The Developer Business Report *January 1995*

AppleDirections

Macworld Developer Central
Come see the latest devloper products fro mApple Computer, Inc., at Developer Central, part of this year's San Francisco Macworld Expo, January 4-7. We're located in Hall E–North Hall of the Moscone Center. We'll be giveing away free copies of *Apple Directions* and *develop*, the Apple Technical Journal, and running demos of our latest tools and technology, so come on by!

Apple News

Announcing Pippin

It's called *Pippin,* and it's going to expand the market for your CD-ROM–based entertainment, education, and other multimedia products by reaching customers who have never before purchased computers.

The Pippin platform, just announced in Tokyo by Apple Computer, Inc., is a small CD-ROM playback device, derived from Macintosh architecture, that hooks up to customers' televisions or computer displays. The new platform is aimed at consumers who are looking for something in between today's action game players (such as Nintendo and Sega) and full-blown, more expensive personal computers. Its approximate $500 (¥50000 in Japan) price tag will give users inexpensive access to many of the benefits of Macintosh technology—especially entertainment and educational CDs—although it won't allow them to run traditional Macintosh productivity applications, such as word-processing, spreadsheet, and desktop publishing applications without, major modifications to them.

Apple has designed the Pippin platform to be licensed to other companies. The first licensee is Bandai Co. Ltd., a leading Japanese manufacturer of entertainment products, including the Power Rangers characters. Bandai expects to have the first Pippin-based product available by the 1995 holiday season. Apple expects other manufacturers from the consumer electronics, toy, computer,

Strategy Mosaic

From Developer to Developer: Go Native

By Paul Dreyfus, Apple Directions *staff*

Developers Share Their Power Macintosh Successes

Native Power Macintosh products are vital to your future.

We've been saying that in one way or another ever since we started talking about PowerPC processor–based RISC Macintosh systems a couple of years ago. We could make that point ourselves for the umpteenth time; that's how important it is. Instead, in this month's column, we're going to let some of your developer colleagues try to convince you that it's time to go native.

The Power Macintosh introduction has been a genuine success, at least according to these two important measures:

• Apple Computer, Inc., has sold approximately 750,000 Power Macintosh systems worldwide since the March introduction and is on target for meeting its goal of selling 1 million units in the first year, far more than the number of Intel-based Pentium PCs shipped. Included in those numbers are significant

AppleDirections ▶

Editor's Note

Games Lead the Way

It's a *great* time to develop Macintosh games. Allow me to explain:

Some years ago, I was shocked to learn that computer customers were complaining that there weren't enough good games for the Macintosh computer. This was at a time when Apple Computer, Inc., had been fighting hard to combat the perception that the user-friendly Macintosh computer was a toy and, thus, unfit for serious use, especially by business customers. I theorized that Apple had been so successful with its fight that most serious Macintosh game development had simply fallen away.

As folks used to say in the sixties, "What goes around, comes around." It's now apparent that games drive computer sales, at least in the home market. A recent survey conducted by Computer Intelligence-InfoCorp., a market research firm based in San Diego, California, shows that entertainment is by far the most popular category among home computer users. Seventy-one percent of the survey's 10,500 users said they used their computers most often for entertainment, followed by 46 percent for personal finances, 41 percent for school homework, and 38 percent for office work at home.

In other words, people are buying computers for their homes *in order to play games.* What this means is, with computer platforms, as in life, whoever has the most toys, wins—and the computer that gives people the most entertainment options has a strong advantage in the home computer market sweepstakes.

Fortunately for us all, Apple and some of you developers recognize that fact. Apple has just announced a new CD-ROM–based "edutainment" platform, Pippin, derived from Macintosh technology, that licensees will be able to sell cheaply to consumers. (See the news story on page 1.) The idea behind the new platform is to introduce new customers to Macintosh technology by letting them play games and run multimedia education titles; current Macintosh titles will run with little or no modification on Pippin. Then, once Pippin customers feel ready to make the plunge into full-blown personal computing, they'll buy the computer they already know. Macintosh

CD-ROM title sales have increased 400 percent in 1994 from the year before (while sales of Sega and Nintendo systems have declined), and Apple expects Pippin to take full advantage of the public's current yen for multimedia computing.

A variety of developers are also recognizing the opportunity to create Macintosh games. What I heard a few years ago is still true: there's a relative dearth of Macintosh games, and even fewer that run in "native" PowerPC RISC mode. Several developers, some of whom are new to the Macintosh platform, are working hard to fill that need, recognizing that Power Macintosh technology allows them to give customers features—and speed—they've never experienced before. They're also getting renewed technical and marketing support from Apple.

Says Apple's entertainment evangelist, Eric Klein, "The games space is attracting new developers to the Macintosh platform, and we're getting behind them." In this issue, One of those new developers, former PC-only developer Domark Software, talks about why they've embraced (their word) the platform so tightly. (See Strategy Mosaic, starting on page 1.)

What Apple, Domark, and other games developers are recognizing is that there is huge demand—and low supply—for quality entertainment Macintosh products. Somebody is going to make money with new products designed to meet that demand, and it might as well be you. In addition, by developing Macintosh games, you'll be serving the interests of all those who depend on the platform's success by giving home customers more reasons to buy Macintosh computers.

No, the Macintosh isn't a toy. But the more some of you can make it seem like one, by developing quality entertainment products, the better it will be for all of us.

Paul Dreyfus
Editor

P.S. If you want help with your entertainment products, Eric Klein will be happy to talk to you; send him an AppleLink message at KLEIN.E.

Filling Genuine Customer Needs

Prepared by the Apple Directions *staff*

Editor's Note: IndustryWatch is our regular compilation of news about events happening outside the Apple Computer, Inc., R&D complex and the ways we think they affect your development efforts. Each month, we gather the most notable items, which we hope will point you toward new opportunities, help you avoid mistakes, and alert you to key developments in the industry. We're not trying to cover everything in the computer industry; other publications already do that. Instead, we're digging through the news to present you with the most interesting tidbits, ones that translate into ideas you can use. If you have comments about IndustryWatch, if you want to alert us to issues you think the Apple development community needs to know about, send us an AppleLink message at A.DIRECTIONS.

Now Open for Shopping: The Net

Several influential companies recently announced plans to develop systems that will enable the buying and selling of goods over the Internet. In one scheme, Netscape Communications and First Data have joined forces on an online shopping service expected to debut in time for the 1994 holiday shopping wave.

Netscape, formerly Mosaic Communications, recently released a commercial version of the Mosaic client, the popular graphical-interface Internet browser, making it available for free over the Internet. The company, made up of some of the members of the original University of Illinois Mosaic development team, will sell World Wide Web server software that merchants can use to set up online stores. Customers can use the client to make purchases. First Data, which handles credit card transactions for hundreds of financial institutions, will take care of electronic transactions between customers and merchants. The system will use encryption technology to make sure information about customers and merchants remains secure and to keep hackers and others from sticking their hands into electronic tills.

Microsoft and Visa International have also teamed up to provide software for online shoppers. The software will let computer users shop electronically with their credit cards, authenticating the identities of buyers and sellers and preventing online thieves from stealing the proceeds of transactions. The software will be based on software encryption technology developed by RSA Data Security, which also provides encryption services for Apple's PowerShare collaboration server software. The Microsoft-Visa product is expected to ship in 1995.

Implications/Opinions: As we said in the Editor's Note two months ago (see *Apple Directions,* November 1994, page 2), shopping over the Internet is expected to increase phenomenally in the very near future. The announcements of these major players confirms that such online shopping will soon be available to millions of users around the world.

We wouldn't point this out if we didn't think there were an opportunity for you to join in. Only a small portion—approximately 6 percent, according to Apple data—of the world's 150 million personal computer users have access to the Internet. (See this month's Market Research Monthly on page 21 for data about Macintosh users on the electronic highway.) The companies that have already announced plans to provide online shopping products are getting in on the ground floor of what will ultimately be a very tall shopping mall. There'll be room for many others, as well, with quality products that provide online shopping services, such as customer ordering, advertising, merchant sales handling, billing, data security, accounting, cash management, inventory, and so on—in other words, the entire gamut of activities associated with traditional shopping at "real" stores.

There's a specific opportunity to provide security systems that are both secure and easy to use, as the above examples indicate. Data encryption solutions are key to both the Netscape-First Data and Microsoft-Visa partnerships; CommerceNet, a firm based in Menlo Park, California, and backed by computer makers and banks, is developing a standard for Internet electronic payments using RSA data encryption technology.

If you are at all inclined to develop products that support shopping on the Internet, there couldn't be a better time to get going. Selling a product into a growing market can help guarantee its success. And who knows? Your product could be that legendary Internet shopping package everybody talks about years from now as the product they wish they'd developed.

Mac OS—and Your Software—
Will Remain Easiest to Use

Symantec plans to release a new version of its PC Tools utility that incorporates Intel's Plug and Play for DOS and Windows software. This software helps DOS and Windows users combine hardware that is compatible with the new Plug and Play standard with hardware that is not. It will support Windows 95 (expected to ship in mid-1995) as well as the current versions of Windows and DOS.

Implications/Opinions: The need for this kind of utility reinforces the fact that the Plug and Play capabilities of Windows 95 will not completely address the problems of installation and configuration of Intel-based personal computers. For at least the next few years, the majority of Intel PC users will be running machines that aren't Plug and Play compatible, or will be running machines that contain peripheral cards and devices that aren't Plug and Play compatible. Although improved software utilities and "smarter" CPU/peripheral devices will significantly simplify PC installation and configuration, Intel-based PCs will still fall short of delivering Macintosh-style integration of hardware and software for some time. This means that the Macintosh computer, and your

applications, will continue to have a clear ease-of-use advantage, an advantage that's expected to leap with the release of the next version of the Mac OS.

Know Your Chinese Customers

Microsoft recently announced that it will develop a Chinese version of Windows 95 with substantial help from business and government in mainland China. The announcement comes only a few months after Microsoft released a Chinese version of Windows 3.1, which was met with a hail of official criticism because it had been developed in Taiwan instead of on the mainland.

Implications/Opinions: The government of mainland China is committed to using personal computers to help expand its economy, presenting clear opportunities for Western hardware and software developers. If you're considering entering this burgeoning market, you won't be alone. Apple has been targeting China for some time already, working closely with mainland Chinese institutions to develop appropriate software solutions. Apple recently beefed up its presence with a special publishing center (see the news story on page 10). As the above report shows, Microsoft is also interested in becoming a force there.

Anyone entering China will do well to learn from Microsoft's experience. You'll need to remember that there are are actually two Chinese markets—the island nation of Taiwan and mainland China. The two markets are divided politically and socially, and customers in them are likely to have very different needs. For example, the two nations' governments support different versions of the Chinese written language.

So be sure to study your customers well before making the critical error of insulting mainland Chinese with a product for the Taiwanese market (or vice versa). You'll also want to proceed very carefully for another reason: The software piracy rate in both Chinese markets is over 90 percent, according to the Software Publishers Association and other industry analysts.

Electronic Games Come of Age

Games magazine has named *Myst,* Cyan's HyperCard-based CD-ROM adventure, as its Game of the Year, the first time the magazine has ever honored an electronic product with its top award. Fully half of the magazine's annual list of the top 100 games, "The Games 100," is made up of electronic games, which is a 35 percent jump over the number of electronic games included in the 1993 list.

Implications/Opinions: This development suggests that a lot more creativity is going into the creation of electronic games than into traditional "physical" game design (the current phenomenon of *Magic: The Gathering* notwithstanding). Interestingly, though, there is a shortage of games for the Macintosh computer, especially "native" Power Macintosh games that take advantage of the performance of the PowerPC RISC microprocessor. Apple wants you to change that: By developing a Macintosh game, or porting one from another platform to work with the Macintosh computer, you could reap the benefits of filling a genuine customer need. See this month's Editor's Note on page 2 for more details. ♣

---

purchases by large businesses and institutions across the globe, including U.S. West, Honeywell, Holiday Inn, Mirror Group Newspapers in the UK, Quaker Oats, Drake University, and University of Oslo, to name a few.

• Thanks to many of your efforts, over 370 native PowerPC processor–based applications are available, with more than 100 each coming from Europe and Japan, and that number is growing every week. There are native products currently available in virtually every major application category.

Now, on to the stories of three successful Power Macintosh developers: Domark Software, a games developer who only recently shipped their first Macintosh

product; Fractal Design, whose engineers developed Painter; and Altsys, developers of Freehand. The bottom line for them is that Power Macintosh computers have proved a significant boon, and they can only urge the rest of you to take advantage of the opportunity to go native before the market is flooded with products.

The Power Macintosh Boon

Here is a summary of the points they made:

• Power Macintosh products will provide you with increased revenue opportunities, allowing you to capitalize on the current success of the Power Macintosh family of computers and related products. The developers we spoke with believed this early on, which is why they decided to go native so early; results of their Power Macintosh development efforts have since proven their early belief correct.

Another developer, ABVENT S.A. of France, reported an immediate impact on sales. Says company CEO Xavier Soule, "Until the May launch of our Power Macintosh native products, Zoom 4.5 and Atlantis Render 1.5, Apple's new technology was considered an incredible source of speed for our high-end applications. But it also proved to be powerful in other ways: In three months, we discovered an incredible acceleration of our cash flow, increasing our sales revenues by 23 percent in May and over 85 percent in June."

• The mere presence of Power Macintosh computers in the marketplace has refreshed developers' business prospects in ways they, and Apple, hadn't anticipated. The RISC-based systems are bringing new ideas to the platform, creating renewed excitement about developing Macintosh products.

• Developing Power Macintosh products is relatively

straightforward, even easy (particularly when compared with Windows development) if you follow a few simple rules. Our developers spoke of getting native Power Macintosh versions of their products up and running very soon after beginning the porting process. Fractal's story (corroborated by Altsys) demonstrates the ease of porting and contains general advice that you'll want to follow if you haven't yet begun the porting process.

A Warning

One other important point— actually a warning—emerged from what developers told us: If you don't develop Power Macintosh versions of your existing Macintosh products, your future will be increasingly uncertain. We say this not to scare you but just to alert you to a few realities. One of these realities is that, according to the developers we spoke with, software distributors are already

beginning to require that products they stock work with Power Macintosh systems.

Another reality: Apple is no longer designing new systems around 680x0 processors; by the end of 1995, it expects that more than 80 percent of the computers it sells will be equipped with PowerPC processors.

Also, it's becoming clearer all the time that RISC processing is the future of personal computing. Apple, IBM, and Motorola got the ball rolling in 1991 when they began collaborating on the PowerPC processor; Intel said this summer that they're working on next-generation RISC technology with Hewlett Packard; and last month, Apple, IBM, and Motorola announced that they're developing a standard RISC hardware platform to run multiple operating systems. The common denominator is RISC, and if your products don't take advantage of its inherent power and capacity for innovative solutions, they'll be left, literally, in the dust of warehouse shelves.

That said, here's what Domark, Fractal, and Altsys had to say about their experiences developing and marketing native Power Macintosh products.

Domark Software: From the PC to the Macintosh
Ten months ago, Domark Software had never released a product for the Macintosh computer; instead, they had spent the previous ten years developing games for other platforms, including Intel-based PCs, Amiga, and Sega/Nintendo. Now, says Paul Baldwin, public relations manager for Domark, "Thanks to Power Macintosh, we've totally embraced the Macintosh platform." The story behind their conversion shows how the power and potential innovation—not to mention new revenues—made possible by RISC technology are pumping new life into the entire Macintosh platform.

Domark was founded in the United Kingdom, and started developing and selling products in the United States in 1992. They were among the few games developers to decide early on to make native PowerPC software ready for the March 1994 Power Macintosh introduction, and their flight simulator for Intel-based PCs, Flying Nightmares, was ported and shrink-wrapped in time to be purchased by the first Power Macintosh customers.

