Inside this issue

TECHNOLOGY & INDUSTRY

1992 WORLDWIDE Developers Conference sessions are now available on audio cassette and videotape.
(see the News Folder)
***

TO DISCUSS AND GET HELP with MacApp, there are a number of AppleLink group addresses you can join.
(see the News Folder)
***

IT SHIPPED! announces new and revised third-party products that shipped in March 1992.
(see the News Folder)
***

THIS MONTH'S CD, "The Byte Stuff," features the latest *develop* (Apple's quarterly technical journal), as well as the Apple Events & Scripting Development Kit and a variety of other information and materials for developers.
(see the News Folder)
***

APPLETALK for Programmers is a new networking course offered by Developer University. It's only one of a number of self-paced courses, hands-on classes, and reference tools available to developers.
(see the News Folder)
***

APPLE IS EASING version number problems with its world-ready release of System 7.
(see the News Folder)
***

APPLESCRIPT WILL BE shipped this year, making it easier for developers to implement Apple events.
(see the News Folder)
***

THE OPEN COLLABORATION Environment was the subject of an entire day of sessions at the 1992 WWDC. This month's article summarizes the latest O.C.E. developments, including store-and-forward features.
(see the News Folder)

***

APPLE DIRECT **BIDS** goodbye to Bruce Tognazzini.
(see the News Folder)

***

APDA LAUNCHES its new Hot Product of the Month promotion in this issue, and makes a special limited-time offer. Also, several new Apple and third-party products, including DAL servers, have been recently released.
(see the News Folder)
***

BUSINESS & MARKETING

GEOFF MOORE of Regis McKenna describes how you can move a product from initial success to mainstream market leadership .
(see Business & Marketing Folder)
***

DIRECT MAIL marketing saved Working Software from an untimely demise. The company's president recounts how direct mail helped them to rebound.
(see Business & Marketing Folder)
***

MARKET RESEARCH QUARTERLY

THIS SECTION makes its debut this month. Market Research Quarterly gives you International Data Corporation’s (IDC) latest computer-market data and analysis. Look for this section each quarter to help you stay up-to-date with timely data about Apple's markets. This month we start with a discussion of the worldwide Mac installed base.
(see Market Research Quarterly Folder)
***

THE MACINTOSH installed base has increased more than seven-fold since 1986; IDC anticipates it will continue to increase an impressive rate.
(see Market Research Quarterly Folder)
***

IDC GIVES YOU the latest installed base information on Apple’s product mix.
(see Market Research Quarterly Folder)
***

Solution Puts PC Disks on Desktop

Macintosh PC Exchange

Almost 50 percent of all Macintosh users interact with PCs (IBM personal computers or compatibles), and many Apple customers have asked for an easy way to convert files on 3.5-inch floppies between Macintosh and PC systems.

Because of the demand for an Apple-labeled solution, last month Apple released Macintosh PC Exchange, a software extension that allows a Macintosh with a SuperDrive floppy-disk drive to read and write files on standard 3.5-inch PC floppy disks.

Macintosh PC Exchange also allows PC disks, subdirectories, and files to appear on the Macintosh desktop as normal disks (with “PC” on the icon’s label), folders, and files, respectively. Users can manipulate PC floppy disks and their contents with normal Macintosh gestures—clicking, double-clicking, and dragging.

Macintosh PC Exchange requires a Macintosh running System 7 or later and 3 megabytes of memory (4 megabytes recommended). Macintosh PC Exchange sells for a suggested retail price of $79 in the U.S. (with localized versions to be available in other countries); users are entitled to one year of toll-free telephone support.

Macintosh PC Exchange meets the needs of the many users who want an intuitive way of exchanging files between Macintosh and PC (both DOS and Windows) formats. The interchangeability of file formats (that is, *file translation*) has become less of a problem since many third-party companies have introduced both Macintosh and PC versions of their products.

Among the most popular applications available for both platforms are Lotus 1-2-3, WordPerfect, Microsoft Word and Excel, Aldus PageMaker and Persuasion, Ventura Publisher, and Adobe Illustrator. With these programs, users can customize Macintosh PC Exchange to automatically open a file within a compatible Macintosh application when they double-click on a PC file on an inserted PC floppy.

Also, many other programs can exchange files via Macintosh PC Exchange if you save a document in one of the above file formats. For example, Borland Quattro (on the PC side) and Claris Resolve (on the Macintosh side) can exchange files if documents are saved in the common spreadsheet .WK1 format, which both prod-ucts support. Apple designed Macintosh PC Exchange to be a simple file-exchange program that meets most users' needs. Apple recommends that users needing to translate files between application formats use existing third-party file-translation utilities.

One of the features that distinguish Macintosh PC Exchange from similar products is that Macintosh PC Exchange is System 7-savvy. Because of this, users can make aliases of files on PC floppy disks, and the System 7 Find command can search on PC disks. Macintosh PC Exchange also supports the new System 7 file system calls.

New IIGS System 6, HyperCard Upgrade Available

New system software and an upgraded version of HyperCard IIGS are available for the Apple IIGS personal computer. IIGS System 6 software offers an enhanced user interface, greater speed and performance, and data-exchange capabilities not available on current Apple IIGS operating systems.

HyperCard IIGS version 1.1 has been upgraded to include a Media Control stack for operating CD-ROM drives and videodisc players, and new HyperTalk scripting capabilities.

Apple IIGS System 6 features three new File System Translators that provide easy access to Macintosh disks, Pascal disks, and Apple II DOS 3.3 disks. In addition, it offers significant feature enhancements to control panel and desk accessory functionality, providing an enhanced new "look and feel" to the Apple IIGS. Control panels can be opened directly from the desktop, and Find File and Calculator desk accessories have been incorporated.

Finder Help can now be accessed through pop-up menus and can be kept on screen while users step through procedures. Window handling and window appearance have also been enhanced, making it easier to move between and manage multiple windows on a single screen.

Apple IIGS System 6 also offers two new applications—Teach and Archiver. Teach is a desktop text processor that enables you to jot down notes, read disk files, and create formatted or unformatted text documents.

Teach also provides file-im- port capability from the ASCII, AppleWorks version 3.0, AppleWorks GS, MacWrite version 5.0, and AppleWriter formats.

Archiver offers flexible hard-disk backup functions to save and restore either individual files or entire volumes.

The Apple IIGS System 6 Media Control Toolset is a new tool/ driver/control panel/desk accessory combination that allows users to integrate, configure, and manage highly sophisticated multimedia effects. The performance of this toolset is optimized when used in conjunction with HyperCard IIGS version 1.1.

The Universal Access suite (also available on Macintosh System 7 software) opens the Apple IIGS to disabled users via the Video Keyboard, Easy Access, and Closeview programs. It simplifies system use for the visually or physically impaired.

System 6 also improves networking functionality via EasyMount, a new feature that allows users to mount a network server with a simple double-click command.

Apple IIGS System 6 replaces Apple IIGS System 5.0.4 for the stand-alone Apple IIGS, providing a consistent graphical interface and high performance for individual as well as networked users.

HyperCard IIGS. HyperCard IIGS version 1.1 features a Media Control Stack for the control of external media devices such as laserdisc players and CD-ROM drives from within the stack. Control is provided to the user through a common interface to two external devices, by standard buttons such as Play and Fast Fwd or floating control panels.

A Tune Builder stack lets users easily create short original tunes by clicking the mouse on the stack's music staff. Notes can be played in variable time and voices from a large collection of instruments. Users can then use these tunes as enhancements to other original stacks by cutting and pasting.

HyperCard IIGS version 1.1 incorporates features from Macintosh HyperCard version 2.0 and 2.1, such as HyperTalk extensions and windoids (floating windows). Like the original HyperCard IIGS, version 1.1 is in color.

The Apple IIGS SuperDrive Controller card is also available for the enhanced Apple IIe and Apple IIGS, allowing users to utilize Apple's SuperDrive—which permits the use of 1.4-megabyte floppy disks. The SuperDrive card also operates all other Apple II 3.5-inch disk drives.

System Requirements. Stand- alone Apple IIGS System 6 software requires at least 1 megabyte of RAM, ROM version 01 or 03, and one 3.5-inch disk drive, although a configuration of 2 megabytes of RAM and a hard drive is recommended for optimal performance.

Networked systems require at least 768K of RAM, ROM version 01 or 03, and appropriate Local-Talk cables.

HyperCard IIGS version 1.1 requires 1.5 megabytes of RAM, one 800K disk drive, a hard drive or connection to a networked environment, and system software version 5.0.3 or later.

Price and Availability. The System 6 package includes six 3.5-inch disks containing system and setup software and system tools, as well as the *Apple IIGS System Software User's Guide.*

The package is available from authorized Apple dealers; Apple Education sales consultants; and Resource Central, Inc. [(913) 469-6502], at a suggested retail price of $39 in the U.S. Apple is also making Apple IIGS System 6 software available from licensed user groups and licensed on-line services.

HyperCard IIGS version 1.1 is available from authorized Apple Dealers and Resource Central, Inc. for a suggested retail price of $69. HyperCard IIGS 1.0 owners can purchase an upgrade to version 1.1 from Resource Central, Inc.

Imaging in the ’90s

Graphics, Type, Printing Improvements Ahead

by Gregg Williams,
Apple Direct staff

With personal computers, if you can’t see it or print it, it doesn’t exist. That’s why Apple engineers have spent the past few years working on the next generation of graphics, printing, and typography for the Macintosh.

You should be interested in the results: These new technologies will give you the power, control, and ease of development that you need to write the leading-edge applications of the 1990s, programs that are difficult or impossible to write today.

By looking at today’s Macintosh imaging technologies and listening to developer and user feedback, we found several things we wanted to add to our next generation of imaging technologies. These technologies will deliver more powerful and more consistent color-management tools, a much improved system of typography, and much better printing and control over the printing process.

The sections below give you a few details about what’s ahead. Much of this material comes from presentations Apple engineers gave attendees at the 1992 Apple Worldwide Developers Conference (held May 11–15 in San Jose, California). For background information, see “Blueprint for the Macintosh,” in the April 1992 *Apple Direct*.

QUICKDRAW GX GRAPHICS SOFTWARE

QuickDraw GX is the next generation of graphics software for the Macintosh. It is a resolution-independent graphics model that gives you far more capabilities than QuickDraw does.

Perhaps its most important characteristic is what it does *not* do. It does not break any existing program or render QuickDraw obsolete: QuickDraw GX coexists with QuickDraw. You can use either in your programs, or you can use both in the same program. However, you will probably want to use QuickDraw

GX. Its built-in features will allow you to finish your program much faster than you could with QuickDraw.

Object-Oriented Drawing. Unlike QuickDraw, which is *pixel*-oriented, QuickDraw GX is object-oriented. I don't mean "object-oriented" in the programming sense but rather in the sense that you deal with self-contained entities and manipulate them as such. (For example, the same QuickDraw rectangle may be drawn differently, depending on, say, the pen width of the current graphics port. A QuickDraw GX rectangle "knows" all its attributes; it is more like a physical object in that it will always be drawn the same.)

In QuickDraw GX, the class of objects includes points, lines, curves, rectangles, polygons, text, bitmaps, and pictures. (They are also collectively called *shapes.)* Given an arbitrary shape, you can modify it with any of several transformations, which include scaling, rotating, translating (moving), and clipping (drawing within a given area), among others.

