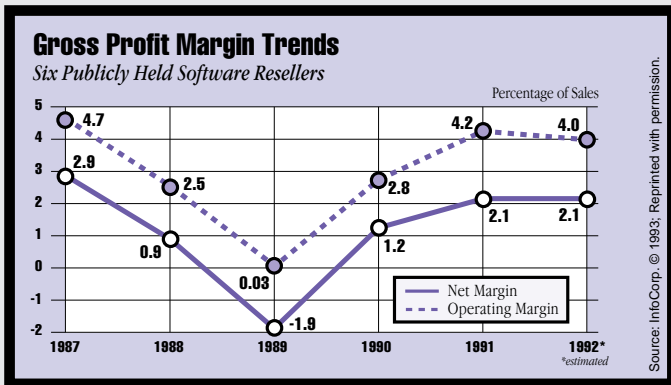
According to a recent InfoCorp research, however, six firms, using extremely different business models, have been able to stay a step ahead as markets continually change around them. These companies are Babbage, Corporate Software, Egghead, Micro Warehouse, Software Etc., and Software Spectrum.

Despite declines in gross profit margins (gross profit divided by gross sales) of more than three percent over the past six years, these companies are doing more than simply surviving, as shown in the graph "Gross Profit Margin Trends" (page 4).

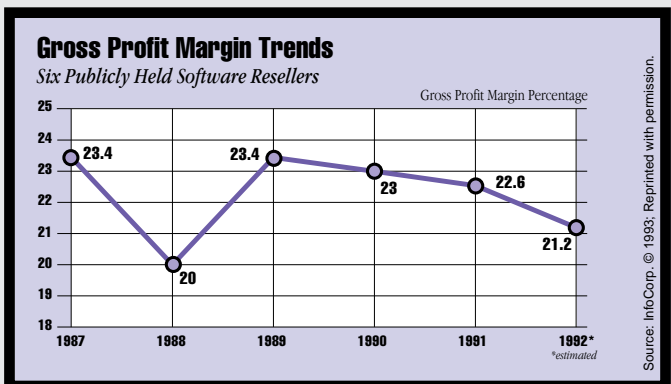
IndustryWatch



Although declining hardware prices are occupying the minds of most folks these days, especially around Apple, InfoCorp points out that software price wars are also heating up and software vendors can anticipate tremendous pressure as prices go down and volumes go up. Contrary to what you might expect, the six companies in InfoCorp's analysis are not only standing their ground, but are actually increasing their operating profit margins (the deduction of costs and operating expenses from sales) and net profit margins (the deduction of all costs, expenses, and income taxes from sales). See the graph "Profit Margin Trends."



To achieve profit margin growth, these companies have increased revenues while cutting costs. Perhaps most significantly, these companies have all reduced sales, general, and administrative costs (as shown in the graph "Selling, General, and Administrative Expense Trends"), a reduction that many companies fail to achieve, including some large personal computer companies that readily jump to mind.



If you'd like information about obtaining *Changing Business Models in the Software Channel*, the InfoCorp report on which this piece is based, contact InfoCorp at (408) 980-4300.

Pssst! Wanna Buy a Used CD-ROM?

According to a recent article in the *Wall Street Journal*, universities and colleges are the next great target for media vendors intent on selling into potential high-growth markets. We're not talking software here, however, we're talking CD-ROM-based books.

It seems that college textbooks are now way too expensive (often running in the \$70-plus range), weigh as much as small boat anchors, and are about as current as most technology companies' published strategies or a four-year-old child, whichever is older.

Because of the time it takes to update books and get them printed, the *Journal* states that many colleges are now requiring students to purchase computers (claiming a penetration rate of 20 percent for all colleges and somewhere between 42 and 95 percent at schools in the high-rent district). Along with their computers, many students also receive subject-specific materials and books on CD-ROM for much lower prices than their paper-bound equivalents.

Surprisingly (or not, if you've ever worked in the book publishing business), many of the CD-based materials are not being produced by traditional textbook publishers. It seems that book publishers, already bitten early on in the PC business by the failure of numerous software ventures, are taking a wait-and-see attitude. (When I was an acquisitions editor for a well-known book publisher some years back, educational software had to be pedagogically sound, which I later realized meant pedantic and boring.)

Traditional publishing's reticence to participate in a medium that offers quick turnaround on information (to match the six-month turnaround of current technology information) is a wonderful opportunity for those of you willing to connect with folks who know the publishing business and who aren't squeamish about trying new delivery mechanisms. In fact, opportunities to take part in bundled CD-ROM software distribution are rapidly approaching.

As James Noblitt, head of the Institute for Academic Technology at the University of North Carolina said in the aforementioned *Journal* article, "The textbook is the wrong medium for colleges today; I call it pedagogical polyester."

And so do I. ♣

Amanda Hixson is currently a consultant in the area of product and process management. Along with being a five-year Apple alum, during which time she was, among other things, an evangelist, product marketing manager, and software project leader, she is also the author of four books, a successful CAI training tool, a journalist, industry analyst, former book acquisitions editor, accounting manager, and perpetual realist (or cynic, depending on who you talk to).

Strategy Mosaic

1994

continued from page 1

Why Compound Documents?

Let me explain. (I chose the example that follows to show a trend, not to single out this particular product.) A certain word processor includes optional modules for spell checking, grammar checking, equation editing, object-based drawing, file searching, playing QuickTime content, and at least five other functions. The user's manual is over 800 pages long, and the software will take up over 7 MB of disk space if I install all of it.

Both developers and users agree that this ballooning of both size and complexity must stop. It is too hard to develop, support, and use such applications. The industry has been talking about "component software" for years, but no one has come up with the mechanism to make it happen.

OpenDoc is Apple's response to these problems; it includes technologies that Apple began developing as far back as 1985. OpenDoc will bring component software and compound documents to the Apple Macintosh and Microsoft Windows platforms by mid-1994.

When?

"When will OpenDoc ship?" is an important, if not key, question. Here's the latest word I can get (as of June 26): OpenDoc will be available to developers in a widely distributed Alpha seed by the end of 1993. We plan for the OpenDoc libraries to be final in the summer of 1994. At that time, the first applications that use OpenDoc will ship.

Apple will support a small number of developers in the next few months (before the end-of-year release to all developers). If you're interested in being an early site for OpenDoc, send an AppleLink mes-

sage to OPENDOC.IDEAS with a description of what you'd use OpenDoc for. If the members of the OpenDoc team like your ideas enough, they'll get in touch with you.

Overall Advantages of OpenDoc

Here are some of the reasons why you should use OpenDoc:

- *OpenDoc is an open platform.* The OpenDoc source code will be made available, and other vendors can implement both OpenDoc and OpenDoc parts on any platform without any restrictions from Apple. Apple is working with several industry leaders including IBM and Novell (see page 1 news story) to ensure that OpenDoc is available on multiple platforms, including Windows and several versions of the UNIX® operating system. OpenDoc is designed to make it easy for vendors and developers to replace parts of OpenDoc with their own technologies. In this way, system vendors can add platform-specific features into the OpenDoc environment. Developers can also introduce new value-added facilities into OpenDoc.

- *OpenDoc is a cross-platform technology.* OpenDoc will be released simultaneously for both the Apple Macintosh and Microsoft Windows platforms, with UNIX versions probable in roughly the same time frame. OpenDoc documents can be moved easily between any platforms that support OpenDoc. OpenDoc part editors can also be moved between platforms since the OpenDoc APIs (application programming interfaces) will be identical on all platforms. However, note that OpenDoc does not automatically solve most portability problems, such as different imaging models and human-interface toolboxes. You will still need to solve such problems with existing mechanisms.

- *OpenDoc is vendor-neutral.* This means that OpenDoc isn't designed for one platform and moved (perhaps imperfectly) to others. As one OpenDoc spokesperson said at this year's Worldwide Developers Conference (WWDC), "Our goal was to make OpenDoc the best possible design for compound documents on the Macintosh, Windows, and UNIX platforms." The OpenDoc team has been working with platform vendors and ISVs (independent software vendors) for over a year to make sure that OpenDoc will work well on all major platforms.

- *OpenDoc is scriptable.* OpenDoc supports the Open Scripting Architecture and extends it down to the part level. This means you can combine OpenDoc parts with scripts to build a custom solution for you or a client.

- *OpenDoc will work with OLE 2.0 and vice versa.* Users can drag OLE 2.0 "objects" (the counterpart to OpenDoc parts) into an OpenDoc document, and they will work correctly—and OpenDoc parts will also work in OLE 2.0 documents. That's a big win for both you and your customer—there's no advantage in standardizing on OLE 2.0, and you end up providing a better solution to your customer.

For a more detailed comparison of OpenDoc and OLE 2.0, see "OpenDoc vs. OLE 2.0" on page 7.

Developer Advantages of OpenDoc

Although it will take considerable work to make the full transition to OpenDoc (see "Converting to OpenDoc," later in this column), you can easily make your existing applications capable of embedding OpenDoc parts. Furthermore, the OpenDoc environment will make your engineers' jobs a lot easier. Here are some of the advantages:

- Letting your existing application embed OpenDoc parts is only a little more difficult than supporting QuickTime. Once you do this, you get many of the benefits described below.

- OpenDoc lets your users immediately adopt new technologies, with no further work on your part. For example, if Apple provides a new facility like QuickTime or sound, every OpenDoc container—including existing ones—could use it immediately. For you, this means no pressure to change your application every time Apple, Microsoft, or any other company introduces new technologies.

- OpenDoc lets you concentrate on programming what your

Compound Documents—Friend or Foe?

If you won't take my word about the importance of compound documents, here's somebody else's. In the March 30, 1993 issue of *PC* magazine, Michael Miller says, "Most industry observers believe that we are moving toward integration built around reusable components that act as tools you can use in any document. . . . Down the road, you might not buy a new application. Instead, you would install a new tool, which could be used in conjunction with your other tools."

Microsoft is definitely going down the compound-document road with its OLE (Object Linking and Embedding) 2.0 architecture. So if Microsoft is doing it, and Apple is doing it . . . ?

Apple recommends that you begin using OpenDoc technology as soon as you can. See "Converting to OpenDoc" on page 7 for more details. ♣

What Is OpenDoc?

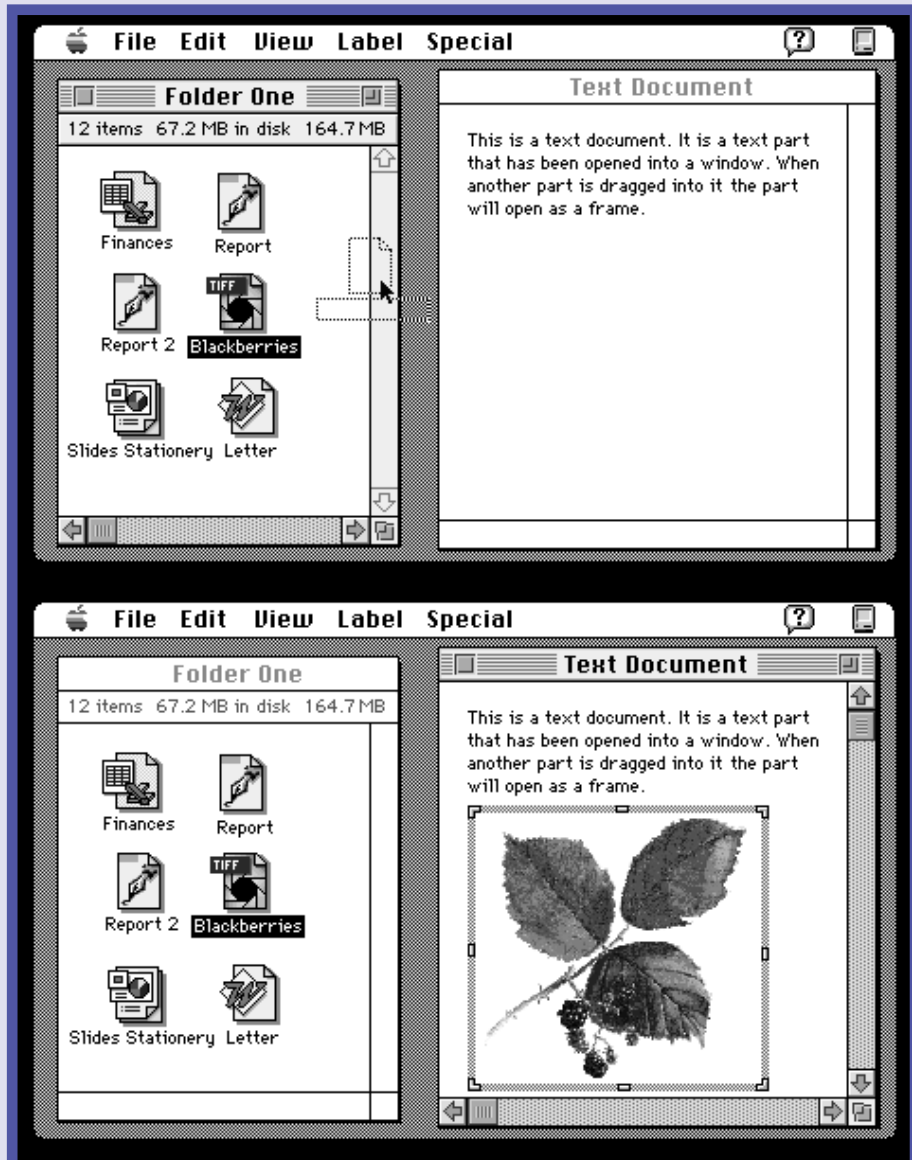
Many people aren't aware of what OpenDoc is and how it works. In a word, OpenDoc is an open, vendor-neutral, cross-platform architecture for documents that contain text, graphics, spreadsheets, and other kinds of data. Here's a short introduction:

With OpenDoc, the documents created by today's large, monolithic applications will be replaced by collections of what are called *parts*. Each part consists of one kind of content—such as a spreadsheet, a graph, or styled text. Part editors are responsible for manipulating, displaying, and printing parts on screen and on paper. (Apple uses the word *part* to mean, depending on its context, a specific occurrence of a part within a document, the visible appearance of a part in general, or a part editor.) An OpenDoc document can be any kind of presentation, from an existing 2-D productivity document all the way up to a full 3-D, real-time, multimedia virtual reality.