Domark had no way of knowing what would happen. In just six months, they shipped over 40,000 copies of the game, more copies of the product than they would have sold of a successful PC title in one year. In one outlet, Computer City, Domark reports that Flying Nightmares was the second best-selling product behind the CD-ROM phenomenon, Myst. Baldwin says that, as a result of its unprecedented sales performance, "We expect Flying Nightmares to have a shelf life of a year or more, when the normal shelf life for a game is about six months."

So happy was Domark with their initial Macintosh product, which was a port from the PC side, that they added two Macintosh-only programmers to their ten-person engineering staff and started developing a handful of original games written just for the Macintosh computer. The first of these, Out of the Sun, has just been released; Absolute Zero and another four Macintosh products are expected to be released by next summer.

Macintosh Versus Windows Development
Domark's Macintosh conversion has been fueled mainly by their sales successes, but the company is equally excited by Macintosh technology. Domark Technical Manager Colin Boswell says that "Power Macintosh is an extremely simple machine to develop for." In our conversation with him, he

pointed to a number of technical advantages he feels the Macintosh platform has over Windows. (Our conversations with Domark also provided insight into some of the differences in marketing to the two platforms' customers, which are detailed in the text box "Macintosh Versus PC Marketing" on the next page.)

"With the Mac OS, you can depend on having your current product work two years from now with whatever version of the operating system is released then," Boswell says. On the other hand, with Windows, "It's just not that easy.

"The Macintosh operating system is easier to work with than Windows, which feels as if it's been thrown together out of bits and pieces," he continues. Aren't Windows 95 and the new Plug and Play standard supposed to change that? "We'll wait and see," he says.

Boswell adds that the well-defined application programming interfaces and managers of the Mac OS make it simpler to get a new game up and running. "For games developers, the optimal operating system is one that doesn't exist, so that you can hit the hardware directly for performance. With the Mac OS, you can write at a higher level first and see if the product will work in the first place. Then, once you've got it working, you can strip away code, get closer to the hardware, and work on performance. On a lot of other machines, you have to start from scratch with lower level code that takes more work, and it takes a lot longer to come up with a working prototype."

Macintosh Human Interface: Challenge and Opportunity
Some people argue that the Mac OS and Windows user experiences are converging, and that the Macintosh human interface

no longer clearly differentiates Macintosh systems. Boswell firmly disagrees: "Macintosh users have very specific expectations of what an application looks like and how it's used. It's very easy to make a mistake with the user interface, and it requires more work to get it right."

Boswell feels that the extra work is worth it; it translates into a product that's more appealing to users, and gives engineers more insight into the customer's needs. "At first, coming from the PC-DOS market, we didn't understand how vocal Macintosh users are about interface, but we found out that that's a tremendously positive thing. We've listened to users now, so we can appeal even more to what Macintosh users want.

"When we developed Out of the Sun, our new flight simulator, our first product written from the ground up for Macintosh, we hired an additional programmer just to work on the user interface, and we probably spent as much time on that as the rest of the product," Boswell says. "So, yes, it increased development time, but we have a much better product tailored to the Macintosh market. In fact, it's so tailored, we won't even release a PC product. And now, for our upcoming products, we have a much better idea of where to get started."

Writing Portable Code Is Key
With all the differences between the two platforms, Boswell says that writing the basic code for a game, whether it's for Macintosh or DOS/Windows systems, is remarkably similar. Domark uses ANSI C to write most of the code for their products, so that they may easily be ported from one platform to another. Boswell points out that about 90 percent of the code is identical in the Macintosh and PC versions of Flying Nightmares.

The developers we talked to for this report take a very similar approach, writing the basic code for their products in C or C++, as Apple has been advising developers to do since it first announced its PowerPC plans in 1992. This makes it possible for the code to be recompiled to run on different hardware platforms without too much difficulty. In fact, it appears that taking a cross-platform approach to coding is a key strategy for success; writing in ANSI C and taking a few precautions with your code allows you to use the same code base for multiple versions of your product. For more about those precautions, see the section on Fractal Design later in this article.

Boswell didn't report any special hurdles in porting Flying Nightmares to the Macintosh computer; he faced some of the same difficulties with low-level routines and system calls experienced by developers porting from 680x0 to PowerPC code. The process, from initial port to shrink-wrapped product, took them about six months. Boswell says that "Initially, moving from one platform to the other seemed like horrendous work. But starting off with extremely portable C code made it much easier than I thought. In hindsight, I'd say that doing it the first time was a bit of a pain, but it's gotten much easier since then."

The hardest aspect of the port for Domark? "Assuming that the 680x0 code would recompile the first time," says Boswell. "Using pointers, dialogs, and callback routines proved a bit of a difficulty, but in time we ironed things out so everything ended up working fine."

Supporting the 680x0 Installed Base

Like most Macintosh developers, Domark is facing the decision of whether to continue to support the 14 million-plus base of 680x0 Macintosh systems, still a vast market. Domark currently ships only one version of their products, using "fat" binaries that can install and run in both 680x0 and PowerPC processor–based Macintosh computers. According to registration card returns, approximately half of Domark's Macintosh customers own 680x0 systems.

Boswell says, "We're engaged in a big debate right now about how much to support 680x0 computers. Obviously, that's a big part of our market, and we wouldn't have reached them if we hadn't been so attracted to Power Macintosh. But we're beginning to do things with our new products just for Power Macintosh systems, and they won't work on 680x0 systems."

Domark's experience mirrors that of the other developers we talked to. No developers want to abandon the huge base of 680x0 customers, but they're beginning to design features and applications that work, or at least work well, only on Power Macintosh systems.

Summing up Domark's story, we can't help but feel good about having a PC developer so eagerly embrace the Macintosh computer and its customers. Domark Software is definitely a case of Power Macintosh computers bringing fresh ideas and enthusiasm—in the form of a new vendor—to the Macintosh platform.

Fractal: Power Brings Creativity to More Customers

You might remember the Fractal Painter porting rumor: When Fractal Design's engineers first decided to go native with Painter, their high-end paint package, they had a workable Power Macintosh version in less than two weeks. Talking to Mark Zimmer, a Fractal engineer, we discovered that the rumor wasn't quite true—but almost. Zimmer says, "It was a

Macintosh Versus PC Marketing

Domark Software's experience selling their flight simulators to the Macintosh market after years of selling to Intel-based PC customers explodes the traditional wisdom that there's always more revenue opportunity on the PC side of the fence. They're in the enviable position of having been a very early adopter of a successful new technology (that is, PowerPC), and their products fall into a category for which there's an apparent shortage (that is, Macintosh games).

Even taking into account those unusual advantages, Domark's Paul Baldwin says the company has found a number of benefits in marketing to Macintosh customers. Baldwin enumerated the following advantages:

• Marketing to Macintosh customers is cheaper because the market is smaller and there are fewer magazines and catalogs to hit with advertising. Because of this, Domark is able to focus more at the retail level, which helps build a presence more quickly, says Baldwin.

• Advertising can be more strategic, specifically aimed at Macintosh customers. Baldwin says, "Macintosh advertising isn't so much a shot in the dark."

• Apple user groups provide a direct route for marketing to Macintosh users at a grass-roots level.

• Development costs are still relatively cheaper for Domark's Macintosh games than for their PC products.

• The Macintosh user interface results in a better quality product that's targeted more successfully to its audience.

• The Macintosh market is much more open, at least for a games developer focusing on Power Macintosh products, while the PC side of the market is saturated with products. Baldwin says that Domark's Flying Nightmares product was the first new flight simulator for Macintosh systems to be released in a year, and that currently there are very few native games for Power Macintosh users.

This last fact will undoubtedly change, as more Power Macintosh games become available. But Baldwin thinks that, at least as far as Domark is concerned, "The more games, the better." This is because quality games help sell the computers that run them—games are the applications most often used in the home market (see the Editor's Note on page 2). According to Baldwin, if there are more games available, it will boost the entire platform and result in more customers for Domark.

Finally, Baldwin points to the marketing help Apple has provided Domark as a reason for their almost overnight success with their first Macintosh products. "Apple has helped us tremendously," says Baldwin. "Our programmers talked to folks at Apple and received a lot of useful feedback, and we've done some joint marketing. For example, a demo of Flying Nightmares was included on the CD that shipped with the first 250,000 Power Macintosh systems that went out to resellers."

Arguably, Domark's experience is unique because the company happened to be an early adopter. But Baldwin feels that other developers will be able to benefit from the Macintosh market's advantages. "Lots of developers just aren't going over to Macintosh," he says. "But there's a window of opportunity, and we think it's going to remain open for a while."

quick port; we hardly had to do anything to get it to work." But before recompiling Painter to run on Power Macintosh systems, Fractal's engineers had taken some important initial steps to be sure the compiling process would go smoothly.

Like Domark and Altsys, Fractal took a cross-platform approach since they were also porting the product to Windows, writing the basic Painter code as much as possible in ANSI C and making sure it was 32-bit clean. Additionally, Fractal engineers use prototypes in their code, which they feel makes checking procedure calls about twice as fast. As Zimmer says, "When you're maintaining a product on several platforms, you have to be careful about your code."

Fractal engineers were careful to write code that made limited use of platform-specific routines, using system calls and low-memory globals only when absolutely necessary. To make sure that Painter would run properly on Power Macintosh computers, they provided descriptors for callback routines, a step that's necessary because Power Macintosh systems currently employ a mixed-mode architecture, with part of the system software running in native PowerPC mode, part in 680x0 emulator mode.

(For more about routine descriptors and other detailed aspects of writing PowerPC code, see "Understanding the Power Macintosh Architecture," parts one and two, in the April and May 1994 issues of *Apple Directions,* which can be found on the September 1994 Reference Library Edition of the Developer CD [path—Periodicals:Apple Directions:Apple Directions 1994: Apple Directions 04/94 and Apple Directions 05/94:Technology].)

Zimmer says, "It took us about six days of significant effort to get Fractal's 300,000 lines of code ready for a different compiler.

Once we recompiled, we had a demonstrable version of the product ready in a very short time. We hardly had to do anything else."

Once the port was complete, the Power Macintosh version of Painter impressed even Fractal's engineers. Zimmer says, "Parts of the program were running much faster than we'd predicted they would. The brush now runs faster than on any other platform, and the features that use the floating-point unit—like lighting, surface texture, and image warp—are exceptionally fast.

"At a demo we did in Seybold, we showed Painter running side by side on a Macintosh Quadra 840AV and a Power Macintosh 8100; some operations ran ten times faster on the Power Macintosh computer."

Unprecedented Sales With Painter for Power Macintosh

Since issuing their Power Macintosh version of Painter (actually, Painter also ships as a single "fat" application that can run on both 680x0 and Power Macintosh systems), Fractal has experienced "unprecedented sales," according to Zimmer. "We're still in the early days of Power Macintosh, and right now the marketing opportunity for us is remarkable."

The reason: Painter is a product that's previously been aimed primarily at high-end graphics users with more powerful computers, but Power Macintosh systems now make that kind of power available to more customers. "Painter uses more cycles than your average program," says Zimmer. "Power Macintosh computers have raised the performance bar, meaning that an increasing amount of the Macintosh customer base will be able to use our product. As a result, we're 'consumerizing' Painter, considering a number of ways of making it useful to more people."

One of those ways has been to ship the current version of Painter—version 3.0—in a configuration that runs in smaller memory. Another way has been to reuse some of the Painter code in a consumer-oriented paint package called Dabbler, which Apple is now bundling with Macintosh Performa 6100 systems, its Power-PC 601 processor–based line of home computers.

For Fractal, the opportunities created by Power Macintosh products are more than just increased revenues. Zimmer says, "When you sell to consumers, your product has to be more than just fun; it has to teach people something, to allow them to be creative.

"Creativity seems to be something that's not being taught as much right now because of government cuts. Something has to come along to replace that and to provide the tools that help people learn creativity and be creative." For Fractal, Power Macintosh computers and their paint products are among those tools.

Altsys and Freehand: Simple Port, Pleasing Performance Boost

What Fractal describes about their experiences porting to PowerPC native mode is typical of other developers' porting stories. If Fractal had an easier time porting than others did, it was only because they took a cross-platform approach from the word go with an earlier 680x0-based version of Painter.

Altsys, another developer we talked to, reported taking similar precautions with the code for Freehand before recompiling it to work with Power Macintosh systems (as well as with Windows, NextStep, and Solaris). Eighty percent of Freehand 4.0 was written in C so that it could be ported easily; that portion of the code recompiled the first time they tried it. Says Steven Johnson, senior software engineer at Altsys,

"We were 90 percent done with our port in two weeks. The rest of the work was just a lot of tweaking and testing. Porting was pretty straightforward, even with the limited tools available before the Power Macintosh introduction in March. Now, the tools are better, and porting is even easier."

One issue for Johnson and Altsys was determining how much of the 680x0 assembly-language Freehand code to rewrite in C so that it could be portable, and how much to rewrite in the assemblers for each of the new platforms they were porting to. "It's very difficult to maintain common code if a lot of it's written in assembler, because then we'd have to rewrite it for 680x0, PowerPC, Intel, and so on," Johnson says. "We had to balance the cost of the extra work required by writing in assembly language against the benefit it might bring to the product itself. Ultimately, we decided to use C for most of the code, although some time-crucial things are still written in assembly."

Another issue Altsys had to resolve when they ported Freehand: how to optimally use floating-point unit (FPU) calculations, which can provide performance ten times that of the 68040 processor. "The current PowerPC version of Freehand is not FPU-intensive, because the same basic code has to be used for our 680x0 version, and you can't rely on the FPU being present in most 680x0 systems," says Johnson. "Once the installed base moves more to PowerPC, we'll see expanded use of the FPU and lots of other features that just won't work on 680x0 Macintosh computers."

Even without using the FPU extensively, the Power Macintosh version of Freehand received a "very pleasant" performance boost, which Johnson says is "a huge win for our customers."

"There's really no comparing the older version with the Power-PC version of Freehand," says

Johnson. His advice to other developers: "I don't understand why everyone isn't ready with Power Macintosh products right now, and I wish more people would get going. Power Macintosh computers give developers and customers a good, fast alternative."

Go Make Your Own

In the competitive software business, if you're not growing, you're in trouble. We think Power Macintosh products provide you with significant growth opportunities, as our developers' stories strongly suggest.

As we said earlier, most of the Macintosh product line, from low-end PowerBook systems to high-end desktop models, will be driven by PowerPC processors by the end of next year. Apple has already released Power Macintosh systems for the home market (the Macintosh Performa 6100 series) and will soon have PowerPC processor–based systems designed to meet the needs of customers in other key markets. To come right to the point, PowerPC technology will soon pervade the Macintosh installed base, and you'll need native Power Macintosh products if you want to keep your business growing alongside Apple's.

We turned the podium over to some of your colleagues this month to convince you of the ease of developing native Power Macintosh applications and the opportunities they'll create for you. We hope you liked their stories—but, to paraphrase the esoteric radio commentator Scoop Nisker, whether you liked them or not, go out and make some of your own. ♣

---

publishing, telephone network, and cable TV industries to create their own versions of Pippin tailored for their specific markets and channels.

Pippin will include the following hardware features:
  • PowerPC RISC processor
  • quadruple speed CD-ROM drive
  • support for NTSC, PAL S-Video, and VGA displays
  • ample system memory for playing most existing CD-based Macintosh titles
  • dual stereo input/output with CD-quality audio
  • support for up to four Apple Desktop Bus input controllers (Pippin will not ship with a keyboard; a prototype was pictured with a multibutton keypad)
  • on-board memory for saving and restoring games (Pippin will not ship with floppy or hard-disk drives)
  • it will be expandable with many after-market add-ons that will be available in 1996

On the software side, Pippin will employ a subset of the Mac OS, and most existing CD-based titles will run on it with little or no modification. We'll tell you what that means as soon as we can: Pippin was announced just as this month's *Apple Directions* went to press, so we haven't been able to learn what you'll have to do to make your current products work with the new platform. We can tell you, however, that an Apple software engineer spent only four hours editing more than 15 existing Macintosh titles so that they could run on Pippin prototypes. As soon as more technical details are available, we'll pass them on to you.