QuickDraw GX will often do work that you once had to do manually (if you decided to do it at all). With QuickDraw, you must write your own code to rotate and scale a bitmap; in QuickDraw GX, the code to do that is already built in.

To create complex images, you use a *picture,* an ordered list of shapes and subpictures (each of which is also a picture). QuickDraw GX procedures allow you to add, remove, and manipulate shapes within a picture. You can also use a debugging envi- ronment called GraphicsBug to "debug" shapes and pictures at
a symbolic level.

Since text is just another shape, it can be manipulated with any available QuickDraw GX transformations—it can be stretched, skewed, distorted, rotated, whatever. This one feature makes it easier (if not trivial) to add many high-end type manipulation features to your programs.

QuickDraw GX allows extensive use and manipulation of color. It lets you specify color by using one of several different color spaces, including RGB, HSV, HSL, CIE, YIQ, LAB, and LAV. You can also create your own custom color spaces.

In addition, the ConvertColor routine can convert a color from any color space to any other, and other QuickDraw GX routines exist for dithering a picture or turning it into a halftone.

THE NEW TYPE PLATFORM

The New Type Platform consists of two separate pieces, the Line Layout Manager and TrueType 2.0.

Line o' Type. The Line Layout Manager, a new addition to the Macintosh Toolbox, provides powerful routines and data structures that give you considerable control over one basic unit of text—a single line of type. The Macintosh has always provided better written display of more of the world's languages than any other personal computer. This new manager makes the Macintosh even more powerful in its ability to handle printed text.

The Line Layout Manager allows you to create and use *layouts,* which are shapes (in the QuickDraw GX sense of the word) that describe the appearance of lines of text. A layout describes a single line of text, which contains one or more *runs* of text. (A run is a contiguous segment of text in the same script—Japanese, for example—and with the same styling.)

This manager can handle left-to-right, right-to-left, and top-to-bottom layouts. Because a layout is a shape, it can be manipulated by QuickDraw GX—in other words, layouts can be drawn, stretched, clipped to a given area, and so on.

Scripts and Glyphs. The Line Layout Manager is not tied to any one human language. It handles all the scripts (writing systems) that Apple supports, including lines that contain a mix of both single- and double-byte characters. (All western scripts use a single byte to represent a character. Many eastern scripts—Japanese, for example—need two bytes to represent a character.)

The Line Layout Manager manipulates *glyphs,* which represent the visual appearance of (depending on the situation) a character, part of a character, or a combination of characters. This manager supports very sophisticated glyph manipulation, including automatic or manual kerning, tracking, and letterspacing.

If the proper rules are built in to the font being used, the Line Layout Manager can interpret the actual text, as stored inside the Macintosh, and automatically substitute the correct ligature (such as *æ* for *ae*) or contextual form—without your having to add any code to your program or make any changes to the original text as it is stored in the computer. This is particularly helpful in scripts such as Arabic, in which the shape of a glyph changes based on the glyphs that surround it.

One important point about the Line Layout Manager is that many of its features are driven by information within the TrueType font. They then appear automatically when you use this manager to create a layout of text. Equally interesting is the fact that TrueType 2.0 allows font designers to add additional, arbitrary features.

True Type Too. In creating the New Type Platform, Apple engineers decided that certain parts of the solution belonged in the fonts themselves, and this resulted in an expanded TrueType standard, TrueType 2.0.

Several improvements came from the desire to make TrueType 2.0 robust enough to meet the challenge of double-byte scripts:

•TrueType 2.0 fonts can access as many as 65,535 glyphs (up from 255).

•These fonts now support the precise control of text printed vertically (such as traditional Chinese, for example).

•These fonts can contain built-in bitmapped fonts (so a font can better display itself in sizes smaller than the outline font can handle, given a fixed pixel resolution).

•Compound glyphs (needed for complicated glyph shapes) are stored in a more compact form than before.

•Synthetic glyphs allow the Line Layout Manager to create a composite glyph from components, use it, and then throw it away.

Styles 2.0. Everybody knows what styles are—bold, italic, underline, outline, and shadow—right? Well, TrueType 2.0 changes all that. QuickDraw and PostScript may have styles, but TrueType 2.0 has *styles*.

TrueType 2.0 implements algorithmic fonts—that is, fonts whose visual appearance can change if the values along one or more axes are varied. One such axis is that of “boldness.” If you change the bold-axis value, a TrueType 2.0 font can vary continuously between specified minimum (or plain) and maximum (or bold) appearances.

A TrueType 2.0 font can have any reasonable number of axes (also called *styles*), each of which modifies the font independently from the effects of other axes.

Font designers can create and register a new style with Apple. Apple engineers have already defined and registered styles for *weight* (light to bold variations), *width* (condensed to expanded), *posture* (normal to italic), *contrast* (defined by the ratio of vertical to horizontal stem weight), *optical size* (different outline masters for different point sizes), and *ductility* (the amount by which a font modifies itself to accommodate text justification).

OPEN PRINT ARCHITECTURE

The most important feature of the Open Print Architecture is that it's (guess what?) *open*. Not only does the Open Print Architecture make printing easier and more customizable for users but it also makes it easier for developers such as you to create and customize printer behavior when writing a new printer driver or modifying an existing one. The Open Print Architecture will stimulate the growth of output peripherals and enhancements for them.

Faster, Easier Drivers. To developers of output peripherals—dot-matrix printers, plotters, film recorders, and the like—printer drivers are *hell*. The creation of printer drivers is regarded as a black art, and it can easily take months to make and test a modest change to an existing driver.

The Open Print Architecture will change all that. Printing will occur in a device- and resolution-independent way, and you will have less trouble making printer drivers and enhancing them.

One of the reasons that printer drivers will be easier to write is that the Open Print Architecture will supply many of the code pieces that, until now, developers have had to provide themselves.

The Open Print Architecture provides routines that allow a printer driver to convert the image of a page to whatever form it needs: PostScript (for PostScript printers), a raster image (for non-PostScript printers that can print a bitmapped image), and vectors (for plotters—a category that currently isn't supported). Most printer drivers will be easier to write because they are no longer responsible for doing the conversion itself.

Print Extensions. If you want to make a small change in print functionality, why should you have to go back to the original printer-driver code, change that, and then be required to test the entire driver to make sure your code additions haven't broken something? System 7 has system extensions. Wouldn't it be nice to have something like that for printing?

Guess what? The Open Print Architecture provides for small modules called *printing extensions.* Drop them into the System Folder, and they automatically appear in an application's Document Setup dialog box. Printer drivers under the Open Print Architecture use *messages* to request services of the printing system. (Support for messages comes via a new Message Manager, which the Open Print Architecture—or any other code—can make use of.)

Printing extensions can intercept control of the printing process at any given message, making them an extremely flexible way to modify, replace, or augment a printer driver's behavior.

Also, anybody can write a printing extension, because you don't need to have the printer driver's source code. This means there should be a good market for small third-party printer utilities and enhancements supplied as printer extensions.

More Flexible Printing. Options in the Open Print Architecture give both you and the user more flexibility and convenience in the printing process.

A new entity called a *job* takes the guesswork out of printing the same document on a different Macintosh. A job, which is saved as part of a document, specifies the printer used to print the document. If users try to print a document on a *different* printer, they are alerted that the printout may differ from that of the original job.

Each page of a printed document contains a *format;* documents can use the same format for every page, or each page can have its own. A format describes everything about a page: its paper type, margins, size, resolution, color space (for color output devices), and so on. This gives you the flexibility to create applications that can print mixed-format documents. For example, a convention-management program might print an attendee confirmation letter in portrait mode, a spreadsheet page in landscape mode, and (on a different printer) a conference badge on heavy paper.

Like most of the upcoming Macintosh technologies, the new imaging architecture is exciting because it prevents you from having to implement many low-level tasks. This frees you to do more creative things with your programs.

The new graphics, type, and printing architectures will make new products and even new markets possible, and that will keep the Macintosh market—and your company—in good shape for years to come.

Apple Expands Business and Marketing Services for Developers

Providing top-quality business and marketing services—in addition to technical support and services—is a critical element of Apple's efforts to strengthen its partnership with developers. To augment the range and scope of services Apple provides to developers, the Apple Developer Group has improved and expanded the family of business and marketing services offered.

Existing services have been revised, and new services are being created. They range from business consultations and direct-mail lists to market research information and a variety of marketing-oriented tools. The current family of business and marketing services includes:

•**Market Research Quarterly.** This issue of *Apple Direct* features the new Market Research Quarterly section, Apple's most recent vehicle for delivering up-to-date market data to developers (see the Market Research Quarterly folder). The section, to be published four times a year, will contain data and analyses prepared by International Data Corporation (IDC) on vital issues affecting Apple's markets. In part, topics for each issue will be selected based on ideas submitted by developers. (For information about how you can submit ideas for future issues, see "Input Wanted" in the Market Research Quarterly Folder.)

The Market Research Quarterly section will provide data you can use to help make key business decisions and hone your long-term strategic plans.

•***Apple Direct* Business & Marketing Section.** *Apple Direct* has long carried business and marketing articles, and in October 1991 we began devoting an entire section of the publication to business and marketing news, features, and market research information. In addition to marketing news, how-to articles, and case studies from de- velopers, you also receive an events calendar to help in your own conference and trade show planning.

•***Macintosh Services Directory.*** The *Macintosh Services Directory* was created to help you find a broad range of service providers and entrepreneurs that support the Macintosh.

Now in its third edition, the directory has been expanded to include marketing-service providers. It includes descriptions of specialists in the fields of advertising, public relations, direct mail, market research, and more. When you need to supplement your internal marketing resources to get a job done, the directory can help point you in the right direction.

The new, updated version is in this month's developer mailing. You can also order the directory by calling APDA at (800) 282-2732 (U.S.), (800) 637-0029 (Canada), or (408) 562-3910 (outside the U.S.) or by sending an AppleLink message to: APDA.

•**Direct Mail Program.** With this program, members of Apple's developer programs around the world (including European and Pacific-based developers who distribute products in the U.S.) can access specific Apple mailing lists. You can take advantage of this program to reach new customers with direct mail campaigns, and then utilize the response from such mailings to edit and expand your own mailing lists.

The types of lists available from Apple include end users, user groups, authorized Apple dealers, value- added resellers, and developers. More information about the program is available on AppleLink (path—Developer Support:Developer Services:Marketing Information and Services:Marketing Services:Apple Services).

•**AppleLink Marketing Information & Services Folder**. If you haven't browsed through the Marketing Information & Services folder on AppleLink lately, take a look. The folder now includes market research data (which will include back issues of the *Apple Direct* Market Research Quarterly section), information about marketing services available from Apple and third parties, *Macworld* magazine's best-sellers list, and new-product data sheets. When we introduce new marketing services, you'll read about them there as well as in *Apple Direct.*

The new Marketing Information & Services folder is located in the AppleLink Developer Support area on the Developer Services bulletin board.

•**AppleLink as a Marketing Tool.** AppleLink as a marketing tool? You bet! You can use AppleLink to distribute information, run promotions, or provide on-line support to customers at a fraction of the cost of using more traditional marketing tools.

To find out about the many ways to publish and manage infor- mation on AppleLink, check the "Publishing on AppleLink" bulletin board in the AppleLink Help Desk area, or call the AppleLink publisher's hotline at (408) 974-1300.