When a user clicks on content in some part of a document, the region represented by the part becomes active, a visible border called the frame appears, and the part takes over the menu bar. (A frame is limited to a single page, so some parts—like a lot of text—will have multiple frames.) The part typically also selects the content where the user clicked, so the user can begin editing immediately. OpenDoc can show a part in two ways—with its visual content and its frame, or as an icon.

OpenDoc supports more direct manipulation than the current Macintosh model does. For example, you can drag a part from one window into another (see "Dragging and Dropping Parts") and the part opens and displays itself in the second window. This contrasts sharply with today's way of doing things: opening the first application, selecting the item to be moved, copying the item onto the Clipboard, opening the second application, and pasting the results—which may not be as good as the original—into the window of the second application.

You may want to create both a part editor and a part viewer for your type of data. A part editor allows the user to manipulate the contents of a part, including viewing and printing it. A part viewer can only do the last two things: drawing the part on the screen and printing it out. You could sell the part editor and give the part viewer away free. That way, anybody can view and print



Dragging and Dropping Parts

When the user drags the TIFF file into an OpenDoc document, the TIFF file displays its contents and its frame.

your parts with full fidelity. Alternatively, you may create a static representation of your part (such as a PICT) and put it in the document along with the native format of your part.

In contrast to the container/contained relationship between two parts, one part can also be linked to another. A link is a unidirectional, permanent reference from one part to the other. When the first part changes, the link notifies the second part to change correspondingly. The classic example of linking is linking an area of spreadsheet data to a bar chart; when you

change the spreadsheet, the bar chart updates itself automatically.

OpenDoc part editors interact by calling C++ interfaces provided by other part editors and by the OpenDoc libraries. For convenience, OpenDoc also provides a simple ANSI C API for all interfaces. In addition, OpenDoc implements the Open Scripting Architecture (OSA). In-house developers, system integrators, and end-users can customize off-the-shelf OpenDoc parts by linking them and by using a scripting language such as AppleScript or Userland Frontier. ♣

company does best. You don't have to spend tremendous effort to support other functions that users want—such as text editors, chart-graphing programs, spelling checkers, paint programs, or support for multimedia documents—because OpenDoc lets your users plug those in.

- Your OpenDoc part editors will be smaller and simpler than equivalent parts of a monolithic application. For your technical people, this means software that is easier to design, program, debug, maintain, and document.

Business Advantages of OpenDoc

OpenDoc won't take business from you. A saying within Apple is that OpenDoc provides you with "business as usual . . . plus." Here are some of the opportunities:

- As mentioned previously, OpenDoc lets you focus your effort on the primary value that you add, without forcing you to put a lot of effort into supporting features that are outside your specialty. In addition, OpenDoc part editors are smaller and easier to develop than large applications. This means that you can greatly reduce your time to market.

- If that doesn't get your attention, how about this? OpenDoc also allows you to start a new product with fewer resources and less money. This means you can do more creative things without staking your financial well-being on just one or two large products.

- Along the same lines, realize that OpenDoc levels the playing field for all developers. No matter how small you are, you can create an OpenDoc part or two and still have a good chance of starting a business with it.

- When you break your application into OpenDoc parts, you can create new products (especially for vertical markets) by bundling different subsets of your parts.

- Because OpenDoc is scriptable (using AppleScript, UserLand Frontier, or other scripting systems), you may find that there's money to be made providing customized solutions to businesses. This would also increase your company's stability by providing multiple sources of income.

- You can either license your parts to others or license parts from someone else to bring your OpenDoc-based products to market.

- Since OpenDoc parts are smaller, you should have fewer bugs to fix. Since the new versions of a part can be small, you may be able to distribute it electronically, saving considerable materials, postage, and administrative expense.

- Some people say that it's getting harder to make sales because many users have all the software they need. Since users will want to upgrade to products that support OpenDoc, you have an opportunity to sell OpenDoc products to both your installed base and new users.

- Finally, you may look at someone else's part and say, "I can do better than that." Go for it! There will definitely be opportunities for after-market parts that improve on your or someone else's part.

User Advantages of OpenDoc

Let's not forget the people we're all working for—the users. OpenDoc solves several big problems for them:

- More power, less complexity.* When developers add just one more feature, users get more power, but they also get more complexity. OpenDoc resolves these seemingly contradictory goals, because users will use one part—a styled text part, for example—and they will use it everywhere.

- Ease of use.* With OpenDoc, you click on a part and edit it in

the context of the document around it. You don't have to switch applications twice and use the Clipboard to edit an item.

- Reliability.* With the cut-and-paste feature of today, you have to worry about compatibility—in other words, would application B understand the data on application A? OpenDoc will make this problem a thing of the past. As long as users have the viewer for your editor, they will (by definition) be able to view the content without any loss of image quality.

- Usefulness in collaboration.* OpenDoc supports multiple "drafts" within the same document. (A draft is like a "snapshot" of the document in its current state.) This means that any user can work on the various drafts of

an OpenDoc document and even rescue material that had been deleted.

Converting to OpenDoc

Converting to OpenDoc is going to take some time, so here's what Apple suggests your engineers do to make the transition:

- Your application should implement Apple events, the Open Scripting Architecture (OSA), and the relevant Apple event suites. See "Message From the Worldwide Developers Conference" in the July 1993 *Apple Directions* for more details.

- In your next minor revision, have your application save its documents in the Bento format. Bento is a standardized, open-ended file structure that will work

OpenDoc vs. OLE 2.0

Apple OpenDoc

Microsoft OLE 2.0

Business issues

Open, vendor-neutral architecture	Proprietary architecture
Source code available	Source code not available
Multiple companies are working together to produce OpenDoc	Microsoft controls OLE 2.0
Developers will be using OpenDoc by the end of this year	Developers are using OLE 2.0 now
Any part can be at the "root" of a document	Only specialized containers can be at the root of a document

Human interface issues

Parts can be any shape	Objects must be rectangular
Parts can overlap	Objects must not overlap
You can start editing any content immediately just by clicking on it	You have to activate an object, then select the content to edit it; when objects are nested, you have to activate multiple levels

Other issues

OpenDoc allows you to link or script any document on the network	OLE 2.0 is restricted to documents on the same computer
OpenDoc makes collaborative work easier by maintaining multiple "drafts" of a document	OLE 2.0 doesn't have multiple "drafts"

on any platform. For more information on Bento, send your request to AppleLink address AMBER.IDEAS.

- Next, have your engineering team make your document into an OpenDoc container. This means that users can drag OpenDoc parts into your document, and your application will work with them. Apple recommends you do this for any products released commercially by mid-July of 1994 or later.

- Finally, design any new products and major revisions of existing products as one or more OpenDoc part editors. For any code written from scratch, your engineers should use either C or C++. This will give you more options in the future.

Remember, OpenDoc parts are smaller and easier to debug. This means that the goal of shipping a product by mid-1994 is more easily attainable than you might think.

What About Bedrock?

You've probably heard of Bedrock, an application framework that will implement an application on both the Macintosh and Windows platforms. A number of you are probably saying, "You told us that Bedrock was the road to the future, and now you're saying that

we should follow OpenDoc. I'm so confused that I may just give up doing either. What can you tell me to make me want to stay?"

Bedrock and OpenDoc are not competing technologies. (In the first draft of this column, I had to be corrected on this point.) They are complementary to each other—Bedrock makes it easier to develop applications for multiple platforms, while OpenDoc completely redefines what a document is and does.

As to the question of which to implement first, the answer depends on your company's situation and goals. However, I've seen several developers get very excited about OpenDoc being an open standard. The alternative—depending on Microsoft for your company's future—does not lead to untroubled sleep.

As I write this, the OpenDoc and Bedrock teams are in heavy consultation. Here is the word direct from both groups: The first level of Bedrock support for OpenDoc will probably be a version of Bedrock, the documents of which can handle OpenDoc parts. The eventual goal is to have Bedrock produce OpenDoc part editors. The manager of the Bedrock group says that Apple will have

"some form of OpenDoc support in Bedrock by the commercial release of OpenDoc." *Apple Directions* will get you more details as they become available.

Also, remember that there are many paths to the full use of OpenDoc. (See "Converting to OpenDoc.") If you have a MacApp 3.x program ready to convert to Bedrock, you may want to adopt OpenDoc at the same rate that Bedrock does. If your code has no ties to MacApp, you may want to focus on OpenDoc immediately. (In either case, you should use C or C++.) You have to ask yourself, "Where do I want to be two or three years from now? What will I sell, and to whom? What platforms will I support?"

Answering the Real Question

In simpler times (ha!), the real question used to be "Should I develop for the Macintosh or for Windows?" But the entire field has changed, and most developers ask for—no, demand—cross-platform technologies. Now the real question has become "Should I use Apple's technology or Microsoft's?" Or, in this case, "Do I use OpenDoc or OLE 2.0?"

Here are two of the best reasons for using OpenDoc. First, OpenDoc is an open standard. Second, OpenDoc is much more flexible, powerful, and easier to use (and develop for) than OLE 2. For more details, see "OpenDoc vs. OLE 2.0" on page 7.

About two years from now, software will begin to look different, and more and more applications will first embed, and then become, OpenDoc parts. You need to start making decisions for your company, now; I hope this Strategy Mosaic column will help you do so. ♣

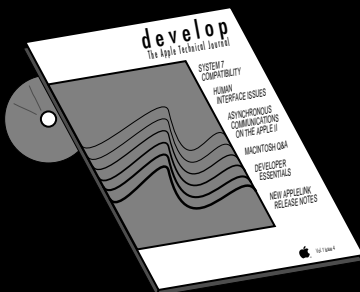
As part of the Apple Developer Group's charter to help developers be successful in their businesses, Strategy Mosaic is a monthly look at important pieces of Apple's overall strategy. Our intent is to give you information that you can use to plan your company's future. This column is based on information from managers throughout Apple Computer, Inc. Gregg Williams has been writing about Apple technology since 1982. You can contact him at GREGGW on AppleLink, or by phone at (408) 974-3264 (in the United States).

develop

The Apple Technical Journal

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

1-800-877-5548 U.S. (815) 734-1116 Outside the U.S.



Apple News

OpenDoc

continued from page 1

you'll also be able to sell to Windows and OS/2 users.

"This announcement by WordPerfect and Novell represents a significant step forward in providing unparalleled flexibility and ease of use on major industry platforms," said David Nagel, senior vice president of Apple's Macintosh Software Architecture Division. "OpenDoc technology will bring integration and interoperability to customers and developers on multiple platforms, ensuring a rich and streamlined computing environment."

Said Alan Ashton, president of WordPerfect Corp. "We applaud Apple's efforts to provide in OpenDoc an open architecture that supports compound documents and collaboration across platforms, and we look forward to working with Novell to develop the Windows implementation."

John Edwards, executive vice president of Novell's Desktop Systems Group,

Said OpenDoc addresses many of the problems Novell customers have experienced in working with compound documents. "We have committed to supporting OpenDoc in future Novell products, and it is a key building block in our AppWare development strategy." The AppWare blueprint, also announced at PC Expo, is Novell's comprehensive, multilayer, cross-platform development platform.

IBM's Personal Software Products will use OpenDoc with future versions of the OS/2 operating system. Apple plans for OpenDoc to use IBM's System Object Model (SOM) to package and execute parts of a compound document. SOM provides an efficient and flexible object interface and is based on the Object Management Group's standard for distributed

computing, Common Object Request Broker Architecture (CORBA).

Apple will license OpenDoc source code to system vendors who want to implement the architecture. Developers for the Apple Macintosh and Windows platforms will be able to begin coding for OpenDoc later this year. Implementation of OpenDoc on additional platforms is expected shortly after the Windows and Macintosh release.

MacApp 3.1 Goes PowerPC

Apple has announced its plans for a commercial version of MacApp 3.1 to be available by the end of the 1993 calendar year. MacApp 3.1 makes it possible for existing C++ MacApp programs to be recompiled into native PowerPC programs. With MacApp 3.1 and C++, developers will be able to make 680x0-based and PowerPC-based programs from the same source code.