Macintosh Brand Is Strongest in the Industry, According to DG Study

Macintosh users remain powerfully loyal to their computers, according to a recent study released by International Data Group (IDG), a leading personal computer market research firm. The study also suggested that it would be far easier to convince DOS and Windows users to switch to different computers than it would be to convince Macintosh users to try a new system.

The study of the relative power of different computer brands rated the Macintosh computer as the strongest brand of all those included in the study, as indicated by a number of factors. First, participants in the study were asked how much less another computer would have to cost before they would consider buying it. Macintosh users said they would consider buying another brand only if it cost at least $606 less than a Macintosh computer. It would be far easier to persuade users of IBM-compatible DOS and Windows systems to consider buying other computers: They'd consider buying other systems if they cost just $332 less than their existing computers.

Additionally, Macintosh desktop computers placed highest on the so-called Brand Power Index established by the study, which was conducted by Griggs-Anderson Research for IDG. The Brand Power Index combines data that measures the following key characteristics of a successful brand:
  • Price premium—Are buyers willing to pay more for their favorite brand? If so, how much more?
  • Commitment—Do buyers intend to repurchase this brand? Would they be inclined to look at their preferred brand first?
  • Advocacy—Do buyers strongly recommend their preferred brand to others? Have they become missionaries for the brand? Would buyers fight for their brand if their bosses wanted to invest in another brand?

Macintosh desktop computers scored highest on the index, with a composite score of 74, while PowerBook systems came in third with a score of 71. The only other computer that scored in the top ten on the index, which was also used to rate peripheral devices such as printers and software products, was the Compaq desktop computer, which placed tenth with an index score of 67.

Also, when asked, "Would you take a strong stand if your boss opposed buying your brand?", 52 percent of Macintosh customers said they'd fight before they switched, while only 11 percent of IBM and Compaq users said they'd be willing to do so.

Griggs-Anderson Research conducted the survey by completing in-depth phone interviews with 1,641 buyers and users, including 640 computer professionals, 455 managers, and 546 home users. Of the 1,095 business users surveyed, 329 were users at small companies and 647 were users at large companies of 100 or more employees.

Macintosh Computer Is Top-Selling U.S. PC in Summer '94

Recent industry reports confirm that Apple Computer, Inc., is maintaining its sales momentum. Apple was the top selling vendor of personal computers in the United States for the third quarter of 1994, according to the market research firm Dataquest. Dataquest said that Apple shipped the largest number of personal computers in the United States from July through September 1994, capturing 13.1 percent market share. Trailing Apple were, in order, Compaq, Packard Bell, IBM, and Gateway 2000.

"In the third calendar quarter, Apple experienced strong acceptance in our target markets in the United States," said Jim Buckley, president of Apple USA. "We were particularly pleased with the demand for our Macintosh 630, PowerBook 500, and Power Macintosh products. And with the momentum of our Power Macintosh line, Apple is on track to ship one million Power Macintosh computers worldwide in the first year of production. This is clear evidence that our customer- and solutions-focused initiatives are starting to reap rewards."

During the same period, Apple reported its highest quarterly revenues in the company's history and quarterly worldwide unit shipments in excess of one million units for the second time in company history.

Windows/DOS Support for PowerPC and 68040 Macintosh Models

Last month at Fall Comdex '94, Apple Computer, Inc., announced several products that will make it possible for many Power Macintosh customers to run DOS and Microsoft Windows software using a 66-MHz Intel 80486 DX2 processor. Apple also announced that it is licensing this technology to Reply Corporation, which will sell add-on cards to give similar capabilities to a variety of popular 68040-based Macintosh computers.

DOS-Compatible Power Macintosh 6100, DOS Compatibility Card

Two of Apple's Comdex announcements were for the Power Macintosh 6100 DOS Compatible system and the DOS Compatibility Card for Power Macintosh 6100 and Macintosh Performa 6100 series computers. These new products combine the high price/performance value of Apple's RISC-based Power Macintosh computers with the capabilities of a Windows- and DOS-based personal computer. Apple plans to begin shipments of the Power Macintosh 6100 DOS Compatible system and DOS Compatibility Card in the first calendar quarter of 1995.

Powered by a PowerPC 601 microprocessor and a 80486 DX2/66 microprocessor, the Power Macintosh DOS Compatible system runs more applications than any other personal computer on the market. Users can easily switch between the Macintosh and DOS computing environments with a simple keyboard command. Both processors can run in tandem, and users can even cut and paste data between Macintosh, Windows, and DOS applications. Dual monitor support allows customers to view the Macintosh and DOS environments simultaneously without purchasing an additional video card. The Power Macintosh 6100 DOS Compatible system supports most VGA, SVGA, and multisync monitors, as well as all current Apple displays.

The Power Macintosh DOS Compatible system runs the widest range of mainstream PC software—including DOS, Windows, and Macintosh applications—in addition to the more than 370 "native" applications that have been optimized for Power Macintosh. Apple believes that this computer should attract various customers who have previously not considered the Macintosh platform, including users who are attracted to the Macintosh platform but need to run existing DOS and Windows software and first-time buyers uncertain of their computing needs.

The Power Macintosh 6100 DOS Compatible system improves on the technology of the popular 68040-based Macintosh Quadra 610 DOS Compatible, which was awarded "Best New System" by *Byte* magazine at Fall Comdex '93. Featuring a faster processor than its predecessor, the Power Macintosh 6100 DOS Compatible also offers enhanced multimedia capabilities, such as built-in support of Creative Labs' Sound Blaster chipset for 16-bit audio playback, as well as improved networking compatibility with support for NetWare IPX and TCP/IP network protocols by means of an ODI-compliant driver. The optional double-speed internal CD-ROM drive runs Macintosh, DOS, and Windows CD-ROM discs, providing users with access to the hundreds of PC-based CD-ROM titles currently available.

Reply Corporation Licenses DOS Compatibility Card Technology

Apple also announced that Reply Corporation, a manufacturer of system enhancement products based in San Jose, California, has become the first licensee of Apple's DOS Compatibility Card technology. Reply intends to manufacture and market DOS-compatible cards for a range of Macintosh systems including the Macintosh Quadra 610, 650, 660AV, 700, 800, 900, and 950 models, and the Macintosh Centris 610 and 650 models.

Availability and Pricing

The Power Macintosh 6100 DOS Compatible system is expected to be configured with 16 MB of RAM, an Apple SuperDrive floppy disk drive, a 350 MB hard disk, and a built-in Ethernet port, and is anticipated to have a U.S. Apple price of $2,499. The DOS Compatibility Card may be installed in Power Macintosh 6100 systems or Performa 6100 series products and is expected to have a U.S. Apple price of $699. Both the system and card are expected to be available in the first calendar quarter of 1995. Specific pricing and availability will be announced at a future date.

Apple Expands Newton Developer Support

Apple recently announced its expansion of Newton developer support. It's now offering two Newton programs: a high-end developer support program, whose annual cost has been

lowered from $2850 to $2500, and a new low-cost program ($400 per year) for developers who do not require direct e-mail technical support. Both programs are designed to support commercial, in-house, and vertical-market developers, system integrators (SIs), and value-added resellers (VARs). Both programs have also been enhanced with new features, including a Newton Orientation Kit, to get developers new to the program started, and a monthly Newton Developer Mailing.

Currently, the *Newton programs* are available only to developers in the United States and Canada, although it's expected that they'll soon be offered elsewhere.

The new low-cost program, called the *Newton Associates Program,* includes the following benefits:

• *Core developer support services from the Developer Support Center.* This includes the developer handbook, development hardware purchase privileges, an orientation kit, Worldwide Developers Conference tickets, access to the Third-Party Compatibility test lab, the AppleLink publishing program, and discounted rates on AppleLink online service.

• *The Newton Orientation Kit.* This packet includes the latest *Newton Developer CD* and other technical, marketing, and product materials to get developers started on the Newton platform.

• *Newton Developer Mailing.* Participants will receive a monthly Newton-focused mailing that includes a quarterly *Newton Developer CD,* the bimonthly *Newton Technology Journal,* a monthly copy of *Apple Directions,* and other timely information on Newton technology. These periodicals contain Newton sample code, human interface guidelines, programming

advice, Newton product news, and relevant market research to help developers make informed decisions about Newton-based projects.

• *Online technical information.* Through AppleLink, participants will have access to the latest technical and marketing information and access to restricted Newton bulletin boards (including a bulletin board for communication with other Newton developers).

• *The Technical Q&A Reference Library.* These documents, which are updated periodically, give participants quick access to commonly asked development questions and can save valuable development time.

• *Access to Apple's third-party compatibility test lab.* Participants can use this Cupertino lab to test products with different configurations of Macintosh and Newton hardware and software.

• *Hardware discounts.* For development-related purposes only, Newton Associates are eligible to receive discounts on a limited number of Macintosh and Newton systems per year.

• *Newton development class discounts.* Participants receive two coupons each redeemable for $100 off any Newton development class at Developer University. They also receive the *Developer University Catalog* of classroom and self-paced developer courses.

• *Developer conference invitations.* Participants are eligible to attend the Newton Developer Conference and Apple's Annual Worldwide Developers Conference (WWDC) for an additional fee. These events allow them to meet other developers and hear details about Apple's business and technology directions.

• *The StarCore Affiliate Label Program.* This program helps developers with various aspects of product production, marketing, and worldwide distribution.

The Newton Partners Program (which costs $2500 per year) replaces the more expensive PIE Partners Program (which cost $2850). This program includes all the features of the Newton Associates Program along with the following benefits:

• *Newton programming-level support.* Newton Partners can submit technical questions by e-mail and get direct support from experienced Newton development engineers.

• *Additional development hardware purchasing privileges.* In addition to the hardware purchasing privileges available to Newton Associates, Newton Partners are eligible to purchase an additional five Newton systems per year at the developer discount.

• *Test site eligibility.* Based on the constraints of Apple's testing objectives and product availability, participants are eligible to receive certain pre-release hardware units and software releases.

• *Selected marketing opportunities.* Partners have the opportunity to participate in selected marketing and public-relations events that are run through the Newton group. At Apple's invitation, typical activities may include listings in a Newton *Solutions Guide,* participation in Apple's press activities, and exposure through Apple point-of-purchase materials. (Some of these programs may require additional fees.)

The Apple Developer Programs

The Apple Developer Programs constitute Apple's official support for all hardware and software developers who use Apple technologies and products. Apple has separate programs for its Macintosh, Newton, and multimedia platforms. As with Newton support, the Macintosh support programs make available two levels of support (Associate and Partner).

For More Information

For more information about the Newton Associates and Partners Programs, the Macintosh Associates and Partners Programs, or the Apple Multimedia Program, developers in the United States and Canada should contact the Developer Support Center (phone: 408-974-4897; AppleLink: DEVSUPPORT; Internet: devsupport@applelink.apple.com). Developers outside the United States and Canada should contact their local Apple office or distributor for information about local programs.

Apple Expands Influence Worldwide

Apple Computer, Inc., is growing—in more ways than one. Three recent announcements point to Apple's expanded geographical reach and increased sales of Macintosh computers.

Latin America. Apple recently opened a new Miami office as part of its continuing expansion in the Latin American market. The office, located in Coral Gables, Florida, was officially opened by Apple Latin America/Caribbean General Manager, Luis Rubio.

Apple Latin America/ Caribbean provides sales, marketing, distribution and training for its 19 resellers to 44 countries throughout Latin America and the Caribbean. Apple is also operating new warehousing facilities in Miami to enable customers to benefit from product resources that are closer to the marketplace. Apple is also working to make its products more available in these markets, help develop local software solutions, and provide instructional seminars to customers.

China. As a measure of Apple's commitment to the Chinese market, Apple CEO Michael Spindler recently opened the new Apple Publishing Center in Beijing, China. This center is a team effort between Apple and the China Research Institute of Printing Science and Technology (CRIPST), a research organization under the Press and Publication Administration of China. Fifteen additional vendors are committing hardware and software resources to the publishing center: Adobe, Quark, AGFA, Linotype-Hell, Scitex, DynaLab, Hanyi, Monotype, Newgen, Trend, Radius, Microtek, Sinotype, WYS System Ltd., and Tektronix. The Center will offer a wide range of services, including system integration, consulting, training and high-quality output services.

Apple has launched a number of business initiatives over the past two years that stress its long-term commitment to the Chinese market. This includes the development of a Chinese version of the Macintosh operating system, as well as the founding of the Developer Services Program at the Apple Technology Forum in Beijing last June, the establishment of a training center at Beijing's Tsinghua University in September of last year, and the recent collaboration with the Weifang Huaguang Electronic Group to set up a software development center.

New manufacturing facility. To help meet demand for Macintosh computers, Apple is building a new 200,000 square-foot logic board manufacturing facility in Elk Grove, California. This facility, which is adjacent to an existing Apple site that performs system assembly, distribution, and service logistics functions, is expected to be operational in late 1995 and will result in approximately 300 new jobs.

QuickTime 2.0 for Windows Delivers Full-Screen Video and More

Advancing the company's strategy to offer cross-platform multimedia standards, Apple Computer, Inc., recently announced the availability of QuickTime 2.0 for Windows. QuickTime 2.0 for Windows offers developers a cross-platform standard for creating, using, and sharing multimedia content between Macintosh and Windows-based personal computers. In addition, it offers Windows-based PC consumers higher-quality video and other multimedia features not currently possible with other Windows-based multimedia software.

Benefits for Developers
Using QuickTime, developers can create multimedia titles once and deliver them to many platforms, including Macintosh, Windows, interactive television, and Fujitsu's FM Towns. This helps multimedia developers reduce the development time and expense necessary to rework titles for different computer platforms.

Already a number of developers have agreed to use QuickTime 2.0 for Windows in their upcoming products, including Simon & Schuster Interactive for its new *Star Trek: The Next Generation Technical Manual.* Voyager, a New York-based multimedia developer, is also utilizing Quick-Time 2.0 for Windows in its new titles *Spinal Tap* and *For All Mankind.* Both Simon & Schuster Interactive's and Voyager's products take advantage of QuickTime 2.0 for Windows's full-screen video and text capabilities—features not available in other multimedia software products.

More Than Just Video
QuickTime 2.0 for Windows includes support for full-motion, full-screen video, and supports MPEG (Motion Picture Experts Group), a standard that allows for smoother, higher-quality video. Unlike the "postage-stamp" sized video of years past, QuickTime 2.0 for Windows now makes it possible for users to play full-motion video even on entry-level personal computers. For example, it's now possible to play video at 30 frames per second on a 25-MHz 486 computer with just 4 MB of RAM.

QuickTime 2.0 for Windows integrates and synchronizes photographs, music, computer animation, musical scores, CD-quality audio, text, and digital video, opening up new creative possibilities and reducing production time.

QuickTime 2.0 for Windows has new music capabilities that simplify the inclusion of music with CD-ROM titles and presentations—without requiring a technical understanding of MIDI technology. Using the capabilities of MIDI and sound add-on cards, QuickTime synchronizes MIDI-generated music with video and digital audio. QuickTime 2.0 for Windows also features audio compression, which brings CD-quality audio into multimedia presentations—while using just one-fourth of the space required by uncompressed audio.

It is also easier to incorporate and use words within reference titles and interactive books using QuickTime 2.0 for Windows, because of its support for searchable, anti-aliased text.