To see how one of your fellow developers has successfully used AppleLink to help market its products, see the article "Publishing Information on AppleLink," in the September 1991 issue of *Apple Direct.*

•**Comarketing Opportunities.** Each time Apple introduces a major new product, a host of comarketing opportunities becomes available to developers, both during and after the introduction. These activities let you leverage product-launch activities sponsored by Apple to get broader exposure for your own related products.

For instance, your product (which must be related to the product Apple is introducing) may be featured in *Apple Intro News,* an Apple publication for resellers and the Apple sales force. Watch *Apple Direct* for notice of when applications are being solicited for introductory comarketing opportunities.

•**The Marketing Track at the 1992 Worldwide Developers Conference.** For the first time, this year's Apple Worldwide Developers Conference (WWDC) offered a marketing track. It consisted of two consecutive days of sessions, specially tailored to the interests of marketing professionals. Topics included generating demand for products, alternative-channel strategies, creative solutions to software piracy, options for financial expansion, expanding into education markets, market opportunities in mobile computing, expanding into international markets, and others.

For information about how to order audio cassettes and video-tapes of selected sessions, see the article below entitled, "WWDC Tapes Available."

•**The Marketing Consultant Is In.** For the second year, the Marketing Consultant Is In program was a special feature of the WWDC. This program offered attendees the opportunity to receive one free hour of consultation with a marketing expert. All meetings were private, and the topics were selected by each individual developer. Participants walked away from their meetings with a host of new ideas for achieving their goals and with the name of a consultant they could call on in the future.

For more details about the range of services available and how to take advantage of them, look on the AppleLink network (using the path mentioned above) and peruse the business and marketing services packet enclosed in this month's mailing.

It Shipped!

Through the It Shipped! program, you can announce new and revised third-party products in *Apple Direct.* It Shipped! listings are also made available on the 3rd Party Connection AppleLink bulletin board. You can obtain an It Shipped! application by downloading it from the AppleLink network (AppleLink path—Developer Support:Developer Services:Apple Information Resources: Developer Program Information:It Shipped! Program). Or contact Todd Luchette at (408) 974-1241 (voice) or (408) 974-3770 (fax). Once you've completed the application, send it to Engineering Support, Apple Computer, Inc., 20525 Mariani Ave., M/S 42-ES, Cupertino, CA 95014; Attn: It Shipped! Program. Or send it via AppleLink to IT.SHIPPED.

Optionally, you may wish to send us a copy of your product to be placed in the Engineering Support Library, where it may be checked out by Apple's testing groups for compatibility testing, or by research and development employees for evaluation. If you would like your product(s) to be included in the Engineering Support Library, send them to the address above. The following products shipped in March 1992:

Publisher	Product (Version)
U.S.A.	
The AG Group, Inc.	Net Watchman, Jr. (1.2)
Alysis Software	More Disk Space (1.0)
Brooks/Cole Publishing Co.	Maple V—Student Edition (V (5))
Coconut Info	ClickAgent (1.01)
	McZip (2.0)
Compact Designs, Inc.	PROclaim! (1.0)
Computed Designs	Swing Shift (2.08)
Dartmouth College	MacPing (2.0.1)
E.E.S. Companies, Inc.	4D LabelTools II (1.0)
	HyperCard Labeltools II (1.0)
Jeddak	NetTime (1.3)
	SoftLibrarian (1.2)

Mirus Industries Corp.	MirusImage—Mac (2.3) MirusImage—MacUpgrade (2.3)
Neon Software	NetMinder Ethernet (2.1)
Nova Development Corp.	Kaboom! (1.0)
Object Factory	Object I/O 1.1 (1.0)
OBjectic Systems, Inc.	Fast Pitch 2.0 Lite (2.0b)
Orange Cherry Software	Talking Classic Tales (N/A) Talking Mac Classroom (N/A)
RealData, Inc.	Real Estate Connections (1.0)
ResNova Software	NovaLink Professional (2.0)
SoftWest	MacCAT (1.0)
Teton Data Systems	STAT!-Ref. (CD-ROM Medical Reference Libraries (3.1)
Vitale Software Group	theDirectory (1.0)

AUSTRALIA

Strategic Studies Group Inc.	Warlords (1.1)

CANADA

Alias Research, Inc.	Alias Sketch! (1.0.2)
epsi computer systems, inc.	epsiTalk (1.03)
ROBBERT Associates	whatIf? Decision Support Tools (A/UX) (2.0)

GERMANY

ATEC GmbH	MacProducer (1.0)

ITALY

MGE Communications	Imaginario CD-ROM (1.0)

WWDC Tapes Available

Whether you attended the 1992 Apple Worldwide Developers Conference (WWDC) or not, you can buy audio cassettes and videotapes for most of its 90-plus sessions. Get the official word on technologies such as AppleScript, O.C.E., RISC, and Apple’s new imaging architecture—as well as development tools, programming tips and techniques, networking and communications information, and marketing advice.

To obtain a list of sessions for which tapes are available and/or to place an order, contact Mobiltape via AppleLink (MOBILTAPE) or by phone at (805) 295-0504. Audio cassettes cost $8 each, and videotapes are $29.95 (plus an additional shipping charge for each).

MacApp Discussion Available on AppleLink

In the MacApp cover story in the February 1992 issue of *Apple Direct*, we inadvertently published the wrong AppleLink addresses to turn to for MacApp information and discussion. There are several AppleLink group addresses you can join to contribute to MacApp discussions and receive help.

By joining MACAPP2TECH$, you can participate in the MacApp 2 Technical Discussion, which covers MacApp 2.0 technical questions and issues. The discussion is archived for ref- erence (path—Developer Sup-port: DeveloperTalk: Macintosh Development Tool Discussion: MacApp Discussion: MacApp2tech$ Archives).

The MACAPP3TECH$ address enlists you in the MacApp 3 Technical Discussion. It is also archived (same path as above).

Send messages of general interest about MacApp to both of these group addresses. And to post timely information about MacApp, send a message to MACAPP.NEWS$.

To request more information or to join these group addresses, send a message to MACAPP. ADMIN (not to the group addresses described).

CD Highlights, May ’92

This column is your guide to the latest Developer Series CD, telling you what’s new and notable. To quickly access everything listed below, see the “What’s New on This CD?” folder (located at Dev. CD May ’92: What’s New on This CD?). Here you’ll find aliases to every new and updated package on the CD, including all those highlighted below.

To suggest something for upcoming Developer CDs, send us the survey located in the Start Here folder of the current CD; we’ll do our best to include suggested items on subsequent discs.

The May 1992 CD, “The Byte Stuff,” includes the following highlights:

***develop,* the Apple technical journal:** This quarter’s issue features several outstanding articles on topics such as “Apple Event Objects and You,” which describes how to support the Apple-event object model to guarantee the future scriptability of your applications. Learn how you can add custom PostScript functionality from “PostScript Enhancements for the LaserWriter Font Utility.” “Drawing in GWorlds for Speed and Versatility” tells how to improve graphics performance with custom drawing routines. Also, choosing the best colors to display an image with is tricky; “In Search of the Optimal Palette” explains how the Picture Utilities methods work and offers tips about developing your own methods.

develop also features a variety of interesting columns. You’ll find info about the top 10 printing crimes to avoid. And have you ever wondered how QuickDraw “magically” handles multiple screens transparently? “Graphical Truffles: Multiple Screens Revealed” gives you the inside scoop, along with tips on how to optimize drawing quality.

Apple Events & Scripting Development Kit: This kit includes five new and updated packages. Apple-event interfaces include the files that define the Apple Event Registry constants. You’ll also find the latest version of the Object Support Library 1.0 and sample applications that demonstrate how to develop programs that support Apple events. AE Tools offers useful things to help in developing Apple event-aware applications. AE User Terminology will help you understand user terminology.

Network Software Installer 1.2: This is the latest installer package for upgrading AppleTalk, Apple Ether-net, and TokenRing software to sup- port the AppleTalk Remote Access Protocol (ARAP).

Personal LW NTR 1.0: Here are new print drivers for the Personal LaserWriter NTR. This printer gives you outstanding value by combining high-performance PostScript printing with a level of flexibility you won’t find in other laser printers.

New Tech Notes: Five new Tech Notes are available on this CD, including one for people developing with OmegaSans (TN Note 314). We are also pleased to welcome Neil Day as the new product manager for Tech Notes and sample code. Neil’s goal is to improve the quality and quantity of the sample code on the CD; stay tuned for upcoming improvements. If you would like to suggest any changes, send an AppleLink message to Neil at DEVSUPPORT.

New Networking Course from Developer U

AppleTalk Course for Programmers

With the tremendous success of System 7.0 and other products that take advantage of network technology, demand is rapidly increasing for applications that provide networking capabilities. Developer University's newest self-paced course, "AppleTalk for Programmers," can help you learn how to implement AppleTalk protocols in new or existing applications.

"AppleTalk for Programmers" provides approximately 40 hours of technical training. The package, available from APDA as part number R0186LL/A for $295, includes a CD-ROM and a lab workbook. The CD-ROM contains a complete set of animated, narrated lessons and demonstrations with self-tests to help you measure your progress. An on-line glossary of networking terms, the ability to search any animation sequence for any narrated words, and the ability to set bookmarks all mean that you have ready-reference and navigation tools to enrich the learning experience.

The reference materials on the CD-ROM include an AppleTalk packet-format reference stack; two applications, Peek and the Phase II node identifier; and full Think C solution code for WindowTalk, a fully-functioning network application each student builds during programming labs. All course components are easy to access via an integrated and cohesive user interface. The lab workbook provides well-designed exercises to assist you in learning and building the course's target application.

The course is divided into two sections, AppleTalk Fundamentals and Programming With AppleTalk. AppleTalk Fundamentals provides in-depth coverage of AppleTalk concepts, such as how it fits into the OSI network model, the function and purpose of each AppleTalk protocol, when to use each in an application, addressing, message passing, sockets, name binding, zones, routing, ATP transactions, ASP sessions, ADSP connections, and more.

Labs based on network-diagnostic tools help reinforce these concepts. This section is well suited for managers who need to understand AppleTalk networking functions.

Programming with AppleTalk teaches you how to incorporate AppleTalk protocols into a C application. In this section's labs, you begin with an existing base of code (the simple graphics editor used in the Macintosh Programming Fundamentals self-paced course) and gradually add AppleTalk functionality to it. By the end of the course, you will have built a cooperative application that allows windows to register on the network and use ADSP to exchange graphic objects over an AppleTalk network in real-time.

No prior networking or programming knowledge is needed to benefit from the AppleTalk Fundamentals section. To complete the Programming with AppleTalk section, you should have a good working knowledge of the C programming language and sufficient knowledge of Macintosh toolbox calls to understand the base code used for the labs.

Since the course is self-paced, it provides you with the opportunity to gain an in-depth knowledge of AppleTalk at your own convenience, in your environment, without taking time away from your job. After you've completed the course, the course materials can serve as an ongoing AppleTalk reference.

You need System 6.0.5 or later, a Macintosh with at least 4 megabytes of memory, a hard drive, and a CD-ROM drive.

You will also need at least 10 megabytes of free disk space for lab projects. THINK C is required for completing the programming lab assignments, but is not required for viewing the lessons or completing the labs in the AppleTalk Fundamentals section of the course.