MacApp 3.1 is different from previous versions because it is pointer-based, not handle-based. This change will allow developers to take full advantage of C++. MacApp 3.1 also fixes over 70 bugs that were found in MacApp 3.0.1. *Apple Directions* will bring you additional details as they become available.

MacinTalk II and MacinTalk Pro Bring Text-to-Speech Capability to the Macintosh

Apple Computer, Inc. is releasing two new products, MacinTalk II and MacinTalk Pro, that convert

typed text into audible speech, otherwise known as *text to speech* (TTS). Apple is licensing them to encourage developers to add text-to-speech capability to their Macintosh applications.

MacinTalk Pro gives a superior software-only text-to-speech facility on the Macintosh. Instead of using digitized voice samples, MacinTalk Pro constructs the speech directly from the text by following sets of grammar, syntax, and contextual rules for the U.S. English language. For example, it knows enough about the rules of spoken English to say "eight o'clock" when given the text "8:00" and "eight hundred dollars" when given "\$800." It has a built-in database of words that have an arbitrary pronunciation—it will say the word "enough" correctly, which a less sophisticated text-to-speech product might pronounce as "ee-noog".

MacinTalk Pro takes between 700 KB and 2.6 MB of main memory, depending on the number of voices in use. Apple recommends that users have a Macintosh Classic® II (or more powerful) computer.

MacinTalk II was designed for applications that have a smaller amount of available memory or that have modest needs for text-to-speech features. MacinTalk II usually takes 150 KB (0.15 MB) of memory. Apple recommends that users have a Macintosh Plus computer or later model. To use text-to-speech, you have to have the Text-to-Speech Manager, which is built directly into the code of MacinTalk II (but not directly into MacinTalk Pro). So if you're going to use MacinTalk Pro, you also have to have MacinTalk II.

Apple is charging a flat fee of \$1500 per application per year to license MacinTalk Pro and MacinTalk II together and \$1000 per application per year to license MacinTalk II by itself. Beta versions of MacinTalk II and Mac-

inTalk Pro are on this month's Developer CD, which is titled *CD Slickers*, and the commercial versions of these products will be available from APDA sometime this year.

PowerPC 601 Processor Wins Best of COMDEX Award

The PowerPC 601 processor, first in the line of PowerPC RISC (reduced instruction set computing) chips under development by Apple, IBM, and Motorola, captured *BYTE* magazine's Most Significant Technology honors in the Best of COMDEX awards at June's COMDEX Spring in Atlanta.

The award came as the result of a demonstration of the PowerPC chip at IBM's booth at the computer trade show. There, two sample applications running native on a prototype PowerPC-based Macintosh significantly outperformed Windows versions of the same applications running at 66 MHz on a Compaq 486.

The demonstrations substantiated Apple's estimate that PowerPC-based applications will have two to four times the performance of applications running on both the highest performance 68040-based Macintosh systems and Intel x486-based systems.

Developers are not required to modify existing 680x0 applications to run on PowerPC-based Macintosh systems, because of capability built into PowerPC Macintosh systems that ensures compatibility with existing applications. Developers will also be able to translate or port their applications to PowerPC. With translation tools, developers can translate their existing 680x0 Macintosh applications to run on PowerPC-based Macintosh systems, allowing applications to tap the power and

performance of the PowerPC microprocessor.

Developers who port applications to PowerPC will be able to take advantage of entirely new capabilities, such as speech recognition, text-to-speech, sound, telephony, video, 3-D rendering and animation, and complex modeling and analysis. (See "Making the Transition to PowerPC" on page 14 in the Technology section for more details about modifying your applications for the PowerPC platform.)

Apple plans to offer a full range of products based on PowerPC, starting with mid-range and high-end Macintosh personal computers. The computers will ship with Apple's standard operating system, System 7. The new computers will offer high performance and a variety of new features at competitive prices.

Macintosh LC 520: All-in-One Design For Education, Multimedia

Apple has shipped its newest "all-in-one" color computer, the Macintosh LC 520. Intended to meet the needs of the education and multimedia markets, the new computer will first be sold only to K-12 and Higher Education institutions in the United States.

The Macintosh LC 520 adds high-quality stereo sound and an exceptional color display to the chip architecture and feature set of the Macintosh LC III. Like the Macintosh Color Classic, it packs all its technology into an integrated design. Its features include:

- internal AppleCD 300i double-speed CD-ROM drive and 1.4MB Apple SuperDrive floppy disk drive
- 14-inch Sony Trinitron color

display delivering 640 x 480, 16-bit image resolution in thousands of colors

- integrated headphone jack and omnidirectional microphone
- 25 MHz Motorola 68030 microprocessor
- 5 MB of RAM (standard), expandable to 36MB, and an 80MB internal hard-disk drive
- two Apple Desktop Bus (ADB) ports
- single LC-style processor direct slot (PDS)

In addition, the Macintosh LC 520 is compliant with the U.S. Environmental Protection Agency's Energy Star Program. Like the Color Classic, it automatically reduces power consumption to less than 30 watts when idle, a feature that could cut by more than 50 percent the electricity used by the system. Through a control panel, users define the period of time that the computer can be inactive before the power-down feature starts. (See the next story for more about the Energy Star Program.)

For the Macintosh LC 520 standard 5 MB RAM, 80 MB internal hard-disk drive configuration, the U.S. suggested retail price is \$1,509.

Apple Products Comply With EPA Power Efficiency Standards

The vast majority of Apple Computer, Inc. products currently comply with the U.S. Environmental Protection Agency's Energy Star criteria for low-power consumption. Apple is also providing upgrades to make noncompliant products use less electricity.

The Energy Star program, a year cooperative effort between government and industry, is

expected to save consumers electricity costs and reduce air pollution caused by energy production. The EPA estimates that users of Energy Star computers will save \$2 billion annually.

The program was announced a year ago to allow industry time to develop engineering designs and marketing plans. President Bill Clinton issued an executive order in April 1993 that requires all federal agencies to purchase only computers that meet the Energy Star requirements.

Apple was an early participant in the Energy Star program, working with other manufacturers to help the EPA develop its technical criteria. The Apple Macintosh Color Classic, introduced in February 1993, was the first desktop computer system that automatically reduces power consumption to

less than 25 watts when it is idle. (See "Energy Star-Compliant Apple Products" for a list of Apple products that meet the EPA criteria for energy use.) A recent Apple survey of Color Classic users indicated that over 80 percent were consistently using the sleep mode feature to save energy.

To qualify for an Energy Star, a computer, printer, or monitor must use no more than 30 watts of energy to operate, or it must automatically "power-down" after a period of inactivity. An integrated monitor and CPU must use 60 watts or less. The requirements for desktop printers are specific to particular performance categories.

If you need information on Energy Star criteria, contact Brian Johnson in the EPA Global Change Division at (202) 233-9114. ♣

Energy Star-Compliant Apple Products

The following Apple products comply with Energy Star requirements:

- Macintosh Classic, Classic II, Color Classic
- Macintosh SE and SE/30
- Macintosh LC, LCII, LC III, and LC 520
- Macintosh Performa 200, 400, 405, 430, 450
- Macintosh IIsi
- PowerBook 100, 140, 160, 145, 145b, 165c, 170, 180, 180c
- StyleWriter
- Apple IIGS, IIe, IIc Plus
- Macintosh 12" Monochrome Display
- Macintosh 14" Color Display (M1198LL/B)
- ImageWriter II
- StyleWriter II
- Apple Color Printer
- Personal LaserWriter 300

Apple is providing energy savings capability for LaserWriter Pro 600 and 630 printers by means of special software upgrades. In addition, the Smartbar line of products from Sequence Electronics bring Energy Star capability to the Macintosh Centris 610 and 650, the Macintosh Quadra 800 and 950 and the Macintosh IIvx as well as most Macintosh computers and monitors introduced since 1986. These products allow most Macintosh computers or displays to automatically power-down after a period of inactivity without loss of data. ♣

Technology

Inside This Section

Human Interface: Users? What Users?	12
Making the Transition to PowerPC	14
Foreign Language Kit Family: Global Opportunity	15
Shop Talk: AppleLink Avenues to Developer Resources	17

Third-Party Compatibility Test Labs Provide Testing Facilities for Developers Across the Globe

Looking for a convenient way of testing your products against multiple Macintosh computers? If so, you need to know about the computer labs Apple maintains at a variety of locations across the globe so that developers in the Partners and Associates programs can test their products.

Perhaps the best known of the labs is the Apple Third-Party Compatibility Test Lab, located in Cupertino's new R&D Campus. It, like the other labs, provides developers with a unique opportunity to rigorously test products using the latest Apple hardware and system software. The lab is open by appointment to members of the Partners and Associates programs worldwide, as long as they are developing commercial products for the Macintosh. Other compatibility labs are located in Apple's local offices outside the United States; for a partial list, see "Apple's Third-Party Labs: A Partial Listing" later in this article.

Please turn to page 18

CD Highlights

Tool Chest Edition, August 1993: *CD Slickers*

It's new! It's improved! It's the August Tool Chest Edition of the Developer CD, *CD Slickers*, containing over 225 MB of new and revised tools, applications, utilities, and sample code of interest to developers.

This month we inaugurate the New System Software Extensions folder, where you'll find new Apple technologies such as AppleScript, QuickDraw GX, text-to-speech engines, and more. Included in the August New System Software Extensions folder are the following technologies.

AppleScript 1.0

This folder contains the complete AppleScript Runtime package including Scriptable Text Editor, Script Editor, AppleScript, and sample scripts. AppleScript allows for scripting between applications and the Finder.

Apple Shared Library Manager 1.1b8

This folder contains the Apple Shared Library Manager 1.1b8 development environment, featuring dynamic linking, loading and unloading of code, and shared libraries functionality, with support of C++ and procedural C (call by function name).

QuickDraw GX 1.0b1

This folder contains a beta release of QuickDraw GX software. It provides all the code you need to use QuickDraw GX plus documentation, development tools, sample code, and human interface guidelines.

SCSI Manager 4.3

In addition to the capabilities of the former SCSI Manager, the SCSI Manager 4.3 supports major new SCSI features such as disconnect

and reconnect, supports services such as fully asynchronous SCSI I/O, provides a more hardware-independent API (application programming interface) that minimizes the SCSI-specific tasks that a device driver must perform, provides full utilization of whatever SCSI hardware is available, and supports existing SCSI device drivers with minimum or no modifications. See "Apple Announces Asynchronous SCSI Manager" in the June issue of *Apple Directions* for more information about the SCSI Manager 4.3.



CD Slickers

Text to Speech

MacinTalk and Gala Tea are text-to-speech (TTS) engines that allow the Macintosh to convert typed text into spoken English. The purpose is to provide an alternative means for the computer to communicate to the user besides the usual audible beeps and screen prompts. TTS is especially valuable for disabled users, for teaching, for multimedia, and for interactive products where verbal instructions and feedback are useful.

Please turn to page 19

Human Interface

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. (Oh, and speaking of fonts, you probably caught the mistake in last month's issue in which we used **Courier** when we meant to use **Chicago**. We know what **Chicago** font is, but apparently the output device at our printer's didn't. It does now.)
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 terms like *PO*, *FOB*, *MRP*, 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,

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

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

(AppleLink: THE.DOKTOR)

Peter "Doc" Bickford is currently spending his sabbatical leave from Apple becoming less familiar with computers so he can design better interfaces. When he's at Apple, he's a member of the Human Interface Group in the Enterprise Systems Division.

Making the Transition to PowerPC

In the first half of 1994, Apple Computer, Inc. plans to introduce the first Apple Macintosh computers based on the PowerPC RISC microprocessor. These new systems will enable significant performance increases and new capabilities. Developers can choose from three paths to prepare their applications for the PowerPC-based Macintosh platform:

- emulation
- translation
- porting

We can't emphasize enough the need for you to begin thinking about how you'll prepare applications to take advantage of the new platform: Anticipating this strategic move is critical to both Apple's and your continued financial success. This article provides more detail about how you can follow each of the three transition paths. It also describes the development tools, from Apple and third parties, designed to support these various paths.

Emulation: No Modification Required

The first path, emulation, protects the developer's and the user's investment in software. By taking this path, you need not alter or modify existing 680x0-based applications for them to run on PowerPC-based Macintosh systems—they'll run automatically by emulation. This is made possible through an LC68040 (no floating-point unit) software emulator that will be an intrinsic element of the system software of PowerPC-based Macintosh systems.

Depending on the application, customers can expect their off-the-shelf 680x0-based applications to run on their PowerPC-based Macintosh at speeds comparable to a 68040-based Macintosh. Emulation capability will allow Macintosh

users to run their existing applications on PowerPC-based Macintosh systems, even if you're not yet marketing a native PowerPC version of the application.

Translation: Native Performance

The second path is translation. Developers who use the translation option can deliver applications offering native performance that significantly exceeds the performance available through emulation. Translated applications won't give developers the ability to take advantage of new features, such as speech recognition, text-to-speech capability, sound, telephony, video, 3-D rendering and animation, and complex modeling and analysis. If you select the translation path, you can choose from several third-party tools for translating existing 680x0-based Macintosh applications to run on PowerPC-based Macintosh systems.