In addition, the compression capabilities of QuickTime 2.0 for Windows enable customers with limited hard disk space and no CD-ROM drive to take advantage of multimedia creation and playback. For example, a 20-slide multimedia sales presentation created in QuickTime 2.0 for Windows

with compressed high-resolution images, music, and text can fit on a single 1 MB floppy disk.

Developer Availability and Pricing
Developers can license Quick-Time 2.0 for Windows immediately for redistribution with applications, titles, and media clip libraries that support QuickTime. The cost is $300 per year, per title. Developers can license both the Macintosh and Windows versions for $400 per year, per title. For licensing information, contact Apple Software Licensing by phone at 512-919-2645 or by AppleLink at SW.LICENSE.

The QuickTime 2.0 for Windows Software Developer's Kit is expected to be available in early 1995 for $195. To order, call APDA at 800-282-2732 from the United States, 800-637-0029 from Canada, or 716-871-6555 from other countries.

Online Distribution for End-Users
QuickTime 2.0 for Macintosh and QuickTime 2.0 for Windows are available online in The Store on ZiffNet/Mac and ZiffNet, respectively. ZiffNet/Mac and ZiffNet are the online information services all about computing, from Ziff-Davis Interactive, hosted on the CompuServe Information Service. ZiffNet/Mac members can type "GO MACSTORE" and ZiffNet members can type "GO STORE" to download the new QuickTime 2.0 extension, a musical instruments extension, sample music movies, and version 1.0 of Apple's Movie Player application, for $14.95. The QuickTime for Windows version will have a special introductory price of $7.95 through the end of the year. An alternative package for Power Macintosh users also includes the QuickTime PowerPlug for full-speed PowerPC performance. ♣

AppleDirections

Technology

CD Highlights

System Software Edition, January 1995

I'm writing this after having just returned from the European Developer Forum in Kista, Sweden, where I was able to schmooze with developers from throughout Scandinavia and central Europe. I have concluded that European developers are, with a few minor exceptions (better taste in clothes, and the fact that everyone seems to *like* DocViewer) very similar to U.S. developers; the major difference being, of course, their perception of the importance of localization. While localization is a "should do" for U.S. developers, it is a "must do" for European developers, who must often ship multiple language versions even within a single country.

An important part of this process is timely access to localized system software and extensions, and I shall try to bring you more, and more quickly, in 1995. For a start, look for newly localized system software in the What's New folders of the Tool Chest and Reference Library editions of the CD, beginning next month. Also, we're trying a new delivery format this month: Most system software will appear as disk images, and we have included a new image-mounting utility, DropDisk. Unlike other disk-image utilities, DropDisk does not load the images into RAM; thus, it allows you to mount (for instance) the entire 30-disk set of Traditional Chinese 7.1.2 disks at once to perform network installations. This will also make it easier for the 40 percent or so of developers who prefer disk images. See the file About DropDisk in the Utilities:DropDisk 1.0b4 folder for details.

Myths About Europe
Myth: Sending your employees to conferences in Europe is wasteful, unnecessary junketing.

Fact: Such conferences are quite useful and informative, and will result in noticeably improved performance levels from your employees. Plus the beer is way better.

Myth: There's plenty of full frontal nudity on European television.

Fact: A quick bout of channel-surfing revealed very little full-frontal nudity. However, old American sitcoms are much better when dubbed into a language you don't understand.

So, in addition to various localized versions of system software 7.1.2 and 7.5, here are this month's new and revised packages.

Developer Notes Update
The *Display Device Driver Guide* describes how Display Manager 1.0 communicates with display devices. It provides the details necessary for programming display device drivers that fully support the Display Manager features on the Macintosh platform.

This updated version of the *Display Device Driver Guide* includes new sample functions for the control and status calls. The sample functions show how to use the low-level PBControlSync and PBStatusSync routines to set up a parameter block containing the appropriate control and status information about a video display device for the Display Manager. The new sample functions are in the "Control Requests" and "Status Requests" sections of the guide.

Inside Macintosh Errata
This folder contains technical notes documenting errors or omissions in four of the New Inside Macintosh books: *Inside Macin-tosh: Processes, Inside Macintosh: Memory, Inside Macintosh: Overview,* and *Inside Macintosh: PowerPC System Software.*

Installer 4.0.3 SDK
Installer 4.0.3 provides the latest Apple technology for installing and upgrading software to both 680x0 and PowerPC processor–based Macintosh systems.

Version 4.0.3 is a bug-fix release of the Apple multidisk installer that also offers the new InstaCompOne decompression atom extender. The revised documentation addresses in greater detail the questions and issues that were raised with Installer version 4.0.

This particular update of the Installer SDK includes minor corrections to two of the example scripts and includes all available localized releases of Installer 4.0.3. Read the document Installer 4.0.3 SDK Rev History for a list of changes from the previous release of this SDK.

MacODBC SDK
The Macintosh Open Database Connectivity SDK allows application developers to connect to many databases through one common application programming interface. It also allows database developers to write database and translation drivers that work with ODBC-aware applications. Licensing for redistribution to end-users is available only through purchase of the Mac OS SDK, sold separately.

Coming Next Month
Next month's CD will feature a new version of ASLM, snippets galore, yet another gnarly application framework, and much, much more.

Alex Dosher
Developer CD Leader

The New Macintosh Run-Time Environment

What You Need to Know About the Code Fragment Manager, Mixed Mode Manager, and System Object Model

By Gregg Williams,
Apple Directions *staff*

An important transition is coming to the Mac OS: a new run-time architecture. It started with system software changes needed to accommodate the PowerPC processor, and it's about to move to the 680x0. Your software must use this new run-time architecture to benefit from some of the new features in upcoming versions of the Mac OS, but there's a more immediate reason for using it—OpenDoc.

This article discusses the three components of the new run-time architecture—the Code Fragment Manager and Mixed Mode Manager for the 680x0 processor and the System Object Model for both the 680x0 and PowerPC processors. It describes what they do for you, and explains what you need to do to use them.

What Is CFM-68K?

The Power Macintosh software architecture is based on *fragments,* which constitute units of executable code and their associated data. All Power Macintosh software is loaded into memory as fragments—applications, extensions, code resources, and import libraries—and along with it comes the Code Fragment Manager. (For more information, see "Understanding the Power Macintosh Architecture, Part One: Overview," in the April 1994 issue of *Apple Directions*.)

The Code Fragment Manager for 680x0 (commonly abbreviated as *CFM-68K*) brings the code-fragment architecture to 680x0-based Mac OS computers. It works on any Macintosh computer that is running System 7.1 or

later, has a 68020 (or faster) processor, and has 32-bit addressing turned on.

Advantages of CFM-68K

The most immediate advantage of using CFM-68K is that it brings import libraries (also called *shared libraries*) to 680x0-based Macintosh computers. (Import libraries are a very important part of Power Macintosh software; see the "Understanding the Power Macintosh Architecture" article, mentioned earlier, for details.) Import libraries allow multiple programs to use the same library of routines, thus saving disk space and memory.

In the immediate future, CFM-68K will be important because it will enable 680x0-based Macintosh computers to run OpenDoc, Apple's component-software architecture. OpenDoc uses import libraries and requires the presence of the System Object Model (SOM), which is implemented on the Macintosh platform as a shared library.

Further down the road, Apple will be making major enhancements to the Mac OS, some of which are tied to this new fragment-based run-time architecture. By using CFM-68K, your software will be better prepared to take advantage of new features in upcoming versions of the Mac OS.

A final benefit accrues from having both the Code Fragment Manager and the Mixed Mode Manager (discussed later) on both the PowerPC and 680x0 platforms. Currently, in a number of situations, you must write different code for the 680x0 and PowerPC versions of your program. To maintain one set of source code, you currently bracket the two

chunks of code with conditional compilation commands that have the following effect:

> *if compiling PowerPC*
>   *<PowerPC code here>*
> *else*
>   *<680x0 code here>*
> *end*

Once the Code Fragment Manager and Mixed Mode Manager are available on both the 680x0 and PowerPC platforms, you will be able to call a routine the same way for both of them, and the above code can be replaced with one chunk of code that will compile correctly for both environments. Because the Code Fragment Manager and Mixed Mode Manager work the same way on the 680x0 side as on the PowerPC side, you can use their respective chapters in *Inside Macintosh: PowerPC System Software* to learn how to use them.

CFM-68K Details

In the Power Macintosh world, the code-fragment architecture means that Power Macintosh programs have no segmentation—the Code Fragment Manager loads fragments into memory as they are needed. On the 680x0 side, for the first time, you will be able to write unsegmented applications, by putting the entire application in a single code fragment. (Note, however, that the build tools default to build segmented applications.) If you want, you can still build segmented applications to minimize your application's footprint.

CFM-68K import libraries are not segmented; in fact, because 680x0 System 7 does not have file mapping (which is present in the Power Macintosh implementation of System 7), a CFM-68K application must be able to load into

memory all the import libraries it uses.

Another limitation of CFM-68K is that applications and import libraries created with it will not run in emulation on Power Macintosh computers. This means that if you are using CFM-68K and want one application or import library that will run on all Macintosh models, you will have to package both 680x0 and PowerPC versions into one "fat" file.

What Is MMM-68K?

The second piece of the new runtime architecture is the Mixed Mode Manager for 680x0 (abbreviated *MMM-68K*). MMM-68K performs an analogous but not identical function as the Mixed Mode Manager for Power Macintosh computers. On Power Macintosh computers, some system software (written in PowerPC code) assumes the new run-time architecture, and some (written in 680x0 code) assumes the old run-time architecture. The Mixed Mode Manager switches between executing these two types of code and invokes the 68LC040 Emulator when 680x0 code needs to be executed. (For an excruciatingly detailed—but useful—explanation of mixed-mode mechanics, see "Understanding the Power Macintosh Architecture, Part Two: A Skeleton Key to Mixed-Mode Issues," in the May 1994 issue of *Apple Directions*.)

On the 680x0 side, the code being executed is always 680x0 code, but two modes are mixed, just as on Power Macintosh computers: code that assumes the old run-time architecture, and code that assumes the new run-time architecture. For this reason, you should read the Mixed Mode Manager chapter of *Inside Macintosh: PowerPC System Software* to learn how to use MMM-68K.

Human Interface

Doc's Holiday Wish List

By Peter Bickford

Here in beautiful Silicon Valley there's a business called Fry's Electronics. This is geek Mecca, with the Sunnyvale store designed—I'm not making this up—like a gigantic computer chip. Walking through the doors (marked "Enter" and "Escape" with giant key caps), you enter a vast showplace of discount electronics, computers, CDs, junk food, and the odd hair-care product. In short, it's everything you need to live the propeller-headed equivalent of the Good Life.

A few days ago, as I was making one of my frequent trips to Fry's, I noticed the unmistakable signs that the seasons were changing. That is to say that the bags of fun-size Snickers bars they'd stocked for Halloween were now on sale for 99¢, sitting next to the inevitable bins full of holiday clip-art software. Although the air was only a touch cooler than normal, there was no denying that California's version of Christmas was right around the corner.

So once again, it's time for my human interface holiday wish list. This is essentially a big shopping list of things I'd love to see developers do with their interfaces over the next year. Unlike last year's list, which was pretty cleanly divided into little "stocking stuffer" wishes and more general "big ticket" items, this list is all over the board. I'd love to see developers do everything on this list, but getting any one of them done would put a smile on this human interface Grinch's face this holiday season.

#1—Clarify Clicking

Right now, using some programs is like playing the children's game Manhole—you simply click on every picture or icon trying to find out if it will do something interesting. If you want users to know they can click on a button or picture, at the minimum give it a solid black border. You may also want to use 3-D effects to raise it from its background and make it look more clickable.

The next step is to let users know what will happen when they click something. The obvious solution is to add labels to icon buttons, but this solution has been dismissed by many toolbar makers as overly radical. They would prefer users to discover through trial and error that the icon showing two hands cradling a piece of paper means "save as" or that the whirlpool icon stands for "transmogrify." Actually, in that last case, I'm not sure a label could have helped. . . .

As a final point, make it clear what double-clicking will do—and always make sure there's a way to do the same thing without double-clicking (many folks are physically incapable of making double-clicks). We admit it's not always easy to do this, and we confess that System 7.5's new Find File process, neat as it is, can confuse users when it comes to double-clicking. Once you've searched for an item, it shows you a window with two panes. Double-clicking an item in the top pane launches that item. Double-clicking an item in the bottom pane opens the item's enclosing folder. Adding action buttons to the window would have gone a long way toward clearing up the confusion.

#2—Don't Go Crazy With Color

I've seen programs, desperate to prove how New, Exciting, and Colorful they are, try to dazzle their users by setting all text in red or blue against a bright yellow background. This is great for attracting attention, but it's murder if you actually want to read the text. Just as in books, the best combination for readability is usually black against a white background (or vice versa). Aim for maximum contrast, and use colors only for call-outs or headlines.

When designing in color, it's also important to keep in mind some of the technical issues. Icons should be designed using the "magic 34" icon colors shown in ResEdit. These help harmonize the overall look of icons in the interface, and the colors are handled specially so that they're the last ones swapped out if the palette needs to change. Also, use the regular system color palette for your interface unless you have a very good reason for using a custom one. One database program, which shall remain nameless, decided it needed a completely different color palette to express the sublime reds in its logo. As a result, it not only scrambles the screen colors of any other program you're running when you launch it, but the applications you create using this database do the same.

#3—Lighten Up Those Grays

The dark gray, funereal interfaces have got to go. For those who are lucky enough to know what I'm talking about, I'm referring to the gray chiseled interfaces with the gratuitous 3-D effects that have become a recent interface fad. These essentially are a corruption of the original NeXT interface, except that there, the 3-D effects were used to help you know what could be clicked on. Moreover, the whole color scheme was implemented that way because there were only a few levels of gray available, and it matched the look of the black "NeXT Cube" workstation itself.

If you're sick of your product's current black-and-white interface and are contemplating doing one of these NeAT-O gray things, my advice is this: Design it, get it out of your system, then rip it up and design your product's real interface.

Now, not all 3-D, grayscale interfaces are bad. The problem comes when you throw in lots of visual effects for flash and end up obscuring the interface as a result. If you want to design a 3-D interface, take a look at "Working in the Third Dimension," an article in *develop* Issue 15 (September 1993), for one example of how to do it right. That article features an interface in which the grays are very light—enough for contrast with white, but not so dark that black text is difficult to read. Also, the 3-D effects make it *easier* to know what controls are active and which icons are buttons you can push. Essentially, these buttons seem to rise up from the background,

showing that they're active and pushable. Disabled buttons don't have the beveled edges that make them look pushable.

#4—Leave the System Menus Alone
The newest interface fad seems to be for every utility application to want to install its own system-level menu, often under some sort of icon heading. This is one of those things that's kind of neat when you're the only person doing it, but it gets downright ridiculous when everyone else does the same thing. Of the software I use regularly, DiskDoubler, QuickMail, TouchBase, CPU, Datebook, Timbuktu, and Now Utilities all attempt to add their own menus to the menu bar. Not only is this a notoriously unstable thing to do from a system level, but at this point, there's simply no space for it.

Similarly, don't install an alias to your program in the Apple menu unless there's an extraordinarily good reason to do so. The Apple menu is users' space—it's there for the things they find most useful in working with their computer. Depending on the size of their screen, users tend to be more or less generous in granting applications one of the coveted positions in their Apple menu. Don't assume your application automatically qualifies for this honor.

#5—Make Software More Self-Configuring
One of the biggest advantages of the Macintosh is its ability to sense its own hardware configuration and act accordingly. We need to use this ability, along with some intelligence in our applications, to take the burden of setup off the shoulders of our users.

Network software, for example, should follow the lead of America Online and others in trying to aggressively determine the modem port, speed, and type when it first connects. America Online even goes so far as to dial an 800 number and help you locate the appropriate local phone numbers to call when using the system. As a rule, don't ask the user for things you can have the system find out for itself.