For information about how to contact APDA to order this course, see “Now available from Apple” in the News folder. And for more information about course schedules and dates, see the other related article in this folder..

Developer University Courses

Developer University offers a variety of self-paced and hands-on courses to help developers get the most from their development efforts. Self-paced courses can be ordered from APDA (for information about how to order, see “Now available from Apple” in this folder). For a more complete schedule of hands-on courses or for more information, contact Developer University at (408) 974-6215 or send an AppleLink message to DEVUNIV.

Below is a list of available courses and upcoming dates. Current courses for European developers can be found on AppleLink (path–Developer Support:Developer Services: Apple Information Resources: Developer Training:European Training.)

SELF-PACED COURSES

Macintosh User-Centered Design, *approximately 4 hours, $195*
This disk-based class comes with an introductory videotape plus laboratory exercises and solutions. The course presents the principles central to developing for the Macintosh’s unique human interface as well as guidelines for building ease of localization in to your code.
APDA No. M0271LL/A (NTSC version)
APDA No. M02712P/A (PAL version)

Macintosh Programming Fundamentals, *approximately 80 hours, $595*
A fully animated, voice-narrated CD-ROM provides an in-depth Macintosh programming course, enabling each student to develop a simple graphics-editor application, using Toolbox routines from the Memory Manager, Resource Manager, QuickDraw, Menu Manager, Window Manager, Control Manager, Toolbox, Event Manager, Print Manager, File Manager, and Standard File Package.
APDA No. M0997LL/B

Introduction to Object-Oriented Programming, *approximately 50 hours, $295*
An informative videotape introduces the concept and benefits of developing object-oriented applications, and a CD-ROM provides animated, voice-narrated

training in object-oriented programming fundamentals, including instantiation, message sending, inheritance, and polymorphism. Students apply these concepts while building an object-based Macintosh application in either C++ or Object Pascal.
APDA No. R0001LL/B (NTSC version)
APDA No. R0001ZP/B (PAL version)

AppleTalk for Programmers, *approximately 40 hours, $295*
This CD-ROM-based, multimedia presentation incorporates animation, graphics, and sound to provide an in-depth understanding of AppleTalk networking protocols, including taking an existing base of code and implementing AppleTalk calls to build a complete, fully-functioning network application.
APDA No. R0186LL/A

REFERENCE TOOLS

Macintosh Development Tools Advisor, *spend five minutes to five hours, $10*
The Tools Advisor is your electronic guide to assembling a Macintosh development environment. The Tools Advisor contains detailed descriptions of virtually all Macintosh development tools, articles on the latest Macintosh development issues, and technical advice from more than 50 experts (successful developers who have been through it before). A CD-ROM version will also include demos of the tools.
APDA No. RO171LL/A (CD- ROM version)
APDA No. R0124LL/A (Disk version)

HANDS-ON CLASSES

Macintosh Programming Fundamentals, *5 days, $1,375*
MPF is an intensive introduction to Macintosh application development using THINK C. In a lab setting, each student uses interactive modules to learn and implement fundamental Macintosh Toolbox Managers and calls to build a graphics application. Students also implement the four required System 7 events, initialize ROM managers, build multiple windows, master TextEdit,

create custom icons, print, and more. The instructor will provide expert development advice and one-on-one support, and will lead group learning activities. Students will receive a copy of the MPF self-paced course and will learn skills not taught in the MPF self-paced course.
Cupertino, CA: June 1–5, June 29–July 3, July 20–24, Aug. 24–28, Sept. 28–Oct. 2

Advanced System 7, *5 days, $1,375*
Learn advanced System 7 functionality beyond the basic compatibility topics. This class empowers developers to use the Apple events Object Model and Open Scripting Architecture to write new applications that communicate with each other and Apple event-aware scripting environments. Capitalizes on enhanced sound and graphics features of System 7.
Cupertino, CA: May 18–22, June 22–26, July 27–31, Sept. 14–18

Macintosh Programmers Workshop (MPW), *2 days, $750*
Learn to edit and build code with Apple’s own programming environment. Students customize MPW to their own working style, edit and rebuild an example program, write sample scripts, build and optimize a “make:” file, and create an MPW tool.
Cupertino, CA: May 18–19, Aug. 17–18

Programming with System 7, *4 days, $1,200*
Learn to implement the features of System 7, Apple’s newest system software. Students modify a System 6 application to take advantage of System 7 features, including sending messages between running applications, using the IAC package; implementing a standard set of Apple events commands; and exchanging information with other applications, using the Edition Manager and several File System Tools.
Cupertino, CA: June 8–11, Aug. 25–28
Dallas, TX, EDS: July 7–10*

Macintosh Debugging Strategies & Techniques, *2 days, $750*
To learn how to anticipate potential source-code bugs and create code that is less likely to crash, you employ “antibugging” techniques while developing your application. Gain experience in debugging crashes by applying source-

level debuggers, such as SourceBug and SADE, and machine-level debuggers such as MacsBug and TMON.
Cupertino, CA: June 22–23, Aug. 31–Sept. 1

Intermediate Object-Oriented Programming & Design in C++, *5 days, $1,375*
Master the principles of object-oriented programming and design. Learn step-by-step design methodology, and move to actual application of object-oriented programming techniques to build an application in C++, including sophisticated encapsulation, multiple and single inheritance, advanced accessibility rules, automatic initialization and freeing of objects, static and constant class members, and overloading.
Cupertino, CA: May 18–22, Sept. 14–18

Programming with MacApp 3.0, *5 days, $1,400*
Students apply object-oriented programming skills while they explore the use of MacApp 3.0, Apple's newest version of the industry-standard application framework for the Macintosh.

Students build a complete application with a rich System 7 feature set and place it inside a Macintosh user interface, using all the features of MacApp 3.0, including publish and subscribe, custom events, behaviors, dependencies, iterators, custom failure handling, and many more.

Cupertino, CA: June 1–5, June 29–July 3
Dallas, TX, EDS: July 27– 31*

*Contact the Developer University Registrar at (408) 974-6215 or by AppleLink at DEV.UNIV to obtain information on classes offered by Apple and EDS classes offered by an authorized 3rd Party Training Provider.

World-Ready System 7
Version-Number Problems Eased

by Gregg Williams, Apple Direct staff

It's every developer's nightmare: Apple just released a new version of system software for a new Macintosh computer—will my program run on it without crashing?

It's every user's nightmare: I've got to run this new program on a Macintosh SE—what version of system software do I need, and where can I get it?

(And let's not forget that many customers outside the U.S.—especially those in our Pacific markets—still don't have System 7 localized to their native language.)

The Macintosh has always led the field in ease of use and localization to languages around the world, but however good Macintosh software is today, we can always make it better. We've been working on a solution that solves many of the problems we've had with maintaining multiple versions of system software. And it'll certainly make things simpler for users and make the Macintosh much easier to use in languages other than English.

In a Nutshell. The solution is simple. You have one version of system software that is used for practically every Macintosh (specifically, every Macintosh Plus or later with a hard drive and a minimum of 2 megabytes of memory—or 4 megabytes if you are using Japanese, Chinese, or Korean). When and if it's needed, you will drop a special file—currently called a Hardware Support Mechanism (HSM)—into the System Folder to make the system software work on a specific machine.

In the future, all you will need to do to guarantee that you're running the correct, latest version of system software will be to make sure that you have the latest release of system software and the latest HSM (if any) for your computer.

(To be more accurate, I should point out that an HSM doesn't have to be a separate file—it can be present in a computer's System file or ROM. HSMs execute before extensions do; the main purpose of an HSM is to know how to boot one or more Macintosh models.)

Simplifying System Software. As the section above demonstrates, this way of doing things simplifies things for users. It also vastly simplifies your testing/compatibility process. When a new Macintosh model comes out, you need test only one configuration (the current release of system software plus the newest HSM) on the new machine; also, the addition of a new Macintosh model doesn't endanger the stability of your program on the Macintosh computers already in the field.

World-Ready Macintosh. One of our goals at Apple is to make the Macintosh world-ready—that is, equally easy to use in any (human) language and culture, anywhere in the world. Ultimately, we want users to be able to change the language used for menus and dialog boxes to whatever language they want.

A new system software approach called *release system software* is a big step in that direction. A single release of system software is general enough to be localized for any human language and any particular Macintosh through the addition of the right HSM.

System 7 Release 7.1 is the first fully featured version of System 7 for the Chinese, Japanese, and Korean markets. Its built-in support for TrueType fonts is exciting for many reasons, not the least of which is that this support drastically lowers the cost of a basic Macintosh system that uses outline fonts to print. (In the U.S. and in our most important Pacific market, Japan, we are already shipping a TrueType upgrade for Japanese.)

Currently, to print Japanese with a Macintosh, you need a PostScript printer with Japanese fonts—and a PostScript printer is very expensive in Japan. With System 7 Release 7.1 (or the currently shipping System 6.2 upgrade), any user can use the considerably less expensive TrueType printers (including Apple's low-cost StyleWriter) to put together a useful, affordable Macintosh system.

One major addition to the new release system software is built-in support for *double-byte languages.* (A double-byte language requires two bytes to store more than 256 characters.) System 7 Release 7.1 includes two extensions, World Script I and World Script II. They support single- and double-byte languages, respectively.

System 7 Release 7.1 includes two new Toolbox managers that help make the Macintosh world-ready. The Dictionary Manager provides a standard API (application programming interface) through which multiple programs can access shared user dictionaries. This is particularly important for input methods to be used in the Japanese, Chinese, and Korean markets.

The Text Services Manager allows you to manage, catalog, and access software objects that provide different ways of entering *glyphs* (indivisible units of written language) into the Macintosh. (This is important in languages such as Chinese and Japanese, in which text entry is very complex.)

What You Need to Know. At the very least, you need to make sure your programs run correctly with System 7 Release 7.1 (and later releases). The released versions of extensions such as AppleScript and the Open Collaboration Environment (O.C.E.) will require the current version of release system software. Later this year, new Macintosh models will ship with (and require) System 7 Release 7.1.

In addition to taking bottom-line compatibility into account, you should think about what it will take to make your programs great to use in whatever global markets make sense for them. You should consider localization issues and opportunities as you design, implement, and improve your programs. If possible, your programs should support whatever scripts users want to use. For more details, see the “Degrees of World-Readiness” article that follows.

PostScript. The release system-software approach is a better way to distribute system software because it simplifies the intersection of new system software with new Macintosh models. Apple has to work harder to make release system software a reality, but if the result makes life easier for users and developers, we’re glad to do it.

****

Degrees of World-Readiness

In order of increasing benefit (and effort), there are four levels of support for programs that can handle multiple scripts:

•*Multiscript-compatible:* The program provides minimal support for different scripts. You can handle this situation in several different ways. If your program cannot handle a certain script, you might have it give users an explanation via a dialog box and then exit. Another solution is to gray out (make inactive) the menu item that switches to that script.

- *Multiscript-aware:* The program lets users switch to any available script to enter data into a field. For some applications (databases, for example), it might be reasonable to assume that a field will not contain text in more than one script.
- *Modularly upgradable:* The program is compatible with multiple languages. At some future point, you may develop and market add-on products that allow the program to handle other scripts.
- *User-selectable-interface product:* The program allows users to change the script used for menus, dialog boxes, and other parts of the program's human interface.