Third-party tools developers are creating 680x0-to-PowerPC assembly-language translators, semiautomatic 680x0-binary-to-PowerPC-binary translators, and translators of Pascal and Object Pascal to C and C++. Assembly-language translators convert 680x0 assembly-language source files to PowerPC assembly language. The resulting PowerPC assembly-language source is then assembled through an assembler for the PowerPC processor. Binary translators convert, semiautomatically, a 680x0 application—without source—to a version that will execute on the PowerPC Macintosh.

Here are some of the third-party translation tools for Macintosh on PowerPC that have been announced to date:

- PortAsm from Micro-A.P.L., a 680x0-to-PowerPC assembly-language translator

- FlashPort from Echo Logic, a semiautomatic 680x0-binary-to-PowerPC-binary translator
- p2c from Sierra Software Innovations, a translator of Pascal and Object Pascal to C and C++

Porting: Native Performance; Added Features

The third developer path to Macintosh on PowerPC is porting. By porting applications to the Macintosh on PowerPC, you can offer users access to the native performance of the PowerPC processor as well as new Macintosh on PowerPC features enabled by the increased performance of the PowerPC processor. Both Apple and third parties will provide the tools required for recompiling existing 680x0-based Macintosh applications to execute directly on PowerPC-based Macintosh computers, and for creating new Macintosh applications specifically targeted for the Macintosh on PowerPC platform.

Apple is concentrating on providing tools for those of you who use C and C++, today's most commonly used languages. The tools required by the developer using C and C++ to port an application to Macintosh on PowerPC will be supplied by Apple in the Macintosh on PowerPC SDK (Software Development Kit). In addition, Apple is supporting third-party developers who are creating tools that complement those provided by Apple.

Macintosh on PowerPC SDK

The Macintosh on PowerPC SDK is a transitional development environment for the PowerPC-based Macintosh platform. It will enable you to bring applications to the Macintosh on PowerPC at or near the time of the introduction

of the PowerPC-based Macintosh systems. The Macintosh on PowerPC SDK will provide developers with a 680x0-based Macintosh cross-development environment for the Macintosh on PowerPC. The familiar MPW (Macintosh Programmer's Workshop) development environment, hosted under System 7, will be the foundation of this cross-development environment. In response to developer feedback, the A/UX dependency that was in preliminary plans for the SDK has been removed.

With the Macintosh on PowerPC SDK, you will be able to use a 680x0-based Macintosh to do all the coding of native applications for the Macintosh on PowerPC. In this environment, you will edit, compile, and link your applications on a 680x0-based Macintosh, and execute and debug the resulting applications on a PowerPC-based Macintosh.

The Macintosh on PowerPC SDK will include the following:

- C and C++ cross-compiler for the Macintosh on PowerPC
- PowerPC cross-assembler
- PowerPC cross-linker
- MacApp for the Macintosh on PowerPC
- a remote source-level and machine-level debugger for the Macintosh on PowerPC
- interface files and libraries for the Macintosh on PowerPC
- tools and documentation to ease the transition to an MPW-based development environment for users of Symantec's Think C
- additional tools to assist in porting 680x0-based Macintosh applications to the PowerPC-based Macintosh
- sample code illustrating how to port a 680x0-based Macintosh application to the Macintosh on PowerPC
- electronic documentation

In addition to providing the Macintosh on PowerPC SDK, Apple is working aggressively with Symantec on a native development environment for the Macintosh on PowerPC. This development environment will be available after Apple begins customer shipments of PowerPC-based Macintosh systems, allowing you to program with the PowerPC computers.

Delivery of Porting Tools

The Macintosh on PowerPC SDK will be made widely available during the third phase of a four-phase tools-delivery strategy for porting applications to the Macintosh on PowerPC. Here are the four phases:

- *Phase 1: Highly Limited; Non-Macintosh Hosted Environment.* The first phase is currently underway and involves participa-

tion of a very small number of software vendors who are working closely with Apple to port their applications to Macintosh on PowerPC using a cross-development environment hosted on an IBM RS/6000 workstation. This environment was created for early software development in the PowerPC program and is no longer available.

- *Phase 2: Limited Access to Prerelease Development Environment.* The second phase of porting will include several hundred Macintosh application developers worldwide. These developers will port their applications to preproduction PowerPC-based Macintosh systems through the use of the 680x0 Macintosh-hosted cross-development environment provided by the prerelease version of the Macintosh on PowerPC SDK.

Taking place before the release of the first PowerPC-based Macintosh systems, this phase of porting will include early adopters who wish to bring their applications to the PowerPC Macintosh at or near the time of introduction.

- *Phase 3: Broad Availability.* The third phase of tools delivery—wide availability of the Macintosh on PowerPC SDK—will follow the introduction and customer shipment of PowerPC-based Macintosh computers, scheduled for the first half of 1994. The Macintosh on PowerPC SDK will be made available to developers worldwide. Pricing and availability will be announced later.

- *Phase 4: Native Development Environment.* Following the first customer shipments of the PowerPC-based Macintosh, a native

development environment will be available. The native development environment is being jointly developed by Apple and Symantec. This development environment will provide developers with powerful, flexible, and easy-to-use development tools for the Macintosh on PowerPC. Once the native development environment is available, you will be able to use a single PowerPC-based Macintosh system for the entire development process from editing, compiling, and linking, to executing and debugging. Additional information about Macintosh on PowerPC is available on AppleLink (path—Developer Support:Developer Services:Macintosh on PowerPC.) ♣

Japanese, Chinese, Korean, Arabic Language Kits: A Global Opportunity

A new family of Macintosh software products from Apple is designed to give the Macintosh computer a clear advantage as the global multilingual system of choice. They'll also open a new market of millions of multilingual users to developers' products.

Following the successful introduction of its Japanese Language Kit in April, Apple Computer, Inc. will release language kits for Chinese, Korean, and Arabic character sets by fall 1993. This new family of products is designed to meet the needs of computer users looking for an elegant solution that lets them work with more than one language at the same time.

The kits serve two general purposes: First, they let users open localized applications with any version of System 7.1, Apple's "world-ready" release of Macintosh system software. Users in this

category are primarily people residing outside their native countries—for example, a Japanese native living in the United States who wishes to work with a Japanese-language application even though her Macintosh has the U.S. version of System 7.1 installed.

The kits also allow users to enter text in more than one language into the same document using applications that support the System 7.1 WorldScript application programming interfaces (APIs). Users in this category include businesses who want to insert non-English text into English documents (or Korean text into Japanese documents, or Arabic text into French documents, and so on), as well as educators and students who need to prepare documents using more than one language or who use interactive language learning programs.

The Global Computing Platform of Choice

The foreign language kit family gives the Macintosh computer, and Macintosh applications, a distinct advantage over other systems in delivering multilingual computing solutions.

No other personal computing platform (including DOS/Windows and UNIX) provides the same multilingual flexibility, giving users the ability to easily and elegantly input, display, and print text from different languages on the same system.

With the advent of Unicode, that's likely to change, although not for some time. Unicode is the future cross-platform character set encoding standard; its 16-bit architecture will allow for the encoding of more than 64,000 characters, enough to include those from all the world's languages as well as

technical symbols in common use.

Apple supports Unicode, as one of the major participants in preparing the standard, Apple plans to modify the Macintosh operating system to support it. Also, Apple engineers have built Unicode support into the Newton personal digital assistants to be released later this summer.

However, the Unicode standard is incomplete and at least a year (some say two or three years) from widespread use. Most of the major personal computer vendors have announced they will support the standard once it's complete. As a result, you need to keep Unicode in mind when you plan your applications of the future.

WorldScript and Unicode

The System 7.1 WorldScript APIs provide an immediate multilingual solution. WorldScript should not

be misunderstood as an alternative to Unicode. Unicode is a script-encoding standard, while WorldScript is a set of APIs that allow developers to create software that can handle all the various complexities of different written languages, such as contextual formatting and right-to-left scripts.

Applications that are written for WorldScript are ready to be localized for any region, because they already deal appropriately with the unique features of the various written languages. By supporting WorldScript until Unicode gains acceptance, you can take full advantage of the multilingual capabilities of the Macintosh platform and the global business proposition they offer.

Opportunity for Developers

The language kits expand the business opportunity for localized applications. Since the language kits ship worldwide, the market for, say, a Korean application will no longer be limited to the Macintosh installed base in Korea. A Korean application can run either on the Korean system that ships with Macintosh computers in Korea or on *any* Macintosh, anywhere in the world, with the Korean Language Kit. The same is true of applications localized for Japanese, Chinese, and Arabic users.

To take advantage of the immediate opportunity to expand sales into multilingual markets as well as future global market opportunities, your applications must be either (a) localized for Japanese, Chinese, Korean, and/or Arabic; (b) "globalized" for potential sale to any System 7.1 user, whatever their native language; or (c) both.

If you haven't localized your applications for one or more of these markets, you'll want to consider doing so. Information on localizing software can be found in the *Guide to Macintosh Software Localization*, published by Addison-Wesley and available in your

local bookstore or through APDA. (See page 27 for APDA ordering information.)

Apple is strongly encouraging developers to write globalized software that can be used anywhere in the world by speakers of any language using System 7.1 and one of the language kits. (For more on the strategy behind world-ready System 7.1 and how your applications can support it, see "Technology: System 7.1" in the October 1992 *Apple Direct*.)

What Is Globalization?

Globalization, in its broadest sense, means writing your software so that it takes full advantage of System 7.1 capabilities, supporting WorldScript 1 single-byte character sets, WorldScript 2 two-byte character sets, or both. It also means preparing your software to be "culturally neutral"—that is, picking interface elements and text strings for dialog boxes, names for menu items, and so on that can be easily understood no matter where in the world they are viewed or no matter what language they are translated into.

A globalized application can be more easily localized or customized for a special market since it is capable of using any script system and already works, technically speaking, in the language for which it is being localized (although text strings, menu names, and other elements of the user interface must still be changed for local users).

To quote *Inside Macintosh: Text*, "Globalization involves careful design and writing of the application and its textual and graphic resources." For more details about how to do this, you'll want to study *Inside Macintosh: Text*, also available in bookstores and through APDA.

User Advantages

The main advantage of the kits is that they allow the Macintosh to

handle more than one language without sacrificing any other capabilities. For example, users won't have to choose between running an English or a Japanese version of System 7.1; with a language kit, they can have the functionality of both on a single system.

Each kit includes fonts, input methods, and language resources for the particular language used by the Script Manager, which tells the Macintosh how to handle more than one script. The Chinese Language Kit includes both the traditional Chinese character set as well as the simplified character set used primarily in mainland China.

In addition, the kits contain a language register that allows System 7.1 to open localized applications using the correct language (actually the fonts for that language) in menus, dialog boxes, and other elements of the user interface, regardless of whether or not the region code in the "vers" resource has been correctly set. When you localize an application, you're supposed to set the region code to indicate the language you're localizing for; the region code tells the system to open the application's user interface in that local language.

Market Potential

The demand for the language kits is driven by the fact that all residents of a single geography do not speak only one language. This has long been true in Europe and is becoming increasingly so in the United States. According to the U.S. Census Bureau, 1 out of every 14 people in the United States speaks a language other than English at home.

The market for language kits is driven by the number of people who speak a certain language but live in a region where another language is used. The largest such market is made of Chinese speakers living outside China. Apple Far East estimates that there are more

than 5 million Chinese-speaking people living outside the greater China region. This includes 1 million Chinese speakers living in the United States, 500,000 in Japan, 2 million in the rest of Asia, 500,000 in Canada, 250,000 in Australia, and 1 million in Europe.

As for the Korean market, there are currently more than 600,000 Korean-speaking people in the United States, more than the number of U.S. residents who speak Japanese. (Note that sales of the Japanese Language Kit, which has been available since April, have far exceeded expectations.)

The United States also has a large Arabic-speaking population, but the largest markets for the Arabic kit will be France, with an estimated 3 million Arabic speakers, and North Africa, where the Macintosh ships with a French version of the system but where users also need to be able to work in Arabic.

When users in these markets purchase the language kits, they'll be buying them to use with developers' globalized and localized applications. In other words, these new Apple markets translate into market opportunities for developers, as well.

Think Globally

What's the bottom line? At the same time that you're rewriting your applications to take advantage of the new Power PC-based Macintosh, you have good reasons to globalize them, as well. By delivering globalized (or localized) software, you'll not only be able to sell to the same customers who will be buying Apple's language kits, but you can also help Apple sell the Macintosh computer, along with your products, as the preeminent multilingual solution to users around the world. ♣

ShopTalk

AppleLink Avenues to Developer Resources

Looking for a QuickDraw GX class? Have a question about the AppleScript Development Kit? Want to develop a Newton product? It's likely that a direct path to the information you need lies in talk boards, files, libraries, and addresses on AppleLink. While several AppleLink areas such as Developer Support, Worldwide Multimedia, and Enterprise Computing hold resources for your product development, this month ShopTalk highlights the Developer Support area.

Developer Support Resources

Double-click AppleLink's Developer Support icon and you'll see the following pictured below.