#6—Develop Fax Software That Works
There's no gentle way to put this. Most of the current fax software is really bad. One of the most popular packages requires three extensions, a phone book editor, an OCR application, a fax monitor, a control panel, and the ever-present Fax menu in order to function. Even then, it barely does the job.

What we need is something like a desktop fax in the same way that QuickDraw GX has desktop printers. To fax documents, you would simply drag them over to the fax machine icon. A dialog box would then appear asking you who you wanted to send it to, the resolution settings, and so on. Of course, it would remember your previous settings and offer those as defaults. In addition, it would be able to pull information from your favorite personal information manager

application, so you wouldn't need to maintain multiple address books.

As the document was being sent, the software could show the pages scrolling through the fax icon with a running page count. The user could also double-click on the fax icon to see more detailed information, cancel the transmission, and so on.

Finally, the ideal fax software would handle the communication port arbitration problems so that your desktop fax machine isn't cursed by the same quirk that makes timed VCR recording a pain. Most VCRs today are really quite simple when it comes to programming the recording times. The trick is that for the program to record, you need to put most VCRs in "timer" mode, which prevents you from using the VCR in the normal fashion to watch television. As a result, you leave timer mode turned off and you end up forgetting to tape *The Simpsons* when you work late. (I don't mean to sound bitter. . . .) Fax software has a similar problem in that it must be set to "receive" mode to receive fax documents. Of course, that usually stops you from using your modem, an activity that is generally more common than receiving faxes. As a result, you leave receive mode turned off, and people wonder why your fax modem never picks up.

I swear, the people who develop the fax software that solves this problem (or the VCR chip set that fixes the timer mode problem) are going to make a bundle.

That's it for this year. As a final wish: I hope you all have a happy and prosperous New Year, and may we all create products that make the lives of our customers a little bit brighter.

Happy Holidays,
Doc
AppleLink: THE.DOKTOR

AppleDirections

Backward Compatibility With CFM-68K

Toolbox routines use the old run-time architecture; CFM-68K assumes the new run-time architecture. How do you reconcile them in your source code without having to rewrite every Toolbox call you make? If your source code is written in C or C++, the process is largely transparent to you. Here's how it works. (See the figure "Making Toolbox code run transparently.")

Under the old run-time architecture, a Toolbox call (upper left) is compiled into an A-line 680x0 instruction (denoted by one small rectangle inside the Application box). When executed, that instruction triggers an exception routine that uses information from the instruction to look up the address of the correct routine and jump to it (lower left).
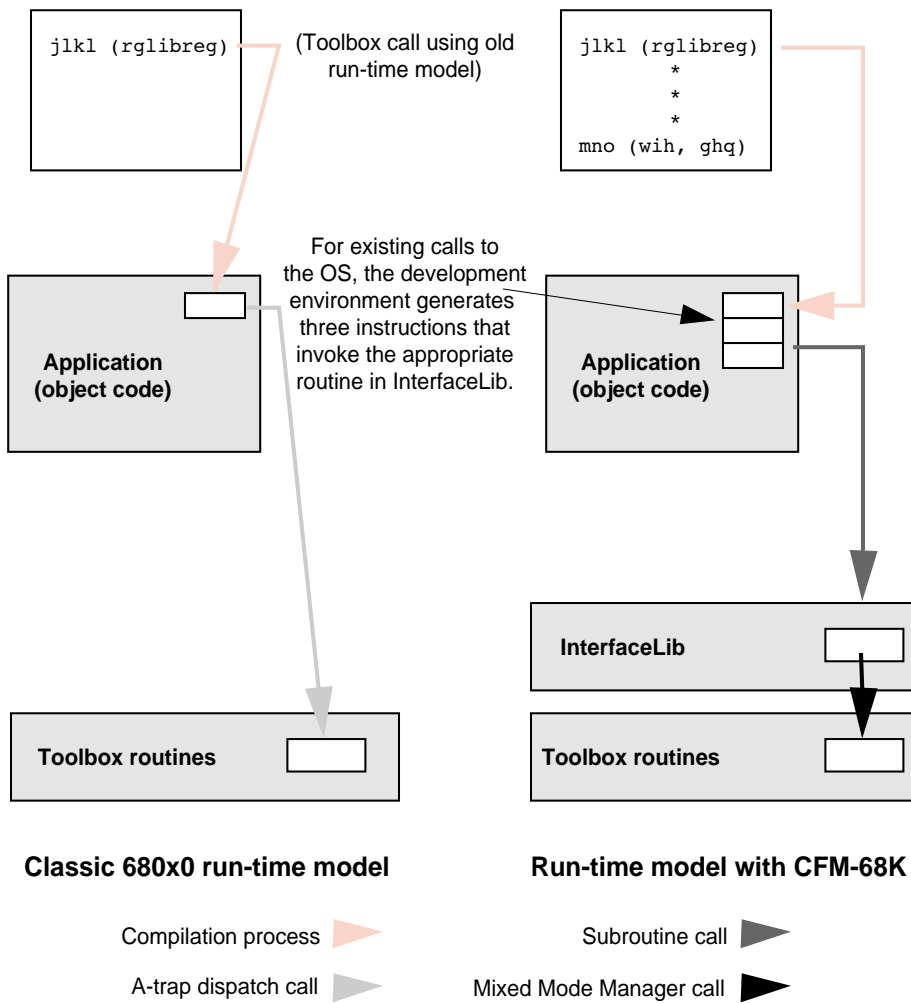
The problem that Apple engineers had to solve is this: CFM-68K uses a different calling mechanism to call *all* routines. How, then, can it call existing Toolbox routines, which use a different calling mechanism (that is, trap dispatching)? The solution to this problem has two parts.

First, you write a shared library of routines that "translate" new run-time calls into the form expected by the old run-time architecture (a process called *marshaling*), call the appropriate old run-time Toolbox routine, and marshal the results to conform to the new run-time architecture. The shared library that does this for Toolbox routines is called InterfaceLib. (The lower-right quadrant of the figure shows this. Note that the call made by the InterfaceLib routine is a mixed-mode call.)

The second part of the solution is a change to the universal header files so that they always "do the right thing." These files have been designed so that when you compile for the old 680x0 run-time architecture, Toolbox calls get compiled into A-line traps, but when you compile for the new run-time architecture (for either the 680x0 or PowerPC processor), you get external function calls. On the 680x0 side, a Toolbox call is replaced by *three* instructions that, when executed, cause the appropriate InterfaceLib routine to execute. (This InterfaceLib routine, as described above, then takes care of executing the desired Toolbox routine.)

In summary, development environments meant to work with CFM-68K make it possible for you to compile your program without rewriting every Toolbox call (which is what you have to do if your source code is in a language not supported by CFM-68K). Currently, only the newest MPW C and C++ compilers (SC and SCpp) support CFM-68K. However, Apple expects more development environments to support CFM-68K in the future.

```
jlkl (rglibreg)
```
(Toolbox call using old run-time model)

```
jlkl (rglibreg)
       *
       *
       *
mno (wih, ghq)
```

For existing calls to the OS, the development environment generates three instructions that invoke the appropriate routine in InterfaceLib.

Application (object code)

Application (object code)

InterfaceLib

Toolbox routines

Toolbox routines

Classic 680x0 run-time model

Run-time model with CFM-68K

Compilation process

A-trap dispatch call

Subroutine call

Mixed Mode Manager call

Making toolbox code run transparently. This shows how the same Toolbox call is executed under both the old (left) and the new (right) run-time architecture.

The New Macintosh Run-Time Environment

Using Import Libraries on 680x0

To use import libraries with your 680x0 programs, you'll have to make some changes. First, you'll need the latest version of MPW—in particular, the SC (for C) or SCpp (for C++) compiler and the ILink linker. Pre-release versions of these compilers are on E.T.O. 16, the latest CD of Macintosh development tools. (E.T.O. 16 should be available to E.T.O. subscribers by the time you read this.) These tools will also be available on the next release of MPW Pro, due in early 1995.

If your code is written in C or C++ and you've converted it to compile for Power Macintosh computers, you'll have little or no extra work to do. You'll need to rewrite your "make" file to compile and link your program using the new tools.

The new compiler and linker will actually rewrite your program to work with the new run-time architecture, even though your code calls existing OS and Toolbox routines, which are written following the old run-time architecture. (See the text box "Backward Compatibility With CFM-68K" for details.)

If your C or C++ program contains inline assembly-language code, you'll have to rewrite every assembly-language Toolbox call to call instead the appropriate routine in an import library called InterfaceLib; the InterfaceLib routine is "glue" code that handles the details of calling the Toolbox call. Optional system extensions—QuickTime and the Drag Manager, for example—currently have their own glue code; under CFM-68K, you will similarly have to rewrite assembly-language calls to

routines belonging to these extensions (making calls to their corresponding CFM-68K "glue" shared libraries). Depending on the release you're using (after all, the version of CFM-68K on E.T.O. 16 is a beta version), you may have to make other changes to adapt assembly-language source code to CFM-68K.

If your code is still 680x0-only and written in C or C++, you'll have about as much work ahead of you as you would in converting your program to run on Power Macintosh computers. (This is because Power Macintosh computers use the new run-time architecture and therefore necessitate the same kinds of changes as CFM-68K does.)

If your code is written in some language other than C or C++, you're out of luck for the moment. Language Systems says it currently plans to have CFM-68K support for its compilers (which includes its Object Pascal compiler) sometime in late 1995. Customer demand, however, is for C and C++ compilers, especially for the Power Macintosh, and I personally would not recommend that you wait for non-C/C++ support for CFM-68K.

If you have routines written in 680x0 assembly language and they contain OS, Toolbox, or system-extension calls, you'll have to rewrite them as described earlier.

What Is SOM?

The System Object Model (commonly abbreviated as *SOM*) is the third piece of the new run-time architecture. It is an object-oriented programming technology from IBM for building, packaging, and manipulating class libraries in binary (object-code) form. Apple has licensed SOM from IBM and has implemented it on the Macintosh platform as a "fat" shared library that works with both 680x0 and Power Macintosh computers.

About the only thing you *must* know about SOM at this point is

that it's built on top of the Code Fragment Manager—so if you want to use SOM (or OpenDoc, which uses SOM) on 680x0 Macintosh computers, you'll have to make the transition to CFM-68K.

Advantages of SOM

In addition to its contribution to the OpenDoc component software architecture, SOM's *raison d'être* is to make object-oriented software more useful in the real world. SOM does this in two ways: first, by enabling class libraries from different sources to work together; and second, by enabling class libraries to be revised and to continue to work without requiring you to recompile the programs that use them. It's also important to note that SOM is implemented in a general way, so that it can provide the

same object-oriented services for different platforms and programming languages that vary vastly in their overall structure.

A *class library* is a type of shared library that implements a class of software objects and the methods associated with it. Client programs use these class libraries to get their work done by creating and manipulating objects. Class libraries are useful because, once written, many programs can use them. Because one class can inherit behavior from another, programmers can often be more productive deriving new classes from old ones than they can from modifying procedural code.

One problem hampering the widespread use of class libraries is that those created in one programming environment are almost never usable in another.

New Run-Time Architecture— What Changes?

The new run-time architecture constitutes a fundamentally different way for one routine to call another, one that is simpler, faster, and more powerful. The two major differences are as follows:

• *Different calling conventions.* For historical reasons, the old run-time architecture uses several calling conventions. Routines written in C and Pascal have their parameters passed on the stack, but in a different order and with other variations. Older Toolbox routines use Pascal conventions, while newer ones use C conventions. In addition, operating system calls generally pass parameters and results in processor registers and do not use the stack for parameter passing.

The new run-time architecture simplifies things considerably: all routines are called in the same way. For more information, see pages 1-42 through 1-50 of *Inside Macintosh: PowerPC System Software.*

• *Brave new A5 world.* In the new run-time architecture, every code fragment has its own A5 world (an area of memory traditionally pointed to by the A5 register of the 680x0 processor). Since, in the old run-time architecture, only applications have their own A5 world, situations exist where the A5 register must be saved before and restored after another routine can be safely run.

On both the 680x0 and PowerPC platforms, the Mixed Mode Manager does the necessary translations to allow both old and new run-time architecture routines to run correctly. (Note: The model described here will not be fully implemented until April 1995, when E.T.O. 17 is available.)

For example, a class library written in Object Pascal has a calling convention that is different from what a C++ program expects to see. Even worse, most C++ compilers create different kinds of class libraries. The effective result is that, to use an existing class library, you must use the same programming environment that was used to create it. SOM eliminates that problem, allowing any supported programming language—even C, which is not object-oriented—to use any SOM class library.

If that first problem weren't enough to limit the usefulness of class libraries, there's something worse, called the *fragile base class problem.* In some object-oriented languages (most notably, C++), there are implementation dependencies between the client program and the class library it uses. The problem is that even the slightest change to the class library results in a binary file that won't work with the existing client program—the two will work together only if the client program is recompiled along with the class library. This makes it impractical to use class libraries in most software, especially commercial software—nobody wants to recompile and redistribute all the client programs every time one of its class libraries is recompiled.

SOM fixes the fragile base class problem. Obviously, if you change the number or type of arguments to a method, you're going to have to change and recompile the client program. But if you're just changing the method's algorithm (fixing a bug, for example) or making some other change that doesn't force you to change the client program's source code (adding a new method, for example), SOM can take care of the situation. You simply recompile the shared library and replace the existing binary shared-library file with the one you've just created.

Most developers won't deal directly with SOM. Although it will be present on their customers' computers, so that OpenDoc will work, developers won't have to do anything differently. However, you should consider the advantages of using SOM class libraries in your products; you might even find a commercial opportunity in creating SOM class libraries to sell to other developers.

How Does SOM Work?

In the future, SOM may be integrated into compilers, thus making its use more transparent. For the moment, here's how you interact with SOM.

If you're building a class library, you proceed as follows:

• You first describe the interface for a class using a language called the *Interface Definition Language,* or *IDL* (which is very similar to a subset of the C++ class-definition syntax). The result is an .idl file.

• You then use the .idl file and an IDL compiler to create a set of header files that are later used in the compilation process.

• One of the files created is a source file that contains "empty" functions that implement the class and its methods. You fill out the bodies of these functions, thus creating the code that implements the class.

• You compile all these files together; this results in a CFM shared-library binary file that implements this class. You then give the binary file and the .idl file to whoever wants to use the class library.

If you're writing a program that uses a class library, follow these steps:

• You take the class library's .idl file and use an IDL compiler to create a header file. You need to compile this header file with your program to enable it to access the class defined by the binary file.

• The binary file implementing the class must be present when your program runs.

Pointing Toward the Future

The addition of CFM-68K, MMM-68K, and SOM to the 680x0 Macintosh world (and SOM to the Power Macintosh world) is another step in the evolution of the Mac OS. It simplifies the run-time architecture and helps pave the way for OpenDoc and future improvements to Mac OS hardware and software.

You don't have to make the change today—today's 680x0 applications, which use the old run-time architecture, will continue to run on 680x0 Macintosh computers for some time to come. However, switching to the new run-time architecture will keep your source-code base current and allow you to take advantage of shared libraries and SOM class libraries. You need to decide when to make this important change. ♣

OpenDoc Human Interface Q&A

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

We frequently receive questions about OpenDoc that need to be answered in a public forum to help our partners' OpenDoc development efforts. These questions come from our colleagues and engineers in the human interface community through conversation and e-mail, and they address OpenDoc topics we think more of you want to know about.

Q: What has changed from today's application-centered world?
A: A fundamental change from today's application-centered world is that applications will be replaced with part editors and part viewers. These are installed in a special folder, the Editors folder, as described in the OpenDoc human interface guidelines.