The Transition to Scripting
AppleScript Due Later This Year

by Gregg Williams, Apple Direct Staff

There has never been a more compelling time to implement Apple events. For one thing, considerable information is available today to help you. The *Apple Event Registry* (which gives you common “vocabularies” of events that other Apple event-savvy programs will understand) has been frozen—meaning that Apple won’t change event formats or definitions.

And the most persuasive reason for supporting Apple events is just ahead: *scripting.*

Apple events and scripting are the yin and yang of a combined technology that will give users the ability to “glue” their applications together to create customized solutions from off-the-shelf products. Scripting brings out the full promise of Apple events: It gives users several ways of directly manipulating the power inside every scriptable application. Similarly, Apple events are the “engine” that powers scripting: Without useful, standardized suites of Apple events, scripts supply the “intelligence” to get something done but no “brawn” to actually do the work.

AppleScript is coming. We have a working version now, and we will ship it this year. Developers can request a seed copy of AppleScript by sending an AppleLink message to APPLESCRIPT. Please be sure to describe your application and how you intend to put AppleScript to use.

Being ready for AppleScript is important enough that Apple recently covered it, on the first day of this year’s Worldwide Developers Conference. The subject: telling developers what they need to do now to create useful, commercially successful products in the world of the script-capable Macintosh.

Suite Success. If you have developed a System 7 application, you’ve already had your first encounter with Apple events—that’s because every System 7 application must respond to the four events in the Required Suite: Open Application, Open Documents, Print Documents, and Quit Application.

The next important step for you to take is to implement the Apple Event Object Model and the Core Suite. The Core Suite includes such events as Get Data, Save, and Copy and such objects as Window, Word, and Item. (The Object Model forms the basis for a standard “language” that applications may use when sending or receiving Apple events.)

You don’t have to create a full-blown application that has its own scripting system to benefit from scripting. If you only implement the Core Suite, all users who can do scripting (with AppleScript or any third-party scripting system) will be interested in your program, because they can make it do something for them. Your program will be able to accept and carry out commands like “Open file ‘Monthly Report 4/92,’ print it, and exit the application” or “Copy item 1 in window ‘Market chart 1’ to the Clipboard, open file ‘1991 summary,’ and paste the Clipboard into it.”

Of course, if your program is a word processor, you’d like to support Apple events like “Copy paragraph 2 of window ‘Monthly Report 4/92.’” With the Core Suite implemented, you can copy *text* or *words* but not *paragraphs.* If, however, you go on to implement the Text Suite (which includes the *paragraph* object), your word processor can now offer its services to any program that wants to use it. Then, if a script wants to hand your word processor a piece of text and have it italicize any paragraphs that are in all caps, your program will automatically be able to assist in meeting such a request. (The script will specify how to get the task done with Core and Text Suite Apple events—all your program has to do is know how to execute the Apple events in those two suites.)

(Interestingly enough, the Text Suite defines no new events, just new objects. In other words, the events defined in the Core Suite are in fact *core.* When you define new objects, the existing events—think of them as verbs—allow you to do new, useful things in an entirely new arena.)

So, once you’ve implemented the Core Suite in your program, you should implement whatever defined suites make sense for it. (See Table 1, below, for a list of all the currently defined Apple-event suites. Apple is working with developers to create additional suites where they are needed.) Once you’ve done that, your program will have become fully scriptable. Users will be very interested in it, because it will make their scripting system very useful.

For Your Information. Here are the resources you need in order to learn more about Apple events and scripting:

•“The Byte Stuff” (May 1992 Developer CD): If you are an Apple Associate or Partner or if you subscribe to the APDA Technical Information Mailing, you should have received this CD-ROM in the same mailing as this issue of *Apple Direct.* The “Apple Events/AppleScript” folder on this CD-ROM includes documentation and sample code about Apple events and the Object Model. The Apple Event Registry, Apple Event and Scripting Developers’ Kit, and other resources are also on this CD-ROM, with the pathname Dev.CD May 92:Tools & Apps (Moof!): OS/Toolbox:Apple Events.

- AppleLink: AppleLink contains an assortment of items related to Apple events. Use the BB Pathfinder program (in the AppleLink Information folder), and search for items by using the phrase “apple and events”.
- Developer University: Developer University will teach a five-day course called “Advanced System 7: Apple Events and the Open Scripting Architecture.” For more information, see “Developer University Courses,” in this issue.
- “Apple Event Objects and You:” This article is in the newest issue, issue 10, of *develop* magazine. An electronic copy of it is on “The Byte Stuff” Developer CD (see above). This is an in-depth code-level article about different ways to implement the Object Model; the code in the article is written in C. Unless you’re already an Apple-events expert, you absolutely must read this article.
- APDA sells two products in which you may be interested: *Apple Event Registry: Standard Suites* (part no. R0130LL/A, $85) and Apple Event Education Suite (part no. R0218LL/A, $25). For information on ordering from APDA, check the bottom of the “Now available from Apple” located in the News folder.

Table 1: *Apple Event Registry* Suites

Name	**Coverage**
Required Suite	Needed to be a System 7 application
Core Suite	Covers events and objects useful to almost every application
Text Suite	Adds text-related object
QuickDraw Graphics	Adds objects for most QuickDraw operations—arcs, grouped graphics, pixel maps, etc.

QuickDraw Graphics Adds scaling, rotation, translating,
Supplemental Suite and coloring to basic QuickDraw objects

Table Suite Adds cell, row, column, and table objects,
for manipulation of tabular material

O.C.E. Update Includes Directory and Store-and-Forward Features

by Gregg Williams, Apple Direct staff

If you haven't gotten interested in Apple's Open Collaboration Environment (O.C.E.), it's because you don't know enough about it. You heard about it in the April 1992 *Apple Direct* ("Blueprint for the Macintosh"), but that was just the beginning. This month, Apple talked a lot more about it—an entire day at this year's Apple Worldwide Developers Conference, which was held May 11 to 15 in San Jose, California.

Apple has worked on the O.C.E. for several years, in conjunction with key developers in fields as varied as networking, database services, and data security, in-cluding established Macintosh developers as well as companies for which the Macintosh is a new market.

The Open Collaboration Environment is too complex to describe fully here, but the sections below highlight some of the most important things about the O.C.E. that were covered at the Worldwide Developers Conference.

O.C.E. Hands You the Building Blocks

The Open Collaboration Environment extends the Macintosh platform to provide mail, messaging, directory, and digital-signature services to all Macintosh programs, using open, standard APIs (application programming interfaces) that encourage easy interface to existing and future programs, services, and data. Later in this article, I'll summarize the major software components to give you a more concrete idea of how the pieces of the O.C.E. fit together.

Store-and-Forward Messaging

System 7's interapplication communication (IAC) allows any two Macintosh programs connected to the same network to communicate with each other—but both of them must be executing at the same time. The O.C.E.'s Interprogram Messaging Manager gives all Macintosh programs the ability to send messages to any other Macintosh program, but with fewer restrictions. The sending program need only know the “address” (whatever form that might take) of the recipient, and the receiving program need only be reachable (through a network, gateway, phone/modem connection, or other form of connection).

Thus, the Interprogram Messaging Manager gives you a standard, easy-to-use way to do store-and-forward messaging, something that Macintosh programmers have wanted for a long time. Now you can concentrate on building something interesting on top of it.

DIRECTORIES FOR EVERYBODY

Directory is a deceptively simple word that most people associate with telephone books—when you want to telephone people, you look their name up in the telephone book (directory) and dial the number, right? But directories can be really much more than that—you can use them to hold *any* structured information.

The Open Collaboration Environment provides several human interfaces for viewing directories. From the desktop, users can open and view O.C.E. directories inside Finder windows. This interface also allows them to drag directory entities directly from the Finder into mail-aware applications. The O.C.E. also provides a Standard-File-like interface that you can add to your application; this gives users the ability to browse directories from within your application.

Since the contents of a directory are entirely up to you, your directory can contain mail addresses, telephone numbers, graphics, QuickTime movies, audio, or whatever else sounds interesting to you.

COMPONENTS OF THE O.C.E.

The Open Collaboration Environment is not really a single technology but rather a handful of technologies, each of which hands developers such as you a new set of capabilities, along with new product and market opportunities. Here are some brief overviews that will begin to “flesh out” the main components of the O.C.E.:

•*The Collaboration Packages:* These three packages give you the routines you'll need to provide a user interface to O.C.E. directories and the Interprogram Messaging Manager.

The *Standard Mail Package* provides a more sophisticated way of sending mail (with attachments) to another user's In tray, which is a single window that contains all the mail—in various forms—a user receives through the O.C.E. This package lets you add a region called a *mailer* to any document window; the mailer specifies various mailing options and can be used to turn any of your program's documents into electronic mail.

The *Standard Directory Package* provides a Standard-File-like directory browser. With the routines supplied by this package, you can easily add to your program the ability to browse any O.C.E. directory.

•*The Interprogram Messaging Manager:* Within the Open Collaboration Environment, a *message* is a collection of data grouped as a *message header* (containing addressing, priority, and other control information) and a *message body* (containing one or more blocks of arbitrary data). The Interprogram Messaging (or IPM) Manager is responsible for creating, sending, receiving, displaying, and manipulating messages in various ways. It exchanges messages between message queues, which are located in a local queue, a queue on another computer, or a Store and Forward server.

•*The Directory Manager:* A *directory* is some arrangement of information in a hierarchical structure similar to that of the Macintosh file system. It can be a personal directory that resides on your Macintosh, a directory that resides on an O.C.E. directory server, or a foreign database that is accessed through an O.C.E. directory gateway. Even though these directories work in different ways, users manipulate them through one consistent user interface. More importantly, you as a developer can manipulate all of them by using the same set of routines.

•*Directories Extension templates:* If you are providing some service or information through the Open Collaboration Environment, you will design one or more Directories Extension templates. These templates tell the O.C.E. how to manipulate and then display information coming from some directory. Without such templates, all directories would have to fit a predetermined format that the O.C.E. would dictate. With these templates, the O.C.E. can display data from existing databases and services, with the template designer controlling how the data is interpreted and displayed.

•*The Digital Signature Toolbox:* This set of routines allows you to apply a *digital signature* to any set of data. A digital signature does two things. First, it guarantees the authenticity of the person “signing” the document. Second, it embeds a code that your program can check to verify that the data that was “signed” has not been changed. You can apply a digital signature to anything you like: a document, a field, a memory address—whatever data you want to authenticate.

It’s also interesting to note that the digital signature and its associated data don’t have to travel together. For example, person A can leave a document on a file server. By “signing” the document and sending the signature to persons B, C, and D, the O.C.E. can verify for these people that the document on the server is unchanged from the moment it was signed and that person A is in fact the person who signed and sent it. We feel that the Digital Signature Toolbox opens the doors to entirely new uses of Macintosh computers.

More to Come…. We’re very excited about the Open Collaboration Environment and hope that you are, too. Look to future issues of *Apple Direct* for more details.

Now available from Apple

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

If you're interested in the latest version numbers of all Apple system-software products, check "Latest Rev" in the Information Resource folder on the current Developer CD. Latest Rev also tells you where to obtain those system-software products. In addition, the "CD Highlights" section of this issue (see the News folder) tells you which new system-software releases appear on the current CD.