These icons and the resources associated with them evolve continually in response to Apple's new products and your suggestions. Considering the size and fluidity of this knowledge base, the first icon to note is BB Pathfinder, a tool that can help you locate bulletin board postings

quickly. If you haven't already used this tool, give it a try next time you're on AppleLink.

Developer Services Bulletin Board

Developer Services bulletin board postings are organized into the following folders:

- Headlines for Developers
- APDA – Tools for Developers
- Apple Information Resources
- Development Platforms
- Macintosh on PowerPC
- Marketing Information & Services
- System Software
- Periodicals
- Technical Documentation
- Tool Chest

Folders such as System Software, Periodicals, Technical Documentation, and Tool Chest parallel those on the Developer CD. While it's faster and less expensive to get content from the Developer CD, the AppleLink folders hold some content that isn't on the Developer CD—very

new files as well as older, archived files. For example, *Apple Directions* content in the Periodicals folder is updated bimonthly to give you access to news before you receive your printed copy. At the beginning of each month you'll find preliminary articles; midmonth you'll find final versions as well as late-breaking news.

Headlines for Developers is a collection of Apple and industry news that might be relevant to your product development. For example, current news items include "Apple New Multimedia Authoring Tools," "Apple's PIE Division Outlines Electronic Publishing Strategy," "Newton Architecture and Toolkit to Provide Open Platform for Developers," and "Apple Announces Support for Both TCP/IP and OSI Transport."

For software licensing, training schedules, and other Developer Services information and forms, Apple Information Resources is the place to check. This folder also contains new product data sheets, standards, and a handy list of Apple contacts.

Developer Talk and Exchange Boards

Developer talk boards are a good place to go for help—through want ads, developer code exchange, and posting questions. While scanning for tips from other developers, you might be able to help someone else with a question as well.

As do other AppleLink developer resources, talk board content and organization change frequently. Here's a sample of current topics.

• *Developer Talk*. Discussion boards include Game Develop-

ment, Anti-Piracy, Developer Tech Pubs, International Development, Comm Toolbox, Development Tools, Media Integration, Network & Connectivity, and Printing. Other postings include Want Ads, Developer Code Exchange, Surveys, Events, and Third-Party Press Releases.

• *System 7 Talk*. Discussion boards include Apple Events, Edition Manager, Text Services Manager, WorldScript, Help Manager, Sound, TrueType, Data Access Manager, and Telephone Manager. Other postings include New Information.

• *QuickTime Talk*. Discussion boards include General QuickTime, Human Interface, and Technical. Other postings include New Information, Code Exchange, and Code Updates from QuickTime Team.

• *AppleScript Talk*. Discussion boards include Scripting, OSI API, and Human Interface. Other postings include Want Ads, Script Exchange, AppleScript Development Kit Information, Apple Updates, Code and Scripting Additions Exchange, and Tools.

• *QuickDraw GX Talk*. Discussion boards include General QuickDraw GX, Tools, Human Interface, Graphics Architecture, Printing Architecture, and Type Platform. Other postings include QuickDraw GX Training, Apple Updates, and QuickDraw GX Want Ads.

• *AOCE Talk*. Discussion boards include General AOCE, Human Interface, DSAM, and MSAM. Other postings include New Information, Want Ads, Developer Code Exchange, Questionnaire, AOCE Contest, Marketing Programs, and Apple Updates.

Shop Talk

- *Selling Into Europe.* Postings include European Marketing Opportunities, Distributing Your Product, Localizing Your Product, Consulting and Marketing Services, and Developers/Distributors Exchange.

Dev Tech Answers and Tech Info Library

AppleLink has several technical libraries for your reference. Dev Tech Answers provides development-specific technical information such as new *Inside Macintosh* books, developer notes, technical notes, and code samples. The Tech Info Library, in the Support area, provides user-level technical information such as

product specifications. Both libraries retrieve the content that you specify using words, phrases, operators, and wildcard characters. Each library also has a Library Index button for direct access to the library folder hierarchy.

Technology Addresses

In contrast to the public Developer Support boards, you can use AppleLink addresses as a direct, confidential communications line to people writing on a number of projects at Apple. The Technology and Feedback Addresses table lists some project-specific addresses mentioned in recent Worldwide Developers Conference sessions.

When you send an AppleLink message to a technology address, be sure to include your phone number and mailing information, as well as your developer status (such as Partner or Associate). Also, by using the developer feedback addresses listed in the table, you'll help Apple to make sure you have the resources you need.

You can use the AppleLink Find Address command (in the Network menu) to locate other technology addresses not included here. If BB Path-finder or Find Address doesn't produce what you want, please send an AppleLink message to TONI.T to blaze a new information path.

Technology and Feedback Addresses

Technologies

COOL.DRAG	Drag Manager information and feature requests
AMBER.IDEAS	OpenDoc questions, seed requests, and product ideas; Bento format information
APPLESCRIPT	AppleScript-savvy applications; comarketing support
NEWTON.DEVS	requests for Newton and PIE development information
GXSAVVY.DEV	color-matching and leading-edge applications, fonts, printer drivers
SPEECH.IDEAS	feature requests and speech development ideas
BEDROCK	cross-platform feature requests, feedback
WORLDSCRIPT	WorldScript-savvy applications; language kit seed requests; enhancement requests; script system development
BLNOTE.IDEAS	Apple DocViewer bug reports and enhancement requests
AOCE.IDEAS	AOCE products
AOCE.HI	proprietary AOCE human interface issues
PEN.IDEAS	requests for purchase of hardware prototypes
QTIME.IDEAS	QuickTime application uses
HC.REQUEST	HyperCard® enhancement requests
POWERPC.DEVS	PowerPC development information
KALEIDA.DEV	ScriptX intermediate development

Feedback

DEV.CD	Developer CD feedback
A.DIRECTIONS	<i>Apple Directions</i> feedback
DEVELOP	<i>develop</i> , The Apple Technical Journal, feedback
TECH.NOTE	Technical Note feedback
DTA.FEEDBACK	Dev Tech Answers feedback

Compatibility Labs

continued from page 11

Cupertino Test Lab

The Cupertino lab actually features three separate labs equipped with more than 50 Macintosh computers representing the entire range of Macintosh models and system software (including international system software and keyboards). Prerelease versions of PowerPC-based Macintosh computers are expected to be available in the lab by this September.

Also available in the lab are some variety of third-party products, including video displays, hard drives, printers, plotters, and scanners.

Developers using the lab can take advantage of a variety of resources, including the following:

- *Support.* An engineer is available for help with system configuration, debugging, and Virtual User.

- *Networking.* Developers can test products over an extensive network, which employs LocalTalk, EtherTalk, and TokenTalk network connections as well as an AppleShare server and AppleTalk Internet Router.

- *Virtual User.* Virtual User 2.0 is a test automation system for multiple Macintosh computers. A host Macintosh executes tests on target computers connected to a network. The tests, described in a high-level scripting language, are executed by the host and acted on the target. Engineering support is available to help you use Virtual User.

- *Documentation.* Apple product specifications and technical documentation are provided as references to help test with previous and current versions of Macintosh system software.

Developers may work by themselves or team with others on joint application testing for one- or two-day sessions. Currently, no fee is

charged, but developers are required to share test data with Apple's compatibility engineering groups.

Apple's Third-Party Labs: A Partial Listing

Here's a partial list of Apple's third-party test labs in alphabetical order. Contact them directly for details about lab services or to make a test-session appointment at the lab available to you. If your country is not included here, get in touch with your local Apple office to find out which lab is available to you. As of this writing, Apple offers the labs as a part of the Partners and Associates programs at no extra charge. A future issue of *Apple Directions* will provide a complete list of Apple test labs.

Benelux (Belgium, Netherlands, Luxembourg). Labs in

Zeist and Brussels for registered commercial developers, distributors, system integrators, and in-house developers. In addition, Benelux developers can take advantage of a networking and communications lab at Cap Volmac equipped with Apple and third-party hardware and software. Contact Apple Computer BeNeLux, Third Party Relations, Handelsweg 2, 3707 NH Zeist, Netherlands; phone: 31-0-3404-86911.

IEA (Africa, Mediterranean, Middle East, Central Europe, Eastern Europe). For use by Partners from the IEA countries. Contact Apple Computer Europe—IEA, Le Wilson 2, Cedex 60, 92058 Paris la Defense, France; phone: 33-1-49-01-49-01 (ask for Sami Ben Romdhane); fax: 33-1-47-78-79-82; AppleLink: IEA.DEVSERV.

Japan. For Partners and Associates in Japan. (Japanese developers who have not joined the Partners and Associates programs may also use the lab for a fee.) Contact Macintosh Testing Lab, Zanfuku Bldg 6F, Shinjuku-ku, Takadanobaba 4-13-11, Tokyo 169, Japan; phone: 03-5330-6517; fax: 03-3368-3080; AppleLink: MIZUNO2.

United States. Available to members of Partners and Associates programs worldwide, as long as they are developing commercial applications. Contact Apple Third-Party Compatibility Test Lab, Apple Computer, Inc., 20525 Mariani Ave., M/S 302-1BD, Cupertino, CA 95014; phone: (408) 862-7175; AppleLink: TPC.LAB.

Spain. Two labs for Partners, one in Madrid, the other in Barcelona.

For the Madrid lab, contact Apple Computer España, S.A., Avda. Europa, 19 - Parque Empresarial "La Moraleja," Edificio Citibank, 28100 Alcobendas; phone: 34-1-663-17-80; fax: 34-1-663-17-90. For the Barcelona lab, contact Apple Computer España, S.A., Balmes, 150 atico, 08008 Barcelona; phone: 34-3-415-01-94; fax: 34-3-217-43-01. Reservations for either facility may be made by sending an AppleLink message to SPA.TPS.

Switzerland. For Partners and Associates. Contact Industrade—Apple Computer Division, Developer Services, Hertistrasse 31, 8304 Wallisellen; phone: ++41-1-832 81 11, fax: ++41-1-830-63-06. ♣

CD Highlights

continued from page 9

Thread Manager Extension 1.2

The Thread Manager is a System 7 extension that allows applications to make use of both cooperative and preemptive multithreading within an application context on all Macintosh platforms. This package is an update to the previous Thread Manager development package released in May. The current release contains Thread Manager 1.2, new header files, and an updated source-level debugger along with an updated utilities library.

Other compartments of our Tool Chest include the following new or revised items:

Communications Toolbox

This folder contains the complete development environment for Apple's Communications Toolbox: documentation, header files, CTB installer for System 6, and the basic connectivity set of tools to enable developers to write CTB applications.

Developer Notes Update

Included here is the developer note for the new Macintosh LC 520.

Inside Macintosh: Interapplication Communication

IM:IAC describes how applications can work together and how your applications can share data, request information or services, and allow the user to automate tasks.

Installer 3.4.3

This folder contains the latest update for the Apple MultiDisk Installer Software version 3.4.3. Installer version 3.4.3 addresses several bugs associated with Installer version 3.4. Included here are revised documentation, additions to the Installer Error's Appendix, and revised sample scripts.

KanjiTalk 7 Update

This update fixes bugs, adds extended symbols to some of the Kanji fonts, and improves compatibility when printing vertical text. The previous KanjiTalk 7 update is included.

Game Developer's Handbook

The Macintosh Game Developers' Handbook is a practical resource that covers issues unique to Macintosh game development. Included in this handbook are Macintosh development guidelines, tips, technical notes,

sample code, articles, and additional resource information. In addition, we have provided marketing information that includes an update on the Macintosh consumer market, information on positioning your product, and advice about getting product reviews. (See the July *Apple Directions* article "The Games Marketing Game" for more information about games marketing.)

MacSNMP

This folder contains documentation, header files, and test utilities that enable developers to write SNMP agents. It also includes support for the SNMP network management protocol and information about how to write an SNMP management agent for your application or service. This product is not just for networking but for all applications.

Next month: Look for the items on the disastrously optimistic list of "coming attractions" from Apple Directions' July 1993 CD Highlights, plus a new batch of Technical Notes, new Sound and Telephone managers, and more. See you there!

Alex Dosber
Acting Developer CD Project Leader

Business & Marketing

Market Research Monthly

PowerPC to Flood Market

Installed Base to Make Global Shift by Mid-Decade

PowerPC-based RISC Macintosh computers will make sales of their 680x0 CISC-based predecessors virtually obsolete, and the United States will represent only half the global installed base of Macintosh computers by the mid-1990s.

These are among the findings of Pieter Hartsook's latest Macintosh market study, *The Macintosh Market Review & Forecast, 1991-1996*. Hartsook is an industry analyst who has focused his research exclusively on the Macintosh since 1990. Although projecting future computer sales can be as risky as playing the blackjack tables in Vegas, Hartsook's forecasts have been relatively reliable. For example, in the fall of 1991, he forecasted 1992 worldwide Macintosh

shipments of 2,485,000; his later analysis of Apple's 1992 performance showed that actual worldwide sales that year were 2,608,000.

Hartsook bases his forecasts on primary research and analysis, including discussions with Apple Computer, Inc., and with Macintosh developers, distributors, and users. He updates his information on a quarterly basis. The information used for this article came from his latest study, which was released in April 1993.