In OpenDoc the primary focus is on the user's tasks and data instead of on the part editors being used. So the part editors remain hidden until the user needs them to manipulate some content. Today, when a

user opens a document, its application generally starts automatically. In OpenDoc, part editors start in a similar way. But OpenDoc automatically stops part editors when they are no longer needed—that is, when there are no longer any open documents that use them. This means that the user never needs to quit, and a part editor never leaves its menu bar displayed after its last window is closed. We think this minimizes user confusion.

It's important to note, however, that because part editors are started and stopped at random times, your part editor will cause distraction if it displays a "splash screen" every time it is launched. For details, check the *OpenDoc Human Interface Guidelines for the Macintosh* on the OpenDoc with SOM Developer Release CD (sent to all Apple Associates and Partners along with this issue of *Apple Directions.*)

Q: Is everything a document? What isn't?
A: In Gregg Williams's Strategy Mosaic column, "OpenDoc *Is* Cross-Platform" (*Apple Directions,* November 1994), we imagined a future in which all software has been reengineered as OpenDoc parts. Gregg asked, "Document? What document?" Evidently, there is some real confusion over what a document is, and more fundamentally, why this new technology is called *OpenDoc.* So what's in a name?

We currently think in terms of a document as the product of an application—or the realization of a collection of ideas. We've taken that concept a step further in OpenDoc and conceptualized the document as a collection, not of static representations of ideas, but of active, live parts. Consequently, an OpenDoc document is a bounding mechanism for just about anything you'd like to collect together. In short, a

document is a container that is open to any kind of data or active behavior you want to include.

Q: What is an active part?
A: This topic is often confusing the first time it's explained. Fortunately, our user studies show that determining which part is active is transparent to users—they don't have problems using this element of OpenDoc. (Hey, maybe it's just that we didn't try to explain it to them.) But it is important for developers to understand the conceptual model we are presenting to our users.

The active part is the part that is currently receiving input events; it identifies itself by sporting the active border. As a user "mouses around" and changes the selection or adds content, the active border appears wherever there is a selection or insertion point in a part. As in today's Macintosh environment, there is only one selection in a window, so only one part is active at a time. When a part is active, its menus are displayed in the menu bar, and it receives the keyboard and other input device events. (Exception: When a dialog box is displayed by the active part, it normally handles these events.)

You can think of the active part as being like the active window, which indicates where the user is currently working. Today's active window uses distinct visual feedback; similarly, the OpenDoc active part uses its own visual feedback—the active border. This border is the boundary of the part, and it tells users about the part in which they're working: how big an area they have to work in, and how far they can drag and drop content before going outside this part.

Let's assume that we're writing this article using an OpenDoc text editor, instead of a conventional word processor. We're working inside the text part and the text part is active. While

using the text editor, we see the text editor's menus. Now suppose we paste a picture into this text; that would be a different part, but it would be content within the text. To edit the picture, we just click inside the picture, and, because it now receives the input focus, it becomes the active part, and the active border is drawn around the picture instead of the text. Now the keyboard and other input device events are being handled by the picture's part editor.

Q: Why not just click anywhere on a part to select it?
A: In an "inside-out" model, the first click performs an action on the deepest level. The alternative "outside-in" model requires users to click once to activate the content, and then again to set an insertion point or make a selection. We know from research going back to the Xerox Star in the early 1980s (and confirmed by our recent user studies) that users strongly prefer to click once in some content to start editing it. We also know that users edit the contents of a part more frequently than they manipulate an entire part. Although these results are task-dependent and there isn't a large percentage of difference in preferences, we believe we've learned that the "inside-out" model of single-clicking to start editing improves the usability of the system.

Let's investigate the vagaries of single-clicking and the reasons why OpenDoc uses it according to the "inside-out" model just described. Imagine a different model in which the user single-clicks to select the part. Once the part has been selected with a single click, this model requires a different mechanism for editing the contents of the part. But a part may contain more content nested within itself, and often the nesting is invisible. Consequently,

a model that uses a single click for part selection demands that the user navigate an often invisible hierarchy. The navigation mechanism typically used is the double click, because it means "open." But it also means "select a word or some larger content." Thus, when users try to insert a word into a label within a chart, they might be forced to double-click several times to tunnel down through the hierarchy of nested parts. And then at the end, if they misjudge the number of levels, they are likely to inadvertently select a word instead of placing an insertion point. With the OpenDoc model, a single click places the insertion point where the user wants it.

Q: How do users make the transition from applications to parts? How transparent is this transition? How can users tell if something is "OpenDoc-enabled"?
A: From our user studies we kept hearing things like "You fixed some of the bugs" and "That's the way it's supposed to work." We're very happy, because we know it is vital that users feel that the system has improved, rather than having simply changed. While we anticipate that users will notice that working with parts is different from working with today's applications, we've implemented sufficient visual feedback to let users know they're working within OpenDoc. What, exactly, looks different about OpenDoc?

Working in OpenDoc doesn't necessarily look different from working in today's environment. While users may use OpenDoc to create the same traditional types of documents that they create today, they may also create component documents that look and behave differently. It is the change in how one interacts with a document that is significant, and we've developed feedback

such as active and selected part borders to help users shift focus easily among various parts in a component document.

Additionally, OpenDoc has distinctive icons identified with the OpenDoc logo. Since stationery and documents behave pretty much the same way they always have, their icons should follow current icon development guidelines. However, editor and viewer icons do not behave exactly like today's application icons—for example, they won't perform correctly if they're not properly stored, and double-clicking them doesn't produce the same results as double-clicking an application icon. Therefore, we've developed a distinctive shape for OpenDoc editor and viewer icons to help users notice the difference.

To smooth the transition from applications to OpenDoc parts, we have introduced guidelines on how to help your application fit in better in the *OpenDoc Human Interface Guidelines for the Macintosh* on the OpenDoc with SOM Developer Release CD. For example, supporting drag-and-drop capability is highly encouraged.

Q: What is the role of stationery in OpenDoc? How should developers use stationery?

A: OpenDoc uses stationery as a starting point for users to get a new document or new kind of content into their document. Many applications today ship stationery pads, and OpenDoc continues that practice.

Most users will start off using stationery supplied by part developers or by vertical market developers, so shipping some samples is a good idea. You should ship at least one stationery pad that can be used to create a new document containing one instance of your part. As users become more sophisticated in their use of

stationery, they will develop their own novel forms and uses.

For example, if you ship a drawing editor, your stationery pad should create an empty drawing document. If your editor does network database queries, your stationery pad should create a document that contains a blank query form. Of course, we encourage you to ship samples as stationery, too. For an editor that does network database queries, you might provide two stationery pads—one with a simple query and another with a more complex query.

Because stationery should be installed in a location that users can find easily, we are introducing the Stationery Folder. Check the *OpenDoc Human Interface Guidelines for the Macintosh* for details.

Q: What happens when I receive a document that uses an editor I don't have?

A: This is probably the question we're asked most frequently. The short answer is that while there is no magic that can make the "missing editor" problem disappear, there are at least two workable solutions: viewers and translators. If users don't have the appropriate part editor (or application), they pray that they have a useful viewer so that they can at least view, if not manipulate, the document just received. Viewers currently exist, and we suspect that needing to use one in the OpenDoc world should occur about as often as it does now. We encourage developers to provide translation as another option for working with data for which users have no part editor. However, translation offers varying degrees of fidelity to the original.

The longer answer is that in today's offices and homes, users who frequently exchange data tend to take advantage of the

same set of applications. This occurs because the use of common applications is mandated by management, or is tacitly agreed upon among users. Users also tend to have a few applications outside the set they commonly use so that they can read documents sent from outside their immediate group.

All of these same solutions and more are used in OpenDoc. First, a part may store its data in several formats so that quality translation is more probable, if it should be required. Second, Apple strongly encourages you to create a viewer for your part that can be distributed freely. That way, OpenDoc will provide users with a better solution than today's applications, because users will always be able to view a document, even when they don't have the appropriate part editor to manipulate it.

Q: What is linking? For what is it intended?

A: Linking is intended to allow users to replicate data dynamically, much as the publish and subscribe feature does in System 7.0. We know that the publish and subscribe feature is not used by a large number of users. In part, this is because many users don't need this functionality; it's also because publish and subscribe is difficult to use. While OpenDoc's linking has a much-improved interface, not every user will take advantage of linking.

Linking allows users to copy and paste data as a link from one location to another, keeping the copy updated as the original changes. The destination part accepts the link content as either merged or embedded content. Links are one-way; that is, the original source content is duplicated at the link destination. If the link destinations are within the same document as the source, changes made at the

source are automatically and immediately reflected in the copied content at the link destination. Links that cross document boundaries are automatically updated when the document is saved, or the user can manually specify when to update.

Linking is an efficient way to keep copies of changing data in synchronization when it is copied to many locations inside or outside the original data's location. However, linking is also a way to use the same data more than once, and display it differently (this is in addition to OpenDoc's ability to display a part in several alternate "views"). Linking allows multiple copies of the part to stay synchronized. Thus, for example, a spreadsheet can be duplicated so that the original (or source) data is displayed as a spreadsheet, and the other copy (the destination) is displayed as a bar chart; the bar chart changes its display whenever the spreadsheet changes.

We use special link borders to indicate to the user that some content is a link. Consult the *OpenDoc Human Interface Guidelines for the Macintosh* for more details on the behavior and the display guidelines for linking. ♣

Editor's note: The OpenDoc Human Interface Team will be back next month with more Q&As.

Business & Marketing

Apple Initiatives Boost Third-Party Software Sales

In a drive to make the holiday season merry for third-party Apple software developers, Apple Computer, Inc., launched a series of U.S.-based initiatives aimed at helping Macintosh software gain retail shelf space and sales momentum. Apple is sponsoring no fewer than five retail promotions and four CD-ROM–based catalogs of Apple-related products, providing many developers with product publicity and incremental seasonal sales.

"All of these initiatives are focused on letting our mutual customers know that there is an abundance of great Macintosh titles and on convincing software distributors and retailers that these titles are money makers, all of which are vital in keeping Apple's platform strong and competitive," says Steve Franzese, Director–Business Development, Apple's New Media Division.

Many of these efforts will continue beyond the holiday season, as part of a CD initiative that originated from Apple President Michael Spindler's office. For developers interested in participating in these future efforts, this article provides details on specific initiatives and information on the kinds of third-party products Apple program managers are looking for in their continuing programs.

Market Research Monthly

Exclusive Data From Apple's Macintosh User Profile Study

Apple Computer, Inc., has just completed its annual U.S. Macintosh User Profile Study, which for the first time includes comparative data about the DOS/Windows market. This month's Market Research Monthly gives you an exclusive look at the study's key findings. Highlights of the proprietary Apple study include the following:

• CD-ROM penetration has more than doubled in the Macintosh market and for the first time has reached parity with the Windows market.

• Modem ownership has also grown significantly, but modems still remain highly underutilized.

• Although the Internet has aroused much interest, it remains a mystery to most customers.

• Eighty-five percent of U.S. Macintosh users are now running a version of System 7.

• The average amount of memory installed in Macintosh systems has doubled in one year, from 4 MB last year to 8 MB this year. By comparison, the average DOS machine is installed with 2.5 MB; the average Windows system, 5.5 MB.

Here are the details behind these key conclusions. (Note that the data was collected too early in 1994 for Power Macintosh sales to have had significant impact on the installed base.)

CD-ROM Growth

The CD-ROM connection rate has more than doubled in the Macintosh installed base, increasing from 8 percent in 1993 to 19 percent in 1994, the same rate as in the Windows market. The CD-ROM connection rate should continue to surge upward: Apple's monthly Macintosh Recent Buyer study shows that approximately 50 percent of Macintosh systems selling today are CD-ROM–equipped. If you're a multimedia title developer, this means the number of potential customers for your products has more than doubled in the past year, presenting you with significant revenue opportunities. Others take note: You can now reach an increasing portion of the Macintosh installed base with CD-ROM–based versions of your products.

Modem Growth

The modem connection rate has also shown extraordinary growth over the past year; currently, 42 percent of U.S. Macintosh systems include a modem, compared to 28 percent in 1993. This increase is largely attributable to the bundling of modems with Macintosh Performa and PowerBook computers and the availability of cheaper add-on modems.

Although nearly half of the Macintosh installed base now has the capability to access the information superhighway, Apple customer research has shown that many of these customers don't know how to reach the onramp. Many users who have a modem either haven't gotten around to using it or simply don't know what to do with it. Only half of all modem owners—in both the Macintosh and Windows markets—access online services and use e-mail. Most of the remaining modem owners use their modems infrequently or not at all. This presents a clear opportunity for you to create products that make it easier for customers to use modems, whether you provide front-end software, instructional material, or some other product.

AppleDirections

The Internet

The Internet is attracting a vast amount of interest, but it's creating confusion, as well. Only about 6 percent of Macintosh and Windows customers have used the Internet. Most users don't even know how to connect, let alone know what to do once they get there. Many of the customers who have used the Internet complain that it's a nightmare to navigate. To repeat something we've said before, this situation gives you the opportunity to develop software providing customers with easier Internet access, simpler interfaces, and more productive navigational methods.

System Software Usage

Overall, 85 percent of U.S. Macintosh customers are running System 7. Nearly all the owners—more than 90 percent—of Macintosh LC and Macintosh II models are running System 7. However, a substantial proportion of low-end customers continue to use System 6. Nearly one-half of Macintosh Plus, SE, and SE/30 customers are running System 6, and over half of Macintosh Classic® users run System 6. For details, see "Who's Using Which Version of the Mac OS, 1994" on this page; keep in mind that these numbers don't include any figures for usage levels of System 7.1.2, the first Power PC processor–based version of the Mac OS, or the recently released System 7.5.

Memory and Hard-Disk Drives

With Macintosh computers having shipped with larger amounts of memory and more hard drive capacity for some time, the average Macintosh system is now equipped with 8 MB of memory and an 80 MB hard disk drive, compared to the 4/40 system of just a year ago. "Memory and Hard-Disk Drives in the U.S. Macintosh Installed Base, 1994," provides details.

Configurations of Macintosh models follow a somewhat natural progression up the product line, starting as low as an average configuration of 2/40 for Macintosh Plus and Classic systems, 5/80 for Macintosh LC systems, 8/105 for Macintosh II computers, and 20/500 for high-end Macintosh Quadra computers. Macintosh Quadra 900/950 computers boast the largest configurations, with a 40/620 average.

By comparison, DOS users have an average of 2.5 MB of memory; Windows users, 5.5 MB. Nearly half of the Windows customer base has 4 MB of memory, making it the most common Windows memory configuration.

The information for the study was gathered during May and June 1994 using a disk-by-mail survey sent to 1,038 registered Macintosh users and a random sample of 174 DOS users and 383 Windows users. The sample consists of users who purchased their computers from 1988 through early 1994, representing a total of 7.12 million Macintosh computers, 15.3 million DOS PCs, and 21.1 million Windows PCs.