Apple Products

AppleTalk for Programmers: Self-Paced Training Course
R0186LL/A
$295

DAL Developer's Toolkit Bundle v.1.3.5
B0582LL/B
$695

DAL Server for IBM MVS/VTAM Bundle v.1.3.5
B0581LL/B
$30,000 (contact Apple Software Licensing)

International DAL Server for IBM MVS/VTAM Bundle v.1.3.5
B0668LL/A
$30,000

DAL Server for MVS/TSO Bundle v.1.3.5
B0579LL/B
$20,000

International DAL Server for IBM MVS/TSO Bundle v.1.3.5
B0667LL/B
$20,000 (contact Apple Software Licensing)

DAL Server for VM/CMS Bundle v.1.3.5
B0578LL/B
$15,000

International DAL Server for VM/CMS Bundle v.1.3.5
B0669LL/B
$15,000 (contact Apple Software Licensing)

Introduction to Object-Oriented Programming v.1.0.1 (NTSC)
R0001LL/B
$295

Introduction to Object-Oriented Programming v.1.0.1 (PAL)
R0001ZP/B
$295

Introduction to Object-Oriented Programming v.1.0.1 Update
R0132LL/A
$79

Local Area Transport Connection Tool v.1.1
M0800LL/B
$50

Localization for Japan
R0250LL/A
$20

VT320 Terminal Emulation Tool v.1.0.2
M8081LL/B
$50

Third-Party Products

AppMaker v. 1.5
T0322LL/D
$299

SPECIAL LIMITED-TIME OFFER*

E.T.O. Complete New Subscriber Package – $995
Order the E.T.O. Complete New Subscriber Package before June 15, 1992 and save $300 off the $1295 regular price. Pay only $995 for a 4-issue subscription to the latest software.

E.T.O. is a CD-ROM subscription that includes the latest versions of MPW, MacApp, MPW C, MPW Object Pascal, MPW C++, MPW Assembler, ResEdit, and MacsBug–plus documentation, release notes, and manuals.

APDA Order No. M0895LL/B

To place an APDA order from within the U.S., contact APDA at (800) 282-2732; in Canada, call (800) 637-0029. For those who need to call the U.S. APDA office from abroad, the number is (408) 562-3910. You may also send an AppleLink message to: APDA. If you’re outside the U.S., you may prefer to work with your local APDA contact. For a list of non-U.S. APDA contacts, see the “International APDA Programs” page in the *APDA Tools Catalog.*

* Offer applies only to the Complete New Subscriber Package, new subscribers only. You may order an unlimited number of subscriptions. No credits allowed with this offer for previously purchased individual software components of E.T.O. Price does not include applicable tax and shipping charges. A completed and signed APDA Customer Agreement form must be on file with APDA to order this product. Forms are available in the current *APDA Tools*

Catalog and on AppleLink. Promotion not necessarily available through other International APDA Programs.

Apple Direct Bids Goodbye to Tog

You won’t find the usual Tog wit and wisdom in this folder, as you have for quite a while. We at *Apple Direct* have had to (very) sadly bid goodbye to Bruce Tognazzini, who recently resigned from Apple Computer after spending more than a decade here.

However, because human-interface issues are critical to Apple and its developers, we will continue writing about them in *Apple Direct.* Our “new era” in human-interface coverage will begin soon; for more information, stay tuned to future issues.

Free Protective PowerBook Battery Case Offered

Apple is offering free protective cases for the replaceable battery packs used in the Macintosh PowerBook 100, 140, and 170 notebook computers.

When the battery pack is out of the computer, its electric contact terminals can be inadvertently shorted by metal objects such as keys and paper clips.

This shorting can lead to fires or burns. The free protective case electrically insulates the contacts, thus eliminating the possibility of shorting.

To receive a protective case or to ask for more information, developers in the U.S. should call (800) 377-4127.

Those outside of the U.S. should contact their local computer-service providers. After May 1992, all Apple PowerBook rechargeable batteries will be sold with the case included.

Apple is working closely with the U.S. Consumer Product Safety Commission and is cooperating with the commission staff to investigate potential safety risks.

Hot Product Specials Each Month From APDA

Each month starting in May, APDA will offer a special promotion for one of its key products to *Apple Direct* readers. The “Hot Product of the Month” will appear in the “Now available from Apple” column of each issue. Also, this month we are offering an additional special limited-time offer.

We start our new promotion with a very special offer: AppleTalk for Programmers, a newly-released self-paced training product from Developer University. (For a detailed description of the course, see "New Networking Course from Developer U,")

APDA Hot Product May ’92

Order by June 30, 1992, and get a *free* Macintosh Development Tools Advisor CD-ROM with either product.

AppleTalk for Programmers Self-Paced Training Product
APDA No. B0880LL/A
$295

AppleTalk for Programmers Special Bundle
Includes AppleTalk for Programmers Self-Paced Training, Macintosh Fundamentals Self-Paced Training, and the Macintosh Development Tools Advisor CD-ROM.
APDA No. B0881LL/A
Special Bundle price: $595
Save $310

Offer applies only to the products specified above. Price does not include applicable tax and shipping charges. Offer good until June 30, 1992.

GetNextEvent

The “**” indicates the trade shows/events at which Apple Computer, Inc., is scheduled to exhibit as of press time. This list may be incomplete. If you have information about a show you want listed here, write to Apple Direct Event Calendar, 20525 Mariani Avenue, Mail Stop 75-2B, Cupertino, CA 95014. For further information, check the Events folder on AppleLink (path—Developer Support:Developer Services:AppleInformation Resources: Developer Events).

May 18 through 22
****INTEROP**
Washington, DC
Contact: Debbie Moessinger
(415) 941-3399

May 28 through 29
QuickTime—The Conference & Film Festival
San Francisco, CA
Contact: Macworld Communications
(415) 267-1745
Fax: (415) 422-0766

May 31 through June 3
****SPA European Conference**
Cannes, France
Contact: SPA, (202) 452-1600

June 1 through 5
****Object Expo**
New York, NY
Contact: G.G. Schafran
(212) 274-0640

June 8 through 11

****AEC Systems,** Dallas, TX
Contact: Kelly Baxter
AEC Systems
(215) 444-9690

June 8 through 12
DAC—Design Automation
Anaheim, CA
Contact: Marie Pistilli
(303) 530-4562

June 10 through 13
****Logic Zurich, Apple Expo**
Contact: Kathrin Mäder
Phone: Switzerland 1 832 81 11
Fax: Switzerland 1 830 63 06

June 15 through 17
NECC—Nat'l. Educ. Computing Conference,
Dallas, TX
Contact: Sharon Finke
(503) 346-3537

June 23 through 25
PC Expo, New York, NY
Contact: H.A. Bruno, Inc.
(201) 569-8542

July 14 through 17
Object Expo—London
London, England
Contact: G.G. Schafran
(212) 274-0640

July 14 through 16
****Mactivity '92,** The Macintosh Connectivity Conf.

Santa Clara, CA
Contact: Paul Kent,
Winehouse Computer Company
AppleLink: MACTIVITY
(408) 354-2500

July 26 through 31
SIGGRAPH, Chicago, IL
Contact: Barbara Voss
(212) 752-0911

August 4 through 7
****MacWorld Expo**
Boston, MA
Contact: Mitch Hall & Assoc. (617) 361-8000

September 8 through 12
****Orbit**
Basel, Switzerland
Contact: Kathrin Mäder
Phone: Switzerland 1 832 81 11
Fax: Switzerland 1 830 63 06

September 23 through 30
Bureau 92
Brussels, Belgium
Contact: BKK
Phone: 02 7627183
Fax: 02 7629434

September 29 through October 1
CAD/CAM Show
Kortrijk, Belgium
Contact: De Hallen
Phone: 056 204000
Fax: 056 217930

Sept. 30 through October 3
****SPA**

Washington, D.C.
Contact: SPA
(202) 452-1600

October 19 through 21
Nat'l. AEC Expo– Architecture, Engineering & Construction
San Francisco, CA
Contact: Expoconsul Int'l.
(609) 987-9400

October 21 through 23
****Winworld, Zurich Window Exhibition**
Contact: Kathrin Mäder
Phone Switzerland 1 832 81 11
Fax: Switzerland 1 830 63 06

Crossing the Chasm

Moving from Early Success to Mainstream Market Leadership

by Geoffrey Moore,
Regis McKenna, Inc.

It was a Wednesday afternoon, and I was hard at work. My feet were up on my desk, a cup of coffee resting comfortably on my stomach, my eyes focused somewhere between my desk and the door. I was just about to have a flash of marketing insight—you get to where you can feel one coming, sort of like a migraine, only different—when the phone rang.

"Hello, Geoff?"

Drat. It was him again. The same guy who'd called up a couple of months ago to find out how to position his product with the elevator test. That's what you get, I thought, for answering your own phone.

"Say, Geoff, I understand you got a new book out, something called *Crossing the Chasm,*" he said, pronouncing the *ch* as in *ch*url instead of as in *ch*arisma."I think maybe we are in the chasm, and I can't wait around to read the book. Just give me a few pointers—you know, like last time."

From past experience, I knew I wasn't going to shake this guy, so I reluctantly put my feet back on the floor.

DEFINING THE CHASM

The basic idea, I explained, is that high-tech markets do not develop in a continuous fashion. Specifically, when you factor in the time element, they do not follow the bell-shaped curve predicted by the Traditional Technology Adoption Life Cycle (see figure 1 below).

Instead, they develop in the phases (see figure 2 below) that follow:

Phase 1: The Early Market. The bulk of customers in this phase are technology enthusiasts and visionaries, the two customer types that populate the very front end of the Technology Adoption Life Cycle. The former get involved with high-tech products early on simply because they love technology and like to play with the latest and greatest toys. The latter, the visionaries, get involved early because they want to revolutionize some aspect of their business to gain a dramatic competitive advantage. Both these groups, in other words, see an advantage to being the first to adopt a new product.

Phase 2: The Chasm. This is a "nonmarket" phase, during which there are really no natural customers for your high-tech product. It is as if the front part of the life cycle has broken off from the rest of the curve; this phase represents the gulf separating the two.

During this time, the product has by now been on the market long enough for technology enthusiasts and visionaries to have heard about it and, if they are interested, to have bought it. But the product is not yet established as a "safe" buy; it has not, in other words, secured a market-leadership position in any particular segment of the market. From the point of view of a pragmatist or a conservative, it is simply too soon to take a chance on the purchase.

As a result, during the chasm phase, sales dip precipitously—often just at the point where you have promised your investors a surge in revenue.

Phase 3: The Mainstream Market. This is the phase of market development in which all high-tech wealth is generated. It is represented by the bell part of the life-cycle curve. The primary customers now driving sales are the pragmatists. Later on, if the company stays in the market and continues to support its product, it will sell more and more to conservatives.

Pragmatists use technology products to improve productivity in a nondisruptive way. They prefer to go with the leading product, because it is one they will be able to live with for a while and because market-leading products tend to attract third-party support, thereby augmenting the value of the product at no cost to the customer.

Winning acceptance among pragmatists is what causes the dramatic acceleration in revenues that characterizes the most successful high-tech products.