His findings underscore two key messages Apple has been sending Macintosh developers:

- Begin preparing PowerPC applications now so they'll be ready when the wave of new RISC-based Macintosh computers hits the market in 1994 and 1995.
- Make sure applications run with System 7.1, Apple's "world-ready" Macintosh system

Inside This Section

Marketing Feature: 22
The Software Entrepreneur's Guide to Getting Venture Capital

software, and make full use of the System 7.1 WorldScript extensions so that they can take advantage of the increasingly global installed base.

The rest of this article takes a closer look at Hartsook's numbers.

PowerPC Acceptance

With a couple of exceptions, Hartsook's analysis bears out Apple's PowerPC strategy: Within several years, every Macintosh computer sold will employ a PowerPC RISC (reduced instruction set computer) microprocessor instead of today's 680x0 CISC (complex instruction set computer) chips, as shown in the figure "Worldwide Macintosh Shipments by Processor, 1993-1996" on this page.

The major exception to Hartsook's findings is that Apple is forecasting even stronger PowerPC-based Macintosh sales in 1994. At May's Worldwide Developers Conference, John Sculley announced that Apple expects to sell 1 million PowerPC-based Macintosh systems in calendar year 1994.

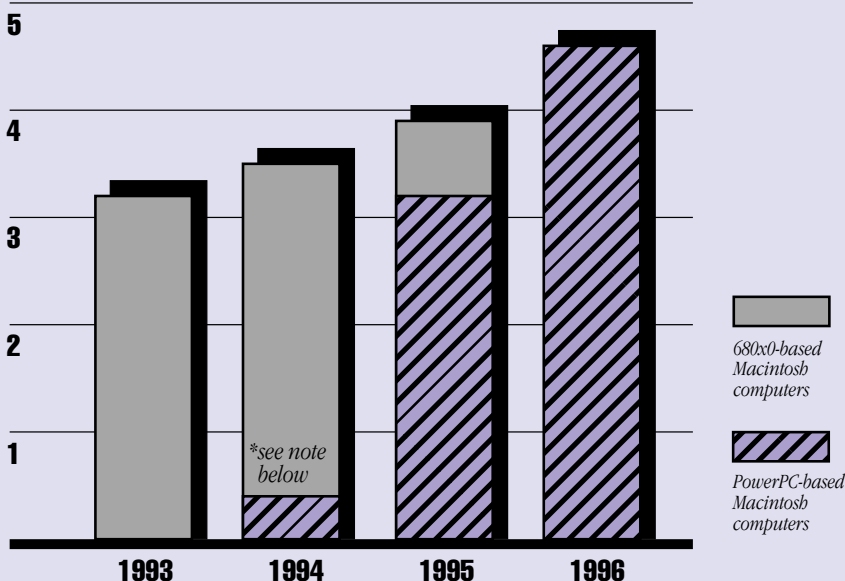
Also, Apple intends to manufacture and sell 680x0-based Macintosh computers well into the mid-1990s, while Hartsook projects that PowerPC sales will completely eliminate 680x0 sales by 1996.

Both Apple and Hartsook agree that by sometime in the middle of the decade, 680x0 Macintosh computers will no longer be sold. There will, however, remain a large installed base of 680x0, CISC-based Macintosh computers—nearly 17 million, according to Hartsook's projections.

Users of 680x0 systems will provide an ongoing market for developers' existing products and will require service and technical support. But the greatest opportunities for new Macin-

Worldwide Macintosh Shipments by Processor, 1993-1996

(millions of units sold)



*Apple has announced it expects to sell 1,000,000 PowerPC-based Macintosh computers in 1994

© 1993 The Hartsook Letter. Reprinted with permission.

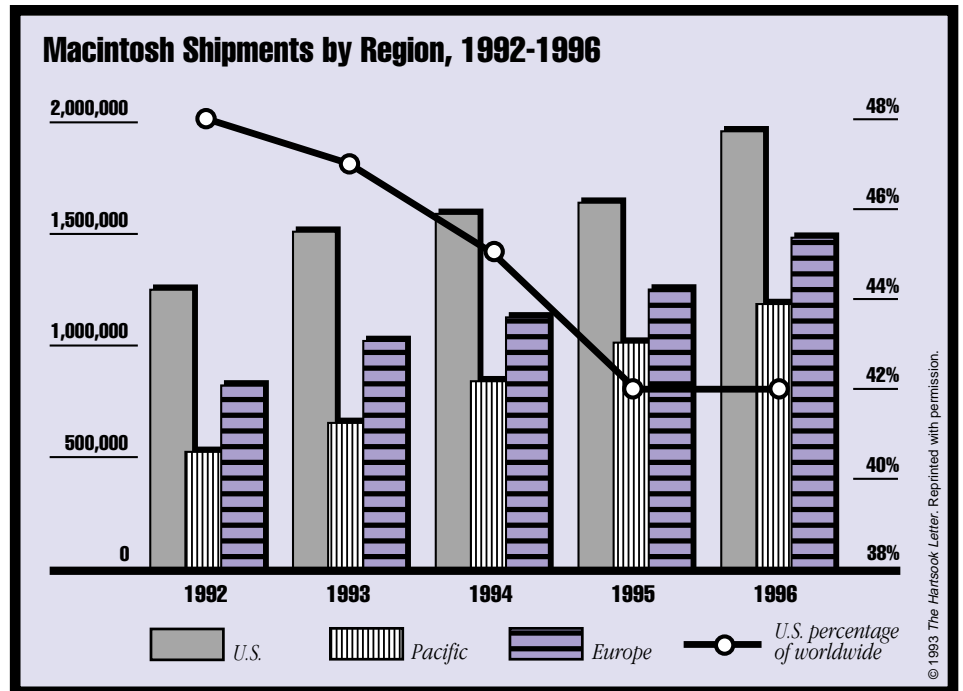
intosh product sales will increasingly be among users of the new high-performance, feature-enhanced RISC Macintosh computers. (See "Making the Transition to PowerPC" on page 14 for information about preparing PowerPC-based Macintosh applications.)

U.S. Percentage of Installed Base to Shrink

Not so long ago, the Macintosh computer was viewed primarily as a product for the U.S. market. That's changed over the past three years, and it will continue to change, according to Hartsook's analysis. What this means for developers is that, even though the United States will remain the single largest market for Macintosh products, there will be dramatic new opportunities for sales in other markets as well.

Through 1989, Macintosh computers sold in the United States made up nearly 80 percent of the worldwide installed base. By 1992, the U.S. share of the installed base had decreased to 62 percent and Hartsook expects it to decrease even further—to just over 50 percent—in 1996. (See the figure "Worldwide Macintosh Installed Base, 1992–1996" below.)

Apple's world-ready version of the Macintosh operating system, System 7.1, was released to help drive the Macintosh into markets other than the United States and to increase sales among customers who communicate in languages other than English. System 7.1 and its



WorldScript extensions, as well as the language kits Apple is presently releasing, can also help developers take advantage of the changing worldwide installed base mix.

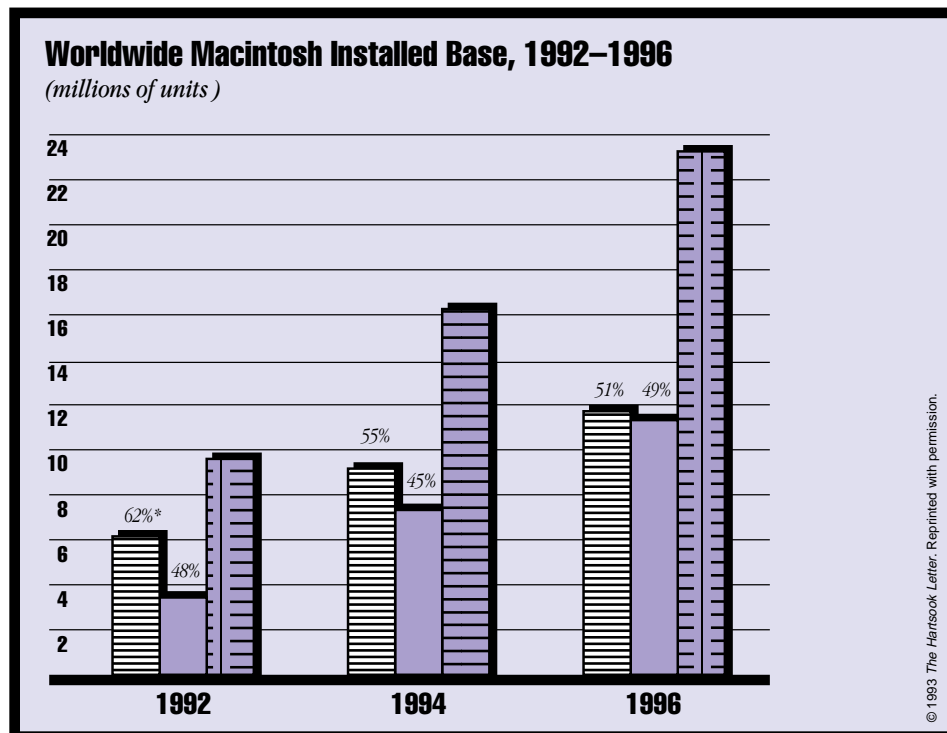
By taking full advantage of System 7.1, and especially WorldScripts I and II, developers can produce "globalized" software—software that is more easily localizable and can have a shorter time to market in new geo-graphic markets.

(See "Foreign Language Kit Family" on page 15 of this issue and "Technology: System 7.1" on page 1 of the October 1992 *Apple Direct*.)

Opportunities in Pacific, Japan

Hartsook groups the world's geographies into three regions: the United States, Pacific, and Europe. Of the non-U.S. regions, he anticipates that Macintosh sales in the Pacific will slightly outpace those in Europe (see "Macintosh Shipments by Region, 1992–1996" above). Japan, which during the first half of 1993 experienced the fastest growing Macintosh market in the world, will continue to be a hot Macintosh market into the middle of the decade. Says Hartsook, "Growth of the Macintosh market in the Pacific, particularly in Japan, continues to be strong with a forecast compound annual growth rate of 24 percent from 1992 to 1996." (A future Market Research Monthly will profile the Macintosh market in Japan.)

For information on how to obtain the complete *Macintosh Market Review & Forecast, 1991–1996*, contact *The Hartsook Letter*, 3001 Marina Drive, Alameda, CA 94501; phone: (510) 521-4988; AppleLink: HARTSOOK. ♣



Important note: All data presented in this month's Market Research Monthly is used by permission of The Hartsook Letter, © 1993; further reprinting without written permission of The Hartsook Letter is forbidden. The data is reprinted for readers' information only; its use in Apple Directions does not constitute an endorsement of the data by Apple Computer, Inc.

Marketing Feature

The Software Entrepreneur's Guide to Getting Venture Capital

By Peter Rosbko,
Mohr, Davidow Ventures

Venture capitalists view software as being one of the hottest investments of the '90s. Lucky for you, because you've just created the coolest software product—one that's sure to save business users lots of time and money, educate and inspire young computing enthusiasts, or simply entertain the family at home. All you have to do is finish coding and testing and get the final version into users' hands. Piece of cake—right?

Not necessarily. In today's market, developing a product and getting it to market isn't a cakewalk. While the process is challenging and exciting, it's also potentially extremely frustrating; there are many paths to choose and many decisions to make before a great concept turns into a software package selling off the shelves of Egghead or CompUSA.

However hot your product and company are, you still could fail for lack of the financial support you need to survive and grow. As an entrepreneur, one of the first things you must decide is where to get the cash needed to fund your enterprise. There are several places developers commonly turn to for funding (outside of family and friends). One potential source is corporations who sponsor smaller companies with strong potential. And, of course, there are banks; they are good sources of cash—if you can get it.

There also are individual investors, or what I call "angels." These are people who possess a

high degree of enthusiasm and knowledge about the software industry, enjoy dabbling in small investments, have capital, and are willing to take risks. These people often are successful entrepreneurs themselves.

Last, there are venture capital firms, such as Mohr, Davidow Ventures (MDV). This article explains what venture capitalists are, describes what they seek from developers, and hopefully helps prepare you to approach a venture capital firm such as ours.

Before you decide to approach a venture capital firm, I strongly advise that you thoroughly assess what you're getting into. Getting venture capital is a tough and demanding process. Venture capital firms naturally are particularly choosy about which products and companies they back; you face a lot of competition, and there's only so much money to go around at any given time. For example, at any given time Mohr, Davidow has approximately 35 companies active in its portfolio, and we look at 500 or more proposals each year.

Given this kind of competition, if you enter the process with your eyes open and having thoroughly done your homework, you'll be much better prepared to face the rigors ahead, and will stand a better chance of getting the financial backing you need.

What Venture Capital Firms Are—and Aren't

The first step is learning what venture capitalists (VCs) are and what they aren't, what they bring to your party, and what they will

expect of you. Venture capital investors are business partners, not bankers. (Bankers bring only money to the table, then bow out of your business.) The best venture capital firms offer you a lot more than just cash. They offer the expertise and resources needed for your startup to thrive and grow. You can expect VCs to give you advice, troubleshoot your business plan, and help you raise more capital. They can make informed suggestions for improving your product, strategy, finances, and organization. Also, contacts are one of a VC's biggest strengths. We're in the networking business and probably have contacts who can help you.