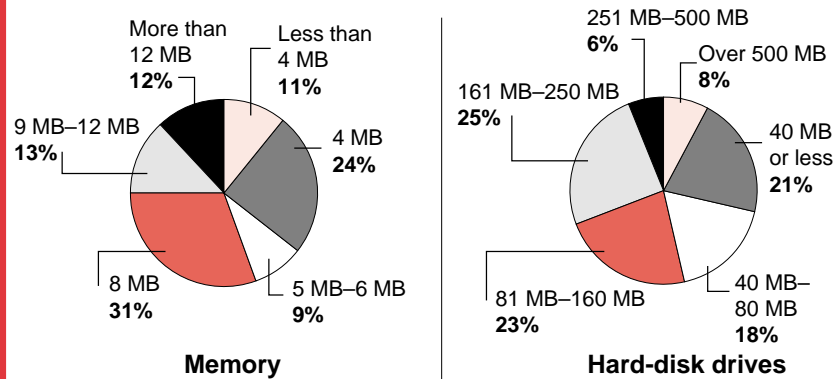We'll provide additional details from the study in coming months. ♣

Who's Using Which Version of the Mac OS, 1994

- Other System 7 **4%**
- System 6.x **15%**
- System 7.0 **7%**
- System 7.0.1 **21%**
- System 7.1/7.1p **53%**
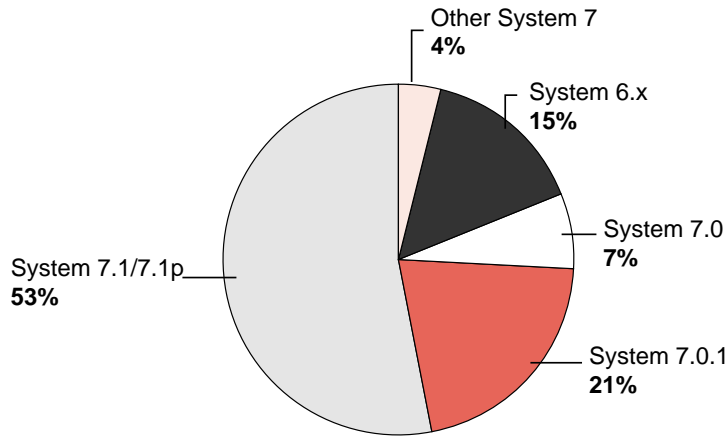
Source: Apple Computer, Inc. © 1994

This pie chart shows the percentages of the overall U.S. Macintosh installed base using different versions of the Mac OS. It does not include any data for Power Macintosh users, since the data was collected too late for Power Macintosh sales to have affected installed base data. Note: System 7.1p is the version that shipped with some Macintosh Performa systems.

Memory and Hard-Disk Drives in the U.S. Macintosh Installed Base, 1994

- More than 12 MB **12%**
- Less than 4 MB **11%**
- 4 MB **24%**
- 9 MB–12 MB **13%**
- 5 MB–6 MB **9%**
- 8 MB **31%**

Memory

- 251 MB–500 MB **6%**
- Over 500 MB **8%**
- 161 MB–250 MB **25%**
- 40 MB or less **21%**
- 81 MB–160 MB **23%**
- 40 MB–80 MB **18%**

Hard-disk drives

Source: Apple Computer, Inc. © 1994

These two pie charts show the percentage breakdown of the U.S. Macintosh installed base by amounts of memory and hard-disk storage. Note that more than half the installed base has at least 8 MB of memory and over 35 percent has more than 160 MB of storage capacity.

Beyond Software Stores

An Overview of Emerging Retail Channels

By Vicki Vance,
Apple Computer

It should come as no surprise to you that it's becoming difficult to get products on the shelves of software stores if your products don't fit a retailer's ideal profile: a broad multiplatform product line, a suggested retail price of $50 or less and a track record that shows that you'll sell more than 50,000 copies of each game titles or 30,000 copies of each edutainment titles per year.

Fortunately, as more low-cost computers with CD-ROM drives end up in the hands of mainstream consumers, you can find increasingly attractive alternatives to selling through software stores. Bookstores, entertainment stores, "try-and-buy" CD-ROM discs, and online services are offering many developers a better way to reach this new breed of customer.

Having worked for Apple in the area of title distribution and bundling for the last few years, I can give you an update and a per-spective on these emerging channels. In this article, I explain why it's important to start gaining a presence in these outlets today and share advice from vertical channel experts on the types of software products that they think will sell.

On the Importance of Alternative Channels

Today traditional software distribution channels have reached a choke point, and the industry is poised for change. Your job as a self-publishing developer or software publisher is to sort through the proliferation of retail experiments underway, and select the ones that will work best for your products.

Software retailers feel that shrinking profit margins, a proliferation of titles, and bulky packaging have limited their ability to offer consumers a wide selection of titles. And this shelf-space problem will only get worse when entrepreneurs, publishers, and media moguls get their title-creation engines up and running.

How will these titles reach their intended audience, and how will you be able to compete? Market experts believe that a presence in one or more of the emerging software channels discussed in this article is essential to long-term developer success.

Joey Tamer, a veteran CD-ROM distribution consultant and president of SOS, Inc., heartily agrees with this premise: "Publishers of CD-ROM titles cannot afford to market solely through software-only channels. By 1997, the dominant channels for CD-ROM titles will be superstores of all types—audio, video, computer, and bookstores. Smaller software stores, which have fewer locations and carry fewer titles, will become only one choice among many channels."
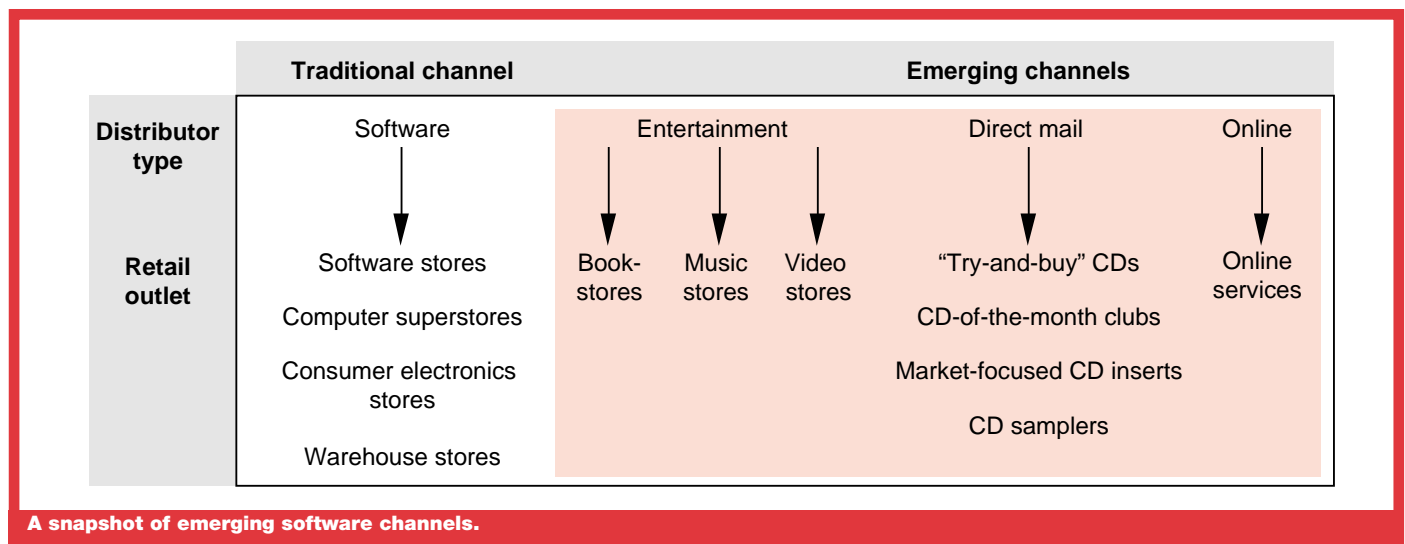
Tamer thinks it's especially important for Macintosh developers to take this advice. "Many software retailers are resistant to Macintosh titles because of the historical tendency for Macintosh users to bypass retail and buy through mail order," says Tamer. "Because of this, Macintosh developers, more than Windows and DOS developers, need to focus on these alternate channels."

Who Will Siskel-and-Ebertize Software?

Music has radio, movies have previews, and books can be browsed through, but how does a consumer new to computers decide which software titles to buy? The absence of an effective preview mechanism for multimedia is an obstacle that the industry must overcome before software can gain acceptance in the mass market. Today, computer-savvy consumers buy products based on computer magazine reviews and word of mouth. Novice computer users are left to base software purchase decisions on software box graphics, a situation analogous to choosing a car based solely on paint color.

Richard Miles, president of Re:Launch, another consulting firm that helps software companies sell

Distributor type	Traditional channel	Emerging channels				
	Software	Entertainment			Direct mail	Online
Retail outlet	Software stores	Book-stores	Music stores	Video stores	"Try-and-buy" CDs	Online services
	Computer superstores				CD-of-the-month clubs	
	Consumer electronics stores				Market-focused CD inserts	
	Warehouse stores				CD samplers	

A snapshot of emerging software channels.

titles, says, "The rise of a true consumer software market will mean a lot of changes, and the first will be that the outside of a box will no longer be sufficient to get your product message across. Only by direct marketing through CD samplers and online demos will software publishers really have a chance to sell their potential customers and convey a full story."

Tamer adds, "I believe video stores, online services, and some cable TV stations will predominate as the vehicles for software previews. Also, CD-ROM magazines that focus on specific themes and lifestyles—for example, *KidSoft*—will provide another preview mechanism." (For more details about *KidSoft*, a mail-order software catalog/demo CD/kids' magazine, see the section "The Mailbox Connection" on page 25.)

Although traditional media stores have always been segmented by the nature of their content—music, video, and the printed word—virtually all of them are looking at multimedia CDs as a way to bring new customers into their stores. As personal computers continue to move into the mainstream, it's natural for these outlets, especially the large chains, to use this trend to gain a competitive advantage. As you'd expect, specialty stores, such as video game outlets, are making the transition quickly, while more traditional businesses such as the booksellers are reticent. But progress is being made: In 1994 I saw tests and pilots conducted in virtually all of these outlet types. Apple has made substantial investments in two such efforts: a bookstore kiosk initiative and a CD-ROM rental pilot with Blockbuster Entertainment. (For details, see pages 27 and 25, respectively.)

I'll now discuss, in detail, recent developments and future prospects for specific retail channels.

The Bookstore Channel

If your product line includes content titles—in other words, encyclopedias, children's "edutainment," how-to, or reference titles—bookstores may be the best long-term vehicle for selling your products.

This year there has been a great deal of talk about multimedia in the bookstore channel, and some stores are experimenting with limited offerings. In November, the Barnes and Noble bookstore chain drove a stake in the ground by dedicating more than 15 feet of store shelf space to 60 CD-ROM titles. And Apple cosponsored a bookstore kiosk pilot with five other software publishers, which the *Wall Street Journal* called "the most ambitious electronic book demonstration project . . . aimed at winning over avid book vendors to the new technology." (See page 27 for details.)

Ingram Book Company, a subsidiary of Ingram Distribution Group, Inc., has been selling multimedia software through bookstores for over a year now. According to Chris Anderson, division manager of Ingram's New Media group, their business through this channel has seen tremendous growth, doubling every month since June. He sees this growth continuing through Q1 1995 as the pipeline fills and new computer purchasers start loading their computers with software. "The channel for content-oriented software is going to be big," Anderson says. "All the trade publishers are talking about it."

Anderson offers a word of advice to developers considering marketing through this channel— take note of the industry's different packaging and returns requirements. Product packaging needs to display the ISBN (International Standard Book Number) and the Bookland EAN (European

Article Number) bar codes rather than the UPC (Universal Product Code) standard used on software boxes. In addition, you have to be prepared for 100 percent returnability of unsold products at bookstores.

Another important difference between software retailing and other types of media is product risk and life cycles. In the music, book, and video businesses, sales rates are much less predictable. The ramp for product sales is slower and the standard 90-day return policy can throw a wrench into your inventory plans.

Anderson's advice for developers wanting to sell into this channel is to work with a major publisher through an affiliate label program. "The types of products that work best in this channel are reference titles and children's titles like the *Living Book* series," he says. "We find our titles through retailer sales information, trade shows, and software publisher demonstrations."

The Music Channel

Pioneering interactive music CDs featuring artists such as Peter Gabriel, David Bowie, Todd Rundgren, and the artist formerly known as Prince have opened the eyes of music industry executives to the profit potential of interactive media. By building up interactive title creative teams and distribution channels, music studios can repurpose existing music properties and attract new talent to their labels.

Bertelsmann Music Group (BMG), best known in multimedia circles for distribution of *Jump: The David Bowie Interactive* CD, (produced by ION), is currently staffing up an interactive media sales group. BMG Direct is considering initiating a CD-ROM–of–the–month club as early as next year. At the helm of this effort is Christian Jorg, their vice president of new technologies.

Jorg sees dramatic changes in the music distribution channel in the next few years.

"In the past, CD-ROM purchases were made by 'early adopters,' the type of consumer who based purchases on magazine reviews, then ordered from catalogs," says Jorg. "As the number of $1,000 computers with built-in CD-ROM drives increases, the profile of these users will change dramatically. These new customers will tend to make impulse buys in video and record stores, a change that will require distributors to focus more on merchandising. This takes money, and in the next couple of years, we'll see major structural changes in the distribution channel as wholesalers merge to take advantage of economies of scale."

Distribution in the traditional music channel is two-tiered. Large record stores usually buy music products directly from the major labels to receive volume discounts. Independently owned stores often go through "jobbers" or "one-stops" that stock a wide variety of titles but don't provide the deep discounts of buying directly from a label. Until software becomes more common in these traditional channels, industry experts recommend that you cross-merchandise and cross-promote interactive music products to both the software and the emerging music distribution channels.

One distribution concept that's being piloted in record stores for music are CDs produced in retail stores as "one-offs" from a single master. This approach saves packaging costs and helps retailers reduce inventory carrying costs. New Leaf Entertainment is currently developing and marketing systems testing this concept. They've developed a prototype system that stores digital content on a large server and transmits it over

a network to different writing devices capable of creating CD-ROM discs, cassette tapes, and video game cartridges.

On the technical side of the interactive music business, an important issue is the need for a single standard format that would enable users to play CDs on both audio players and computers. Such a standard would make it more commonplace for artists to fill the leftover space on an audio CD with computer-readable media, creating a prime opportunity for multimedia developers.

So, on the creative side of the business, what is the music industry looking for in interactive media? Jorg challenges developers to think creatively: "Go beyond just arranging existing music properties on a CD. I'm still looking for a ground-breaking music CD, something more compelling than what's out there today."

The Mailbox Connection

Many of you may find that an effective way to sell to specific consumer audiences is through their mailboxes, by providing users with CD-based software demos that consumers can try out at their homes or offices. Though many of the initial efforts in this area have failed, a few have done well enough that the experts think that this marketing vehicle is here to stay.

One company that has successfully built a software distribution business using this approach is KidSoft. KidSoft sells children's software by delivering three mail-order vehicles to "Club KidSoft" members—an informative catalog, an entertaining kids' magazine, and a CD with product demos that enables both parents and kids to try products before they buy. Customers can download some of the products directly from the CD, by calling the company's 800 number to exchange credit card information for a code that unlocks the CD-resident application.

"Club KidSoft offers its members a choice of purchasing software from home by downloading it from our CD or through traditional mail order," says Rick Devine, president of KidSoft. "This approach has enabled the Club to enjoy tremendous growth since its inception last year. We find that members like having a choice in how to buy their software, and most of all, they appreciate the ability to get information and try software at home."

Since the shipment of their holiday issue, KidSoft has noticed that sales from application "unlocks" have increased exponentially. "This trend confirms that this form of distribution is growing in customer acceptance," says Devine.

Another advantage to distribution vehicles like the Club KidSoft CD is that their focus on a specific market segment can breathe life into niche markets that couldn't survive on retail sales alone. Typical software retailers can only afford to carry a few children's edutainment titles. KidSoft can afford to carry a wider selection, including innovative new products from smaller single-title developers. (Developers interested in selling products through KidSoft can send product samples to KidSoft, Attn: Mary Crum, 718 University Avenue, Suite 112, Los Gatos, CA 95030-3313. Tentatively scheduled themes for upcoming issues are travel/mystery in May, music/creativity in July, and back-to-school/career in September. Submitted products must be "positive, nonviolent, educational, and fun for kids.")

Here are several variations on the theme of "try-and-buy" CDs:

• *CD-ROM–of–the–month clubs.* It was bound to happen—the offering-of-the-month-club concept that has worked so well in the book and music industries is now moving to multimedia. In these scenarios, customers have a featured title offered each month and a series of alternatives from a catalog, offered on CDs with previews and promotional materials. I know of three such distribution efforts that are under way. BMG Direct, mentioned previously in this article, is planning a

Whatever Happened to the Blockbuster Pilot?