The conservatives who follow in the adoption cycle are looking to get by with the safest, cheap- est technology they can find. Fundamentally, they are afraid of technology purchases—afraid they will choose the wrong thing, afraid they won't be able to make it work right, afraid they will break it or just make themselves look stupid trying to use it; so they will choose only a fully proven, absolutely bulletproof solution—and even then will worry about it.

Finally, because they delay their purchases until the back end of the product's life cycle, conservatives also expect—and get—low prices.

During the mainstream phase, the product has been accepted as an established part of the high-tech landscape and usually is the leader in at least one market segment. (Products that are not leaders—that are simply third, fourth, or fifth in the market—typically get "shaken out" over time; they are eventually acquired by the market leader or go out of business.)

ARE YOU IN THE CHASM?

"So how can I tell if we are in the chasm?" my caller asked.

I asked him to tell me about his product.

"How about if I give you my elevator test positioning statement?" [Ed. note: For more information about the "elevator test," see "So What's Your New Product All About?" in the February 1992 issue of *Apple Direct*.]

The guy knew how to go after my soft spot.

"Here goes: *For* market re-searchers and marketing executives *who* use census data to target prospective customers, *Infomagic* is a CD-ROM database and data-retrieval tool for the Macintosh that puts this information at your fingertips. Unlike on-line census information databases, *Infomagic* is usable by anyone in the marketing organization and does not generate expensive phone-access charges."

(For budding entrepreneurs in the information business, Infomagic is not the real product name, so if it passes the trademark test, it's yours.)

"How was that?" he asked.

Pretty good, I had to admit.

"So am I in the chasm?"

Well, I asked, how are sales? Not as good as he was hoping. Why not? "Well," he explained, "at first they took off like a rocket. We even had back orders for a while. But lately they are going nowhere."

How come? All kinds of reasons, he complained. "First of all, not that many people have CD-ROM drives for their Macintosh computers yet. And second, a lot of people think that market analysis is just for specialists. Furthermore, the distribution channel doesn't want to carry the product because it is too specialized. But when we give it to VARs, they try to make the distribution too complicated. For a while we were successful with a money-back free trial, but now even the response to that offer has dried up," he said.

"It definitely sounds as if you are in the chasm," I said. "That free-trial offer is a great tactic when your product is in the early market phase, but mainstream pragmatists aren't worried as much about the cost of purchase as the cost of ownership. They simply don't like to be the first people to experiment with new things."

"Help me out here," he said. "Who isn't in the chasm?"

Who's In, Who's Out. As with everything else, I told him, it is often a matter of opinion, but let me give you some of my picks for 1992:

Early Market:

- Any QuickTime product
- Any pen-based product
- Any kind of collaborative application

Chasm:

- Any CD-ROM-based product
- Any user programming product
- Any kind of scheduling or calendaring groupware

Mainstream (just arrived):

- Fax-based products
- Remote E-mail products
- Any product especially suited to the PowerBook

"You just make this stuff up or what? What's the idea here?"

Articulate chap, I thought. Actually, I explained, the key difference that determines whether a product has made it into the mainstream is the existence of the "whole product."

THE "WHOLE PRODUCT" CONCEPT

I continued my explanation: The whole product is the complete solution, the entire set of products and services needed by users to achieve the value proposition promised by the product you are selling. Pragmatists evaluate products based on whether the whole product is a proven solution and one that is readily available. Until it is, they won't buy.

In the case of new entrants to the mainstream—fax, remote E-mail, PowerBook software—all have been recently enabled by the emergence of the PowerBook as a suitable Macintosh portable. This was the last piece of the solution that was necessary, since other PC laptops had already established the need for, and had driven the adoption of, such things as support for data and fax modems and the installation of remotely accessible E-mail packages.

The reason why QuickTime and pen-based products are still in the early market phase is that the whole product is nowhere near ready for "prime time." Only people who have either the skill (technology enthusiasts) or the drive (visionaries) to make these products work—no matter what—will be customers in the near term. However, because these products represent the latest and greatest technology, they will attract some early adopters.

The whole product is not just made of hardware and software. It can also incorporate something as abstract as "accepted behavior." The reason why collaborative applications, for example, are still early-market products has little to do with hardware or software. Everything you need to implement it is available off-the-shelf—what is missing is the necessary paradigm shift.

"The what?"

OK. A change in behavior. People are not used to collaborating via computers. It's not that it can't be done but more that it "isn't done." As a result, perfectly good solutions aren't really solutions at all, because they don't reflect the way people actually work together. Until some significant evangelism creates this change in behavior (that is, in the way people collaborate) and users can celebrate the competitive gains achieved by it, these products will stay in the early market position.

"But what about the chasm itself?"

Products fall into the chasm when they lose the sizzle of being a brand-new technology, but still lack either the whole-product infrastructure or the paradigm shift needed for customers to readily achieve desired benefits. CD-ROM titles fit into this category simply because CD-ROM drives are not yet ubiquitous. It's a little bit like the old VCR versus VCR-titles situation: There wasn't enough of either one to generate adequate investment in the other.

User programming suffers from a paradigm problem—in my opinion, one that may become permanent. The problem is not only that languages are hard to learn but also that most end users lack the systems-analysis skills to design what they want, regardless of how easy it might become to actually program it.

To climb out of the chasm, group calendaring will also have to overcome a paradigm problem, this one resting on the inability to get any group of people with whose schedules you want to coordinate to consistently obey the protocol of keeping their calendars on line and up-to-date.

HOW TO CROSS THE CHASM

“All right. I think we’re in the chasm. Not only do we rely on CD-ROM drives, but we also seem to be introducing a new paradigm. So what do we do now?”

Crossing the chasm, I replied, is like invading Normandy. (I figured he probably didn’t know much about World War II but must have at least played Axis and Allies.) Your ultimate goal is to liberate all of Europe, but your immediate goal is just to gain a beachhead on the continent. So you direct all your forces solely to that end.

“I’ve never been to Normandy.”

My caller was nothing if not literal.

The key to getting out of the chasm, I began again, is to target one specific segment in the mainstream and focus all your efforts on accelerating the formation of a whole product and affecting a paradigm shift just for that one segment. The idea is to achieve a visible market-leadership position within that segment—proven acceptance by at least one group of pragmatists—and then use that beachhead as a basis for moving into related “adjacent” segments of pragmatist customers.

“Wait a minute, wait a minute. Our product is truly horizontal. It works across many different vertical segments. Why should we narrow it to just one?”

The problem, I explained, is that although all the segments may share common needs, each also has its own unique ones. In the case of Infomagic, this might be special data sets, or special metrics for different industries, or some industry-specific database that most customers merge with their other data, or simply a vocabulary unique to that industry. Accelerating the whole product typically involves doing some custom work to ensure that these pieces all fit together correctly. Most developers simply do not have enough resources to accelerate the formation of multiple whole products at the same time, so they should choose just one.

"Well, it's true that a lot of the fast-food people wanted us to interface to a Geographic Information System mapping program to help with their site-location planning, and the recording industry was looking for additional data about dependents under the age of 18, and the snack-food people needed to correlate data with their scanner-output records. I see what you mean. But how do I pick which segment to focus on?"

SO MANY SEGMENTS, SO LITTLE TIME

This is indeed, I explained, a tough problem. If you do not pick a segment quickly, you can easily become permanently stuck in the chasm. But since all segments at this point in the product's life cycle look pretty much alike and since there are so many of them, how to proceed is often confusing.

The best approach is to begin defining possible target segments through a process called application-scenario generation—each scenario is built around a before/ after-your-product-was-adopted "day in the life" for a potential target customer. These scenarios are then ranked against each other, with two key criteria:

•How compelling is this customer's reason to buy your product?

• How feasible is it for you to ensure the availability of the whole product needed to fulfill this target customer's compelling reason to buy?

The rating and ranking process is conducted in-house, leveraging whatever expertise you have readily available from your own employees or from informal advisers to your company.

"So, what I have now is a bunch of made-up stories about what might be the possible applications of my product?"

Basically, yes, I said, reminding myself once again that marketing is, in essence, just another literary form. Each of these "stories" functions as a kind of placeholder for a possible target market. Segments that score highly, based on the previous two criteria, warrant further attention and should be assessed with a variety of additional criteria, including:

•Where we need the help of a partner or ally to provide some piece of the whole product, do we already have a relationship with this company or must we start building one from scratch?

•Are target customers likely to want to buy from our existing distribution channels?

• Is our current pricing consistent with the value proposition that motivates these particular customers, and with their estimated purchasing power?

•Are customers already well- served by a competitor, or are their needs largely unmet?

Again, prioritize the scenarios, based on what you believe you already know about the market.

Finally, ask one last question: If we win over these target customers—if we dominate the market segment that they, as a group, constitute—will that provide an entrée to additional market segments (the way, for example, that winning over graphic artists to desktop publishing eventually provided an entrée into sales and marketing departments)?

The one or two scenarios that fare best in this entire rating and ranking process are the ones you should nominate as target segments.

"Look, I'm happy that I am not spending a lot of money on you high-priced consultants, but don't I run the risk of bathing in my own bathwater?"

Now there's an appetizing image, I thought.

"Yes, you do, so at this point you'll want to bring to bear whatever market-research resources you can to get an external perspective on the problem."

At a minimum, I explained, you need to phone a representative sample of people who meet the criteria for your target customers, and simply interview them to see if you've understood their needs and issues. Typically, you also call other third parties who have market knowledge—distributors, consultants, industry analysts, hardware evangelists, other software developers, and the like. The goal of these conversations is to validate or correct your marketing assumptions so that at the end
of the process, you have a confirmed strategy.

BUILD THE BRIDGE

"What I have at this point, I guess, is a top-priority target market segment. But what do I do now?"

I looked at my watch. How much longer was I supposed to humor this guy? I covered the mouthpiece, made some rumbling noises, and let him know that I had just been called into an important meeting.

So real quickly, I said, what you have to do now is communicate your whole-product strategy to the key players in your target market—the customers, of course, but also every other constituency that will play a role in either delivering the whole product or establishing your reputation as the leader of this particular market-development effort. The communication is a two-way dialogue, because every player in your scenario has to win something from the endeavor—and you need to confirm both that you understand what constitutes that win and that the others see how your effort supports it.

This win/win approach to all the constituencies is what allows the market segment to develop at an accelerated pace, catapulting your product to early acceptance and your company to the status of perceived market leader.

I was doing my best to end on a high and glorious note: "Once you've won one segment, it becomes a lot easier to leverage your reputation into adjacent markets, transforming what begins as a highly focused marketing effort into what eventually becomes broad-sector leadership."

"OK," he said. "Say, this has been good advice."

"You think I should write a book?"

"Oh, yeah. Sure. Send me a copy."

And with that he hung up.

Geoffrey Moore is a partner at Regis McKenna, Inc., a Palo Alto, California firm that specializes in marketing strategy development and market-relations services for technology-based clients. He is the author of the book Crossing the Chasm: Marketing and Selling Technology Products to Mainstream Customers.