You can't expect a venture capitalist to run your company, but he or she can serve as a valuable resource. Think of it this way: When you're a general manager or product manager in a large enterprise, you can draw upon the expertise of the entire company, even people who are in another part of the organization. But as a small company, you don't have that luxury. Your board of directors provides the experience or direction needed to help you get things done—and your venture capital partner will probably sit on your board. For example, partners of MDV sit on the boards of approximately half of the companies it has invested in.

The partners at a VC firm usually come from a variety of professional backgrounds: consulting, direct industry experience, investment banking, and other occupations. They have held jobs such as sales rep, analyst, accountant,

programmer, entrepreneur, and a variety of other occupations. Therefore, a good VC firm can offer many different perspectives on your business through the experience of the firm's partners. Venture capital firms provide expertise in many areas, including technology; product development; marketing and sales; personnel management; operations, organization, and manufacturing; banking, finance, and legal; and, most important, they provide the industry insider connections to help make it all happen.

Furthermore, VCs come in many shapes and sizes. For example, some specialize in particular industries, others in geographic regions; some make their investments based on where a company is in its life cycle. MDV, for example, focuses on early-stage technology investments in companies located in the western United States.

VC firms also differ in the amount of funding they offer. Those in the software arena are generally willing to invest as little as \$100,000 and as much as several million. This isn't widely known. Most people think it takes needing millions for a venture capital firm to be even mildly interested in them. It doesn't. What it does take is offering a VC a solid, viable business proposition.

For example, MDV invested only \$100,000 in the PenSoft Corporation (based in Foster City, California) to help the founders write a business plan; we invested an additional \$400,000 once everyone was comfortable with the plan. At the other end of the scale,

another of our investments is in Knowledge Adventure, based in La Crescent, California. After Knowledge Adventure's product developed a proven track record in the market, we initially invested \$2.4 million to fund expansion.

What Attracts a VC

Of course, VCs have specific criteria for the companies they invest in. Here are some key things VCs look for in developers:

- *Honest self-appraisal.* Venture capital firms are interested in entrepreneurs who can honestly evaluate themselves, their team and its skills, their technology, product, market and growth potential, and, finally, the constraints under which they are working. VCs don't want to talk to developers who haven't demonstrated an ability to step back and critically view their situation. You therefore must be honest with yourself (and your investors) about your company's strengths and weaknesses. Most VCs are accustomed to the ups and downs of small companies, but what they despise most is being surprised by information that is sugar-coated or hidden.

- *Markets and growth potential.* Venture capitalists want to see that you're targeting markets in which your company can grow quickly and profitably and achieve high returns for the capital invested. Generally these are markets that are large or growing rapidly and have favorable competitive dynamics, or markets to which you bring a strong competitive advantage.

How large is large enough? It depends on how much money it will take to build your business. A software business with \$5 million in revenue will be attractive if you want only several hundred thousand dollars from an investor. But if you need \$3 million from a mainstream VC firm such as MDV, then

you must be able to build the case that you can become a company that earns \$20 million-plus in revenues. High growth translates to high value, so we look for products with strong growth potential. We will ask, "Do you think your company can be a \$25–\$50 million dollar business four to six years from now?" If your answer is not yes, you may not be a good candidate for VC funding. This means you must be targeting the right market segments, and more: For Macintosh developers, venture capital firms are usually interested in cross-platform products. They want to see that you have a Windows version under development, or at least somewhere in the works.

- *A distinct advantage.* Having a competitive advantage that is unequalled in the market is what allows your company to grow quickly and defend itself against competitors. I do believe that unique design insights into customer requirements, architecture, and the like are sustainable advantages for good lengths of time. Other advantages might include new distribution strategies, strategic relationships, top-notch management or technical talent, and other unique qualities.

- *A strong management team.* Of course, your "team" may be just you in the early stages of development, but it will grow as your enterprise grows. VCs want to see superior people in place no matter what phase of development your company is in. Tip-offs that can convince a VC that you have a strong management team include prior work experience, personal presentation, industry reputation and of course, references.

- *Products in "hot" categories.* What's hot now? Here are a few specific areas that are of particular interest to venture capital firms:

Five Assumptions to Avoid When Working With a VC

There are several assumptions that you naturally make about your company and how it operates. However, some of these don't apply to situations in which you partner with a venture capital firm. Here are five key ones.

Assumption #1: *Control is based on 51 percent ownership.* The reality is that control is based on whether you require outside funding—and how much.

Assumption #2: *I started it, therefore I am CEO.* If you own the whole thing, then of course it's up to you who is CEO. However, if you have a partner such as a VC firm, that partner will want you to put the best person in the job. Great entrepreneurs know their limitations, and they hire smartly below and above them, as needed.

Assumption #3: *Fancy organizational structures are a competitive weapon.* They might be, but more often than not they only confuse the situation and complicate your company's mission. If you want to experiment, start with an organizational structure in which every one clearly understands all the roles. Then later, when your business is more stable, you can restructure it to more closely reflect what you are trying to do and how you want to grow.

Assumption #4: *Venture capitalists are trying to take your company away from you.* Clearly, this is not in the best interest of anyone involved. After all, you know your business better than anyone, so it follows that keeping you actively involved in the business is the key to the overall success of the deal.

Assumption #5: *Hiring "just adequate enough" people to save money is smart.* Many companies think that hiring just-good-enough people will save them money, but successful companies know better. Successful companies aim high, hiring the best people possible for every position. ♣

content and electronic information publishing and distribution; software to support the emerging wireless and personal digital assistant (PDA) markets; collaborative work group applications; educational software and software that combines education with entertainment; and, at the higher end, tools and applications for the client-server market.

- *Success.* If your product is already on the market, it must be successful. That means it has reached some point of critical mass—by achieving a break-even financial picture or a recognizable market identity, or earning industry awards, accolades, and positive reviews.

- *Global potential.* For software, international product versions are an important revenue stream. VCs want to know that

companies are writing code so that localization is possible—and relatively easy.

Set Your Expectations

In addition to understanding what VC firms are searching for, to get off on the right foot you should also set realistic expectations about working with these companies. To help you do that, here are some important things to remember.

First and foremost, you must understand that venture capitalists are partners with whom you will develop a close relationship over time. Choosing a venture capital partner is a lot like choosing a senior executive: It must be done with substantial consideration. Venture capitalists aren't bankers who simply extend a loan and send you off to do whatever you please. They want to see your company grow and prosper. They've put their money on it—and they're prepared to invest a lot more than cash to ensure your success.

Also, realize that by accepting venture capital you're significantly changing your company's risk profile. On one hand, the risk is lower because you have additional capital to work with. On the other hand, the risk is higher because you'll be expected to show bigger results in a shorter time—and you'll be allowed fewer mistakes. The reason: Investors need high returns and liquidity. We want companies to grow quickly so that they can be taken public or sold.

Therefore, you must become lean and mean—and think big—rather than protect the status quo. Companies that provide attractive lifestyles for their founders and employees who do not grow will find themselves at odds with their VC partners. These tensions and conflicts can wreak havoc in small companies. The secret is to match expectations before you take on a venture capital partner.

You should also try not to make some all-too-common assumptions

about both VCs and your company. These assumptions, and the resulting ramifications, will turn off a VC. (For more information, see “Five Assumptions to Avoid When Working With a VC” on page 23.)

Selecting a Venture Capital Investor

OK, so you've done some soul searching and have decided you've got what it takes to partner with a venture capital firm. Now you're ready to take the first step on your journey to funding, organizing, and managing your company with the help of a VC.

It's important to choose the right venture capital investor because you're choosing a business partner—a real, live, informed, and opinionated person (or group of people) who will work closely with you for the next five or more years. Chances are, someone from that company will be a member of your board of directors. Your VC also acts as a representative of your company, and as such has extensive contact with the outside world. And because you are choosing a financial partner, you'll want to select extra carefully. In essence, you're choosing a cofounder.

The common-sense way to get referrals to a venture capital firm is to ask around. Talk with peers, your accountant or lawyer, hardware manufacturers, and other software companies; ask who they recommend. The software industry is full of resources such as newsletters, associations, and professional organizations. (MDV is a member of the Software Entrepreneurs Forum and the Software Publishers Association.) Also, meeting people at conferences such as the Apple Worldwide Developers Conference, OpCon, Demo, and the Software Publishers Association biannual gatherings is a good way to find worthy candidates.

Once you have a small list of possible partners, start checking

them out. To start the ball rolling, get information from peers, lawyers, bankers, other companies, and yes, even friends and family. Check out your potential venture capital partners by talking to their portfolio companies, companies they've funded in the past, and even their closest competitors.

Criteria for Selecting a VC Partner.

Here are several criteria against which you should evaluate candidate VC firms:

- *An appropriate investment match.* Does the VC usually invest in companies in your industry and at your stage of development? Is the VC big enough to support your capital requirements? Would the VC's reputation enhance your company's reputation? You'll want to shoot as high as possible and select a firm with a sterling reputation for stability and staying power.

- *Relevant industry experience and success.* Find out if the firm has relevant industry experience and success, and if it has the right contacts and resources to help get the job done. Furthermore, investigate the individual who will be working with you. Does that person have the skills, knowledge, experience, and personal reputation to help you succeed? Look at all the partners in a firm. In the best firms you will have access to expertise of all partners. Furthermore, what does the expertise of the firm look like as a whole?

- *Compatible people.* Are the partners (and the specific individual assigned to work with you) people that you and your colleagues can work with in the long run? It's imperative to spend time with the individual who will work on your deal.

- *Potential conflicts of interest.* Most VCs don't invest in directly competitive companies, but there can always be the potential of future conflict. Remember that competitors can also become

collaborators in some situations. For example, in the future you may wish to form a strategic alliance—whether it is technology-based or marketing-based—with another smaller company to compete against a larger one. In any case, however, it's important to understand the VC's attitude toward portfolio conflicts and how it handles those situations.

- *Attitude and demeanor.*

Don't overlook less tangible criteria, such as the VC's overall attitude and demeanor. A positive attitude is a real plus, as are patience, understanding, honesty, time, energy, enthusiasm, commitment, and, of course, a compatible business philosophy.

Making Your Approach

Once you've identified the firms who meet your criteria, the next step is approaching them with a business proposition. I have to admit that the way in which VCs choose which deals to make is very subjective, and there are many more reasons for us to say no than yes. So, how you approach a venture capital firm is critical. It's true that first impressions are lasting, and you only have about five minutes to create a first impression. Venture capitalists start forming opinions the moment they become aware of you, your product, or your company. This may even be before you contact them.

Personal introductions are the biggest lever you have to get into a VC's door. Remember, a lot of entrepreneurs are vying for our attention. To help develop an understanding of a business opportunity, we spend a lot of time talking to people who we think are smart, credible industry references. If one of them brings you to our attention, you are already head and shoulders above your competition.

If a personal introduction isn't possible, then you're probably forced to send a letter of introduc-

tion. Your objective is to get yourself and your team physically in front of the VC for a presentation or discussion. Your letter should briefly address the high points of the opportunity, and clearly describe your product. Depending on its style and level of interest, the VC may want to see a full business plan before inviting you to make a presentation. As early stage investors, MDV often works with entrepreneurs to build their plans if we're interested enough in the business.

Doing homework about each VC firm is also important. Do as much as possible to find out what a VC's particular needs are. Learn about its current portfolio companies and past investment decisions. This can give you tips about what "hooks" interest the VC. Some VCs are more technology oriented, while others are market or people oriented. The more you can tailor your introduction, presentation or discussion to the VC's needs, the higher the likelihood of generating interest.

Whatever initial approach you take—whether it's a phone call, letter, or fax, make sure it has a personal touch. Whatever you do, don't send a form letter: Don't forget that your primary objective is to develop a positive, working relationship with a business partner.

The Business Plan

Most developers write their business plans with an internal company audience in mind. However, if you're seeking venture capital, you must write your plan with the VC firm in mind, too. Often, your plan is what will get—or lose—a VC's attention. You should take extra care to ensure that the plan is clear, complete, well organized, and compelling.

We all treat plans differently. Some of us read only the executive summaries, some read only the résumés, some only make it

through the first paragraph, and some don't even bother to glance beyond the cover. Therefore, the plan should be concise. If you can't explain the gist of your business in 50 pages or less, something is wrong with either the business or your understanding of what is important about it.

Here is what VCs look for in the business plan:

- An executive summary that covers the high points, including what the opportunity is, who the management team is, who the customers are and why they will buy the product, and how much money you need. This should be contained in about three or four pages.

- The body, which covers details about the product, competition, marketing, distribution—all the milestones for measuring your progress. Describe the market and what your product brings to it. Try to get the VC excited about the potential for business growth. Focus on areas where you bring the most to the party or where the business risks are greatest. We want to see that you've thought about the critical issues relevant to the success of your business, and we want to understand why you think this is a high-growth opportunity.