A little over a year ago, Blockbuster Entertainment, the video rental giant, began testing the feasibility of renting CD-ROM titles and players in 57 San Francisco Bay Area stores. At each of the test sites, kiosks for five major hardware platforms were set up, providing consumers with a preview mechanism for their title rentals. Participating hardware companies included Apple Computer, IBM, Panasonic (3DO players), Philips Electronics (CD-i players), and Sega.

Eight months later the displays were dismantled, leaving eager developers wondering, was the Blockbuster pilot a success? —Yes, says Jack Ferry, Blockbuster's director of new media, who gave the initial trial two thumbs up. Ferry's group has been using the resulting data to build financial and coverage models that they can use in planning their new media expansion. "The test stores in the San Francisco Bay Area have been consolidated down to 15 stores, and an expansion plan has been developed for setting up additional stores in other geographic areas during 1995," says Ferry. "Looking to the future, I have a positive outlook for Blockbuster's multimedia rental business, especially in light of Viacom/Paramount's investments in the content-creation side of the business."

Other good news for the Macintosh platform was that during the pilot the hardware and titles got excellent reviews for ease of use. Michael van der Keift,

Blockbuster's director of business development, reported in *Advertising Age,* May 23, 1994, that ". . . the Macintosh system's 'plug and play' characteristics made for easy use." He went on to say that "more systems will need to mimic that capability before they will be widely accepted."

Publishers interested in pursuing this channel should take note of the data that Blockbuster collected on the customers who rented Apple Macintosh or IBM PC systems:

• 86 percent were male with a mean age of 37
• 72 percent were married
• 70 percent had children
• their mean household income was $75,400

The dominance of male renters was a surprise to Blockbuster, given that their video renters are split evenly down gender lines. "My guess is that the lack of titles that interest women is an important factor in these results," says Ferry.

From Apple's point of view, the Blockbuster trial was critical in meeting some needs in the CD-ROM marketplace. It gave the public a preview capability, both through the demonstration systems in the stores and the rental of Macintosh computers. The pilot also provided a showcase for Macintosh CD-ROM titles. And finally, it helped demystify multimedia in the eyes of the many customers who frequent Blockbuster stores.

club offering possibly by early next year. Columbia House, already a player in the music CD club business, is also planning to offer CDs through a club format in 1995. And the Newbridge Book Club, well known for its extensive library of computer and technology books, will offer CD selections this year.

• *Magazine inserts.* A number of consumer magazines targeting CD-ROM player owners have started to package their own demonstration CDs with their print magazine each month, providing you with another promotional opportunity. For example, *CD ROM Today,* a publication that targets early adopters of Macintosh and Windows CD drives, ships with a monthly CD-ROM that includes dozens of Windows and Macintosh demos. Title publishers who advertise in this publication get free space on the demo CD and a corresponding product brief in the magazine. (For details, contact *CD ROM Today*'s editor, Lance Elko, at 910-852-6711.) On the international front, *Mac Easy* in Germany, Soft•CD in the United Kingdom (on the cover of *MacFormat* magazine), and Info-Express in Australia offer demo CDs with magazines.

• *Review-book inserts.* A few companies (McGraw Hill and Que Publishing, to name two) are offering CD title review books with accompanying CDs that include title demos; at least one company offers a guide on CD

alone (*The ROM Finder*). These avenues are available to you after an evaluation process. If you can provide a demo in the same issue as a rave review, all the better!

• *Low-cost samplers.* Hawked through telemarketers and at the checkout counters of software stores, low-cost software samplers were very visible in 1994. (Samplers usually contain short demos of one particular line of products, such as all the Living Books titles from Broderbund.) They sell for between $4.95 and $9.95 per CD, and, in my opinion, many are just advertising without much substance. Software stores, however, tell me that they're selling like hotcakes. (It just goes to show you how desperate CD drive owners are for previews.) Expect to see many more of these samplers until a standard preview capability emerges in the industry.

Distribution Over the Net

Another viable distribution alternative is the combination of a CD-ROM and online service product. The two delivery techniques take advantage of the richness of multimedia on CD-ROM and the immediacy of the online services for up-to-date information. As developers are painfully aware, multimedia content is too expensive and too slow to download over phone lines today. High-quality graphics, sound, and video clips are best distributed to subscribers on CD, with more compact updates delivered

through online services such as eWorld, America Online, CompuServe, or Prodigy.

Apple is working on two fronts to legitimize this alternative. First, Apple is a partner in 2Market, a company with a CD-ROM and online shopping service that provides shoppers with a unique way to order products electronically. (See page 27 for details on this venture.) And, for the long term, Apple's eWorld online service is working on developer application programming interfaces (APIs) and transaction-based services that will enable developers to carry out credit card sales online, then deliver these products over the net. For more details on these APIs, see the article "eWorld Marketing Opportunities," *Apple Directions,* September 1994.

Positioning Yourself for the Future

The key to surviving the software market transition that's underway is to "multiplex" your channel efforts across traditional and emerging channels.

Joey Tamer of SOS, Inc., offers advice for preparing for this transition: "By the end of the decade these channels should be more stable: Stories and reference CDs will be sold through bookstores, games through video stores, productivity and game titles through software stores, and MTV-type titles through music stores. Software publishers

should prepare now by gaining a thorough understanding of their title's 'genre' and which channel will market these types of titles."

No matter which channel works best for your product—entertainment stores, direct mail, or try-and-buy CDs—it would be wise to invest time in learning about the people, distribution intricacies, and customers of that specific channel. Becoming a member of the Apple Multimedia Program is one of the best ways to learn about these channels. Program members benefit from market research reports, resource directories, online discussion boards, comarketing and networking opportunities, and invitations to multimedia events. And finally, it wouldn't hurt to have an electronic demo of your product ready, so that you can take advantage of marketing opportunities that arise.

By putting time and effort into establishing yourself in emerging channels today, you'll be a step ahead of the competition when software channel dust settles. ♣

Vicki Vance is the manager of the Apple New Media Division's CD-ROM Initiative. She is also co-editor of Emerging Distribution Models for Consumer Interactive Media, *a book published in October 1994 for Apple Multimedia Program members by Multimedia Business Development Group, New Media Division, Apple Computer.*

Apple Initiatives

Apple Promotes 17 "Hot CDs for Macintosh"

The "Hot CDs for Macintosh" initiative, along with the Apple CD-ROM Collection described in the next section, represent perhaps the biggest dollar investment that Apple has ever

made in merchandising third-party software products. The first phase of this program, running from the middle of November 1994 through January 1995, aims to boost the sales of 17 third-party Macintosh CD-ROM titles. (The CDs in this first line-up were chosen mainly for their sheer sales potential.)

Bottom line, this initiative motivates retailers to carry more Macintosh CD products in exchange for co-op funding and aggressive advertising that will get more Macintosh customers into stores. Apple is providing retailers with high-impact display racks and "shelf talker" signs, a promotional mailing to 300,000 CD-ROM owners, co-op ad art, radio spots, and a tie-in to Apple's worldwide

multimedia ad campaign. In exchange for these efforts, which should result in greater title sales, participating developers agree to provide 100 percent returns for the program period and pay retailers a 5 percent bonus co-op fee (based on net sales).

This initiative, along with Apple's "Multimedia You Can Use" advertising campaign, should really get holiday season

shoppers into the retail stores and reinforce the reality that there are many excellent software products on the Macintosh platform. If the retailers and developers are pleased with the outcome of this promotion, Apple hopes to continue it throughout 1995, possibly adding more titles to the line.

Apple expects 350 to 400 stores to participate in this program by the end of the year. Major retailers who have already signed up for the promotion include Egghead, Computer City, Fry's Electronics, Creative Computing, Computer Attic, and J&R Computer World.

Apple's "Collection" Provides Visibility to Many Apple-Related Products

In order to increase the visibility of great Macintosh-based multimedia titles to customers worldwide, Apple began a holiday season promotion for the Apple CD-ROM Collection, a family of eight CD-ROM title packs. Each of these bundles includes three CDs chosen around a common theme and *Xpand Xpo,* a CD-ROM–based virtual trade show featuring demonstrations of all Collection titles, as well as information on more than 10,000 other Macintosh-related products. At $43.50 (U.S. suggested retail price), the Apple CD-ROM Collection effectively offers three CD-ROM discs for the price of one. The Collection line-up currently includes the following bundles:

- Smart Fun for Kids
- Cool Fun for Kids
- Adventures for Kids
- CyberArcade
- Holiday Selections
- Sports and Humor
- Encyclomedia
- Learning for Adults

"These bundles provide a great value to Macintosh customers and high visibility to participating developers," says Andrew Scoular,

the CD-ROM Collection program manager. "And retailers benefit from Macintosh add-on sales and shelf space savings, since our three-packs fit in the space of one standard title."

Apple also plans to continue this program, though developer participation is obviously limited. For the next round of CD-ROM Collection picks, Scoular says he's looking for titles with compelling content that highlights the advantages of using a Macintosh for multimedia. Developers interested in learning more about "Collection" bundling opportunities can contact Apple's program partner, Tom Gaunt of Double Impact Multimedia, at 214-386-8792 or at the AppleLink address GAUNT.T.

The *Xpand Xpo* CD is produced independently from Apple and is distributed worldwide. It enables shoppers to "walk through" a beautifully rendered 3-D Macworld-like trade show floor on their computers and view animated product demos. Developers can find out more about renting virtual booth space or product listings on the CD by contacting the U.S. Xpand office at 408-280-1919 (AppleLink address: XPAND) or the United Kingdom office at 44-709-855497.

Kiosks Capture Interest of Bookstores

Apple and six software companies have just completed a pilot program that tested the sales potential of interactive software in bookstores. The pilot program had two objectives. The first was to convince booksellers that electronic media can be a viable extension to their product offerings—just as audio tape versions of books have been in the recent past. The second objective was to test the acceptance of these new media with typical bookstore customers, and garner more visibility for Macintosh-based titles.

Seven companies sponsored this pilot: Apple Computer, The Voyager Company, Creative Multimedia Corporation, Time Warner Interactive, Macmillan New Media, Putnam New Media, and Discis Knowledge Research. In the pilot sites, Apple provided kiosks with CD-ROM–equipped Macintosh systems on one side and Apple PowerBook computers with an expanded book series from Voyager on the other. Voyager provided personnel at the bookstores to demonstrate both the hardware and titles to customers.

Results of the trial, held for ten months in six cities, were very positive. In the opinion of both store personnel and customers, the preview capability of the kiosk was the most popular aspect of the experiment. Interactive media titles sold slightly better than expanded books, with a 60-percent share of sales.

Of course, the overall success of the pilot program will ultimately be measured in the increased interest the retail book community takes in carrying electronic products in the coming year. The jury is still out, but everyone seems to agree that it is just a matter of time before the large book chains make electronic media a regular part of their lines, with independent booksellers following shortly thereafter.

2Market Pioneers CD and Online Shopping

2Market is an innovative interactive shopping service that includes products ranging from clothing to software from name-brand retailers such as Spiegel, The Sharper Image, and The Nature Company. Based on the pilot CD called *En Passant,* 2Market services are being delivered on CD-ROM for Macintosh and Windows computers and on leading online services, including America Online and Apple's eWorld. 2Market, Inc., is an

independent company receiving financing, technology, and marketing expertise from Apple Computer, America Online, and Medior.

2Market provides developers with another vehicle for selling products, but perhaps its most important role is as a pioneer in delivering online and CD-based shopping to home computer users. Technical innovations on this CD include a search engine that helps shoppers select gifts for specific individuals and an electronic link that enables shoppers to order products directly from the CD, by transmitting product orders via modem. Product ordering is all handled electronically after shoppers have had an opportunity to browse through the CD-based catalog. While online, shoppers can look over new product additions and sales items, and they can even communicate directly with specific retailers.

2Market service is currently accessible through America Online and eWorld, and it will soon be available through other online services.

Macintosh Bundles Move More CPUs

Apple is shipping increasing numbers of market-focused Macintosh CPUs bundled with third-party software. For example, the Macintosh Performa 630 series systems, which target home users with children, are shipping with The Writing Center from The Learning Center, KidWorks 2 from Davidson, Thinkin' Things from Edmark, and the *KidSoft* CD. Apple's Education group also puts together many bundles, such as the Power Macintosh Business Education Bundle, the High School Biology Bundle, and the Teacher Solution Bundle, to name a few.

This kind of bundling provides customers with a positive "out-of-box experience" and developers

with "virtual" shelf space in the homes, schools, or offices of potential customers. Participating developers receive only a modest royalty on each bundle shipped, but each developer's payoff is most likely to be longer term—upgrade revenues and sell through of other products in their line. And bundling titles sells more Macintosh computers, expanding the Macintosh installed base at a rapid rate and increasing the number of potential customers for all developers. Though bundling opportunities are limited, developers interested in exploring this possibility should contact individual Apple CPU and peripheral product line evangelists.

Blockbuster and Apple Collaborate on CD Rentals

In November of 1993, Apple teamed up with Blockbuster Entertainment to test the feasibility of renting CD-ROM titles in their video rental stores. (For details, see the text box "Whatever Happened to the Blockbuster Pilot?" on page 25.)

This initial pilot program gave Apple and participating CD-ROM title developers excellent product exposure, and it provided Blockbuster with data that they're using to design a permanent title rental business. The prospect of having Macintosh title rentals in thousands of video rental stores across the United States would provide a significant boost to the software business, and Apple is working closely with Blockbuster as they formulate their CD-ROM rental expansion plans.

Apple France and Australia Co-Sponsor Sampler CD-ROM Discs

For French Macintosh users, Apple France and CDR Informatique, a CD-ROM content provider, have teamed up to create a CD-ROM–based catalog of about 3,000 Apple-related products. (CDR is managing the project and Apple France is providing product listings and some funding.) The CD also includes demo versions of 150 software products and a text-based product database that is being made available through the Minitel phone system. Minitel shoppers can access the same services as the CD-ROM sampler with an additional feature—since Minitel is text-based, it allows for searches on any word in the database. A paper version of the database will be distributed by mail as well. This CD will probably be updated every six months, and developers interested in participating in the next version can contact Sylvain Deitrich at the AppleLink address CDR.INF.

Down under, Cameron McDonald-Stuart, Manager of Apple Australia's New Media Group, has been working on two efforts that will benefit developers: the Australian CD-ROM Sampler and a Macintosh demo station pilot that targets Australian book, video, and music retail outlets. Their sampler CD features 34 titles, half of which will include interactive demos. Their retail pilot program, slated to begin in early 1995, will provide stores with a Macintosh demonstration unit and a range of titles on consignment. Apple Australia plans on signing up 20 stores in Sydney and Melbourne for the pilot.

Apple Multimedia Program Distributes a Developer Showcase CD

The Apple Multimedia Program (AMP) Developer Showcase CD-ROM is a strategic marketing tool designed to promote and showcase AMP member products and services. On this internationally distributed CD, AMP members can include company information; marketing and technical contacts; a list of clients and associated projects; and a Macromedia Director, QuickTime, or Apple Media Kit movie advertising AMP member products, services, or both. The Developer Showcase was designed to allow members to network, to showcase multimedia developed for the Macintosh, and to provide groups who outsource multimedia products and services with a convenient directory. For more Showcase information or an AMP membership application, call 408-974-4897.

Apple's Commitment to Developers

The sales of Apple multimedia computers are booming, and by next year Apple expects that virtually all Macintosh computers will offer CD-ROM players as a standard option. Apple intends to make the most of this momentum by aggressively helping third-party Macintosh software claim more visibility and shelf space. Though the specifics of the programs under way may change, you can expect to see more programs that help you sell more products. ♣