- Financial projections. To the extent anyone can divine the future, you should make some estimates about what the next three to five years of the business might look like. These typically take the form of a monthly operating budget for the next year, quarterly projections for year two, and annual projections for the following years. A finer level of detail is probably asking too much.

What is even more useful is to project how much capital will be required to get through critical hurdles (milestones) in the business. These are things such as proof of fundamental technology, happy beta users, happy reference

accounts, or the cash flow break-even point. The company's value can increase at these milestones, and they are typically points at which additional capital would need to be raised, if necessary.

Making a VC Presentation

This is your chance to shine! Don't forget—you're trying to sell yourself, and I am your potential customer. Each venture capital firm is a little different, of course, but here are some general ideas for a successful presentation.

Assume you have one hour to make the pitch. Generally speaking, VCs are extremely busy, so brevity is key at this point. Your presentation should cover the high points, much like the executive summary in your business plan.

Talk about what your product does, who will buy it, and the size of the market opportunity. Be sure to talk about competitors, and why they might succeed or fail. Discuss your team and its qualifications. Also cover who or what is missing on the team, so the VC will know you've thought through all the issues regarding running your company.

Say how much money you want, when you want it, and what you plan to accomplish with it. Be prepared to discuss valuation.

Whatever you do, control the meeting so that you say everything you want to in the time allowed. You can respond to questions as they are asked, but make sure you get your messages across before you run out of time.

Due Diligence: Follow Through After the Meeting

After making your presentation to a potential venture capital partner, you must continue being proactive and manage the deal-making process from start to finish. Walking out the door and waiting quietly for the VC to contact you can spell trouble.

You'll undoubtedly develop some further questions or recognize some territory you may not have covered in your meeting. Does the VC need more information to make a decision? ASK! Can you provide more information? ASK! Do you know what the VC's main issues with your deal are? ASK! Do you know where you are in the process? ASK! This is a process you have to manage, and it's your responsibility to make sure everyone gets all the information they need. If you don't push the VC along, another more aggressive entrepreneur might push your deal off the VC's desk.

Due diligence begins at the end of your first meeting. We're interested in talking to people who know you, your team, and your customers. Be prepared to provide the VC with these types of references, if asked.

How long it takes a VC to make a decision can vary depending on how attractive the business and team are, and how familiar the VC is with the product category or technology. Although a decision could be instantaneous (extremely rare), it typically takes from 30 to 90 days.

After the Check Is Deposited

After the deal is final and the check is in the bank, you must continue proactively managing and developing the relationship with your VC partner. Assume it will be a long-term partnership, so do whatever it takes to make it work.

Getting your product to market still won't be a cakewalk, but you'll be in it with a partner who can add a layer of cash and resources to your own strong foundation. ♣

Peter Roshko is a general partner of Mohr, Davidow Ventures, a venture capital firm located in Menlo Park, California.

It Shipped!

Through the It Shipped! program, you can announce new and revised third-party products in *Apple Directions*. It Shipped! listings are also made available on the 3rd Party Connection AppleLink bulletin board. You can obtain an It Shipped! application by downloading it from the AppleLink network (AppleLink path—Developer Support: Developer Services: Apple Information Resources: Developer Program Information: It Shipped! @ Program).

Once you've completed the application, send it to Engineering Developer Support Center, 20400 Stevens Creek Blvd., M/S 42-ES, Cupertino, CA 95014, Attn: It Shipped! Program. Or send it by AppleLink to IT.SHIPPED.

These products shipped in June 1993:

Publisher	Product (version)
AXS	• PhotoProcessor 1.1
Callisto Corporation	• Super Maze Wars 1.0
CEDROM Technologies Inc.	• CDRMac 3.01b
Cherwell Scientific Publishing	• Ball & Stick 3.5
Copestone Software	• CollegeWare 1.0
Database International, Inc.	• Aware Property Manager 7.0
Datawatch Corporation	• Virex 4.0
Emmesoft di Marco Greppi	• MiniMIDI II 2.0
High Risk Ventures	• Space Madness 1.0.4
Iti S.A.	• ShuttleLink 1.0
Jets Cybernetics	• PIXIE: The Benchmark Edition • PIXIE: The Expert Edition
Maui Software	• MacFORMation 2.0
Natural Intelligence, Inc.	• QuickCode Pro 1.0
Now Software, Inc.	• Now Compress 1.0 • Now Up-to-Date 2.0.1
Personal Training Systems	• Adobe Photoshop 2.5 Tutorial
Philmont Software Mill	• Philmont Symposium System 2.2
Power R Inc.	• Model CC-10 1.0
Roger Wagner Publishing, Inc.	• HyperStudio 1.0
Teletypesetting Co.	• Digital Gourmet 2.6 • Digital Gourmet Deluxe 2.6
Working Software, Inc.	• Working Watermarker 1.0
XBR Communications, Inc.	• hi-BBS 1.5

APDA Top Ten

The following were APDA's top-ten selling products in June 1993:

1. E.T.O. Starter Kit/Renewal
2. MPW Development System
3. AppleScript Software Development Kit and Tutorial
4. AppleScript Software Development Kit
5. Intermediate Macintosh Application Programming and Macintosh Programming Fundamentals bundle
6. Multimedia Info Mailing
7. Macintosh Common Lisp
8. QuickTime for Windows Developer's Kit
9. MPW C version 3.3
10. MacTCP Developer's Kit

Hot Products of the Month

There are three this month, all of them important, but especially the new Symantec products.

Hot Product #1

Special introductory offer on the newest Symantec products!

Symantec C++ for Macintosh

Symantec C++ version 6.0 for MPW

THINK C version 6.0

• Symantec C++ for Macintosh

Features powerful object-oriented development tools within a completely integrated environment. Supports multiple editors and translators so that you can use your favorite tools and resource editors as well as scripts you've written with the environment. And, with ToolServer, you'll be able to customize menus and attach scripts based on Apple events, AppleScript, and MPW tools. The built-in SourceServer provides a source-code control system allowing teams of programmers to solve problems faster.

#T0565LL/A Symantec C++ for Macintosh Regular price \$499

Special price \$349 (U.S.)

• Symantec C++ version 6.0 for MPW

If you work with MPW and compact code is essential, you need Symantec C++ for MPW. This product helps you create reusable code and more maintainable, reliable applications. Its full native C++ compiler is lightning fast and generates highly optimized code for 680x0 processors.

#T0566LL/A Symantec C++ version 6.0 for MPW

Regular price \$499

Special price \$349 (U.S.)

• THINK C version 6.0

Version 6.0 enhancements make this product faster and more versatile than ever, improving your productivity with more powerful project management; includes a full set of tools and script support for major script-based languages.

#T0067LL/A THINK C version 6.0

Regular price \$299

Special price \$209 (U.S.)

Hot Product #2

Build AppleScript into your applications to support custom solutions for your users.

Get in on the scripting revolution with this special offer. Order the AppleScript Software Development Kit and Tutorial Bundle before September 30, 1993, and get 20 percent off!

AppleScript Software Development Kit Bundle

AppleScript is Apple's powerful new scripting system that works across applications and networks to deliver automation, customization and application integration capabilities for the Macintosh.

AppleScript will make custom application development faster and easier, because commercial applications can be used as components with AppleScript acting as the "glue" between them.

Look what AppleScript does for you when you build it into your application:

- increases the value of your products—Apple is making scripting an integral part of the Macintosh computing environment
- positions you to take advantage of emerging technologies such as speech recognition
- lets you expand into new markets—Apple is opening up new business markets that will demand AppleScript-compatible products
- allows for customization—AppleScript will allow your products to accept functional add-ons
- reduces time to market by allowing your applications to be combined with other packages
- creates a cross-platform scripting foundation

This special bundle contains both the AppleScript Developer's Toolkit and the Apple Events/AppleScript Programming Tutorial. The AppleScript Developer's Toolkit is designed to help you build AppleScript support into your applications. It contains a Macintosh disk, one CD-ROM disc, the *AppleScript Language Guide*, and *Getting Started With AppleScript*. The Apple Events/AppleScript Programming Tutorial is a self-paced disk and workbook tutorial that teaches you the basics of the AppleScript architecture.

#B1282LL/A

Special price \$275.00 (U.S.)

Hot Product #3

Special savings on Developer University Self-Paced Training Products.

Save up to \$395 on DU's self-paced training courses. Until September 30, 1993, you can order either Macintosh Programming Fundamentals or the new Intermediate Macintosh Application Programming course for only \$395! Or, buy them both at the special bundle price of \$695!

New! Intermediate Macintosh Application Programming

Extend your knowledge and Macintosh programming skills beyond the basics. This course teaches you how to build on the skills you learned in the Macintosh Programming Fundamentals course. Write code that extends the functionality of a single-Finder graphics editor to include QuickTime movies, Publish and Subscribe services, Cut, Copy, Paste, TextEdit, TrueType Fonts, required Apple events, MultiFinder, and more.

#R0438LL/A

Regular price \$495.00 (U.S.)

Special price \$395.00 (U.S.)

Macintosh Programming Fundamentals

Build highly functional Macintosh applications through mastery of fundamental Macintosh ROM routines and the application programming interface.

#M0997LL/B

Regular price \$595.00 (U.S.)

Special price \$395.00 (U.S.)

Get both and save \$395! MPF/IMAP Bundle

Contains both the Intermediate Macintosh Application Programming course and the Macintosh Programming Fundamentals course.

#B1287LL/A

Regular price \$1095.00 (U.S.)

Special price \$695.00 (U.S.)

Now Available From Apple


The following list shows APDA products that have become available to developers within the last several weeks. To get a full listing of all APDA products, check the current *APDA Tools Catalog*. For new-product announcements and the most up-to-date price lists, check AppleLink (path—Developer Support:Developer Services:Apple Information Resources:APDA—Tools for Developers).

#T0565LL/A	Symantec C++ for Macintosh	\$349.00
#T0566LL/A	Symantec C++ for MPW	\$349.00
#T0067LL/D	THINK C version 6.0	\$209.00

APDA Ordering Information

To place an APDA order from within the United States, contact APDA at (800) 282-2732; in Canada, call (800) 637-0029. For those who need to call the United States APDA office from abroad, the number is (716) 871-6555. You can also reach us via AppleLink; the address is APDA. If you're outside the United States, you may prefer to work with your local APDA contact. For a list of non-U.S. APDA contacts, see the "International APDA Programs" page in the *APDA Tools Catalog*.

GetNextEvent

The  indicates the trade shows/events at which Apple Computer, Inc., is scheduled to exhibit as of press time. This list may be incomplete. If you have information about a show that you want listed here, contact Developer Technical Communications,

20525 Mariani Avenue, M/S 75-3B, Cupertino, CA 95014.
For further information check the Events folder on AppleLink (path—3rd Party Connection:Events).

August 1 through 5

 **American Trial Lawyers Association (ATLA)**
San Francisco, CA
Contact: Kathy Kellett
AppleLink: KELLET1
(408) 287-4991

August 3 through 5

 **Siggraph**
Anaheim, CA
Contact: Russ Havard
AppleLink: HAVARD1
(408) 974-4371

August 3 through 6

 **Macworld**
Boston, MA
Contact: Dave Billmaier
AppleLink: BILLMAIER1
(408) 974-6553

August 5 through 8

 **American Bar Association (ABA)**
New York, NY
Contact: Cheryl Bunch
AppleLink: BUNCH1
(408)974-2853

August 25 through 30

Fera
Zurich, Switzerland
Contact: Kathrin Mader
AppleLink: CH.MARCOM

August 26 through 27

Electronic Book Fair
San Francisco, CA
Contact: Sandy Butler
AppleLink: BUTLER.S
(415) 904-0811

September 1993

 **IPEX**
Birmingham, UK
Apple Contact: Francine Rose
AppleLink: ROSE9
44-81-730-2481


September 14 through 20

 **Sonimag**
Barcelona, Spain
Contact: Yolanda De Juan
AppleLink: SPA.EVENTS

September 15 through 18

 **Apple Expo Paris**
Paris, France
Contact: Catherine Massot
AppleLink: FRA.PROMO
33-1-6986-3620


September 20 through 24

 **Data Kontor**
Stockholm, Sweden
Contact: Per Hedlund
AppleLink: HEDLUND1

September 26 through October 1

 **Oracle**
Orange County
Contact: Russ Havard
AppleLink: HAVARD1
(408) 974-4371


October 17 through 20

 **EDUCOM**
Cincinnati, OH
Contact: Eliza Lapé
AppleLink: ELIZA
(408) 974-1248

October 18 through 22

FY '94 Global Sales Conference
Las Vegas, NV
Private event by invitation
Contacts:
Marva Whelan (408) 974-8561
Gail Bridges (408) 974-3094
Al Hoodwin (408) 974-4419
Thad Carhart 33-1-4901-4720
AppleLink: WHELAN2, BRIDGES2 (Apple USA), HOODWIN (Apple Pacific), CARHART (Apple Europe)

October 27 through 29

 **ITTE/NSBA**
Dallas, TX
Contact: Jeryl Gerhardt
AppleLink: JERYL
(408) 974-2368

October 28 through 30

 **Seybold**
San Francisco, CA
Contact: Russ Havard
AppleLink: HAVARD1
(408) 974-4371