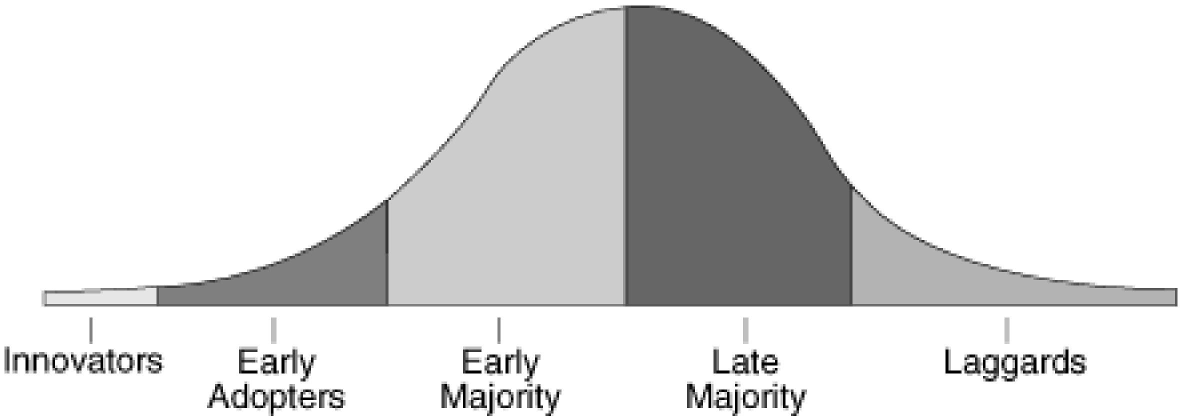


Figure 1: Traditional Adoption Curve

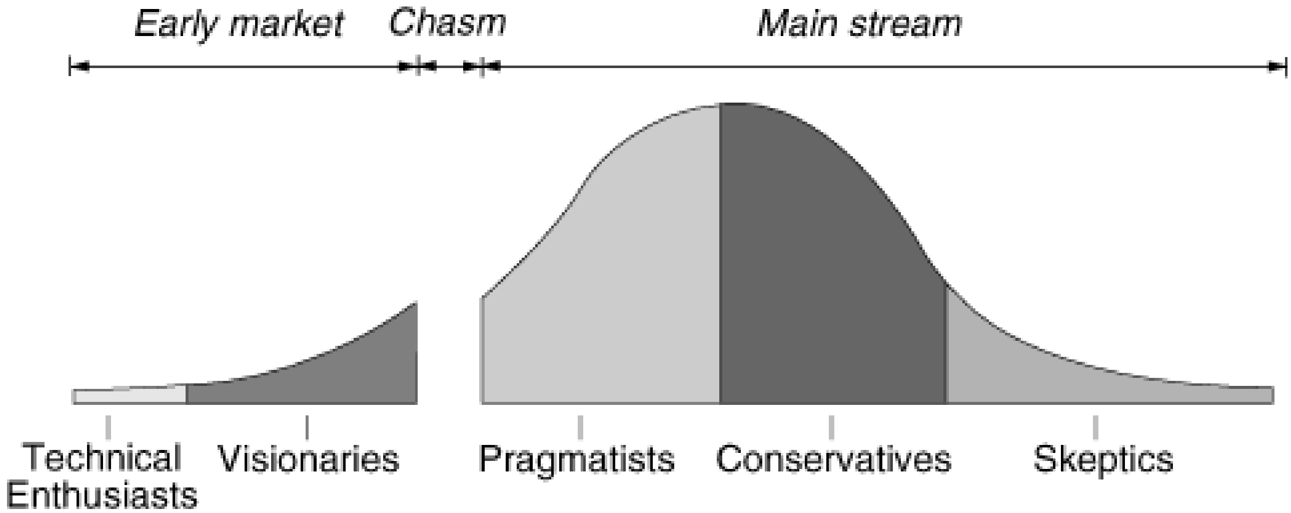


Figure 2: Revised Technology Adoption Curve

How Direct Mail Saved Our Company
Overcoming Fear of Channel Conflict

by Dave Johnson,
Working Software

In 1989, Working Software was broke, few of us were getting paid, we owed a lot of money to other people, and we couldn't seem to sell anything. (Does this remind you of any startups you know?)

Within the following two years, we'd turned things around so dramatically that in the final quarter of 1991 we mailed more than one million pieces of direct mail, sold thousands of units of product, and pretty much broke even—despite the recession being in full swing.

This article is about how direct mail saved our company—what our challenges were and what we learned in overcoming the fear of channel conflict.

FROM GROWING REVENUES TO HARD TIMES...

Working Software was formed in 1985; we first sold Spellswell at the January 1986 Macworld Expo. By mid-1988 we were selling an entire line of products through most major distributors and retail chains.

When our sales took a major downturn in late 1988, we were perplexed. We were receiving a lot of great press coverage, and our products were getting very good reviews. Existing customers told us that they liked the products and the company.

But we weren't getting orders. We did not realize that the Mac software marketplace had changed, and we did not correctly understand who our customer was.

The situation was not really that we couldn't sell any software; it was that we were trying to sell software to retailers—and the rules for selling to that channel had changed.

Only later would we learn that our true customer is the Macintosh user, not the retailer.

Does the following tale sound familiar? Smaller stores couldn’t afford to keep a large number of titles on their shelves, which was bad news for a small company such as Working Software. Most large retailers wanted a substantial amount of cash up front before they would carry our products. They also preferred to carry higher-priced products that yielded a higher profit per unit sold. And they wanted “spiffs”—fees the salespeople receive for recommending one product over a different, possibly superior product.

Furthermore, catalogs wanted money to list us. Magazines were asking almost $20,000 a page for ads. (You have to sell a lot of software to pay those prices.)

This meant that for a small company such as ours—six products, fewer than ten people, and less than $1 million in sales (far less)—it was becoming too expensive to sell in the retail channels. (I also don’t think any of these tactics were doing the customers any favors; the results were often decreased selection and higher-priced products being recommended to customers for the wrong reasons. But that’s another article.)

Our Last Dollar. By the end of 1989, we were literally down to our last dollar. At that time, we had just revised Spellswell and had received quite a few upgrade orders. We decided to put a coupon offering our Lookup and Findswell products at half price into the upgrade package.

To make a long story short, we sold a lot of copies of Lookup and Findswell. Almost 10 percent of customers who received those coupons also bought one or both of the other products. It occurred to us that all might not yet be lost.

...AND BACK AGAIN

The First 200 Pieces. At about the same time, *MacUser* magazine asked if it could mail subscription offers to our customer list, and in return offered to let us make a mailing to an equal number of *MacUser* subscribers. We decided to try it. If retailers started calling us, we would tell them it was only a test.

The catch: We were so broke that we had to scrape to get the money to mail 200 test pieces.

The mailing, made in plain envelopes with only the company name and address on them, included a two-page letter and fliers (all photocopied) for each Working Software product. The offer: Buy each of our products at a 40-

percent discount. It cost about 60 cents to produce and mail each piece (we did it all ourselves)—a total campaign cost of $120.

The Next 60,000+ Pieces. The test was successful. Enough orders came in to finance the mailing of several hundred more pieces to the *MacUser* list. And so many orders subsequently came in that we dramatically expanded our direct-mail effort. We began printing rather than photocopying fliers and offer letters; we tried different messages on the outside of the envelope; we tested different prices. And we sold more and more products. Within four months of the first mailing, we were sending offers to the entire 250,000-plus names on the *MacUser* list.

By mid-1990 we were mailing as many as 60,000 pieces a month—and by the end of the year, we were mailing that many per week and making a healthy profit. The response rates reached 4 to 5 percent, almost no matter what we offered or how the direct-mail piece looked. (Response rates have dropped since then.)

THE EFFECT ON OUR BOTTOM LINE

In 1990 our gross revenues were 600 percent of those in 1989. In 1991 our gross revenues doubled (over 1990), allowing us to develop several new products and upgrade the existing ones. We have also been able to develop a new technology, the System 7 Apple event Word Services Suite, which allows any application that uses it to access any spelling or grammar checker or other word service—as if it were a built-in menu item. (By the way, it's free. Working Software and Apple are providing the source code to application developers. Contact Susan Layman [AppleLink address: LAYMAN1] at the Apple Events Developer Association for more info.)

Our registered-customer list has grown to more than 70,000 buyers of direct-mail products (and we rent our list to others, which provides another source of income from our direct-mail efforts). Our products have received so much exposure from the mailings that we are also once again pursuing—believe it or not—retail-channel sales.

The most important thing we learned was this: You have to reach the end user to sell your product. We discovered that people wanted what our products had to offer but had not been able to find them. For us, selling directly to the customer was the answer.

Overcoming the Fear of Channel Conflict

Why hadn't we tried direct mail sooner? Maybe for the same reasons other companies have steered clear. I had thought about it for years but had always been told that it was important to avoid "channel conflict."

Many people in the software industry have traditionally believed that if you use direct mail, retailers will think you are competing with them and will refuse to carry your products. Why? There are various reasons, not limited to the fact that you *are* competing with them. The retailer loses its percentage of that sale, and direct mail requires using some tactics such as offering a low enough price to convince customers that the "sight unseen" purchase entails an acceptable risk.

However, the reality is that as soon as you sell your products through more than one outlet, you have the potential for channel conflict. For example, if you allow more than one store in a given city to sell your product, they will compete and there will be channel conflict. Of course, individual resellers will want exclusive rights to sell your products. But the fact is that some kinds of sales organizations do a better job than others in selling to a given customer segment, and you have to use what works for you.

I was afraid of channel conflict. Even when we were down to our last dollar, we were afraid to put that half-price coupon into the package with upgrades—but we convinced ourselves to do it. After all, these were already our customers and we were not selling many Findswells and Lookups through stores anyway, so what could it hurt?

I must admit that our use of direct mail did cause problems with retailers. However, now that more and more software companies are using direct mail, I think that retailers are beginning to adapt—they'll have to. And the customer will be the ultimate beneficiary.

The Beauties of Direct Response

Direct mail was once considered a less than desirable, risky way to buy a product. This image is changing. Have you noticed the increase of toll-free numbers used in advertisements from major companies? A recent *ADWEEK* article discussed the movement of major advertisers to direct-response marketing, including Proctor & Gamble's experiments with this concept; it said that P&G is even considering compensating its advertising agencies (and eventually the advertising media) based on the replies it gets to direct-response TV and magazine ads.

Using direct mail is an excellent way to reach customers. Although you are competing with other mail, you aren't buried in a magazine cluttered with many other ads, competing with heavyweights who have more money than you do. The same holds true for store shelves, where you compete for space and for the retailer's attention. In the customer's mailbox, your sales potential is equal to anyone else's.

Also, with direct mail you know exactly whom you sent the offer to and exactly who responded; you can have more space than you would in an expensive magazine ad to tell your story.

You Can Count 'Em. The real beauty of direct mail is that it is measurable. You can test everything. You know what the response is to different things you do. If you put the right tracking procedures in place, you can find out exactly, to within a thousandth of a percent, if changing the offer by $5 increases the response rate, or if printing your letter in blue ink makes a difference.

If You Decide to Test the Waters

Here are some key considerations about using direct mail and some suggestions about how to proceed if you decide to use it.

It Requires a Change in Thinking. Direct mail will change the culture of your company. It certainly changed the way we look at our business. Direct sellers don't seem to worry as much about "corporate image" or "strategic positioning;" they worry more about getting orders and making sure checks clear. This is probably because direct-mail response is immediately measurable. You quickly discover what does and doesn't work, and there isn't a lot of room to get away with silly stuff. In my experience, there is a lot more common sense involved in direct mail than in other forms of advertising.

Margins May Be Lower. The margin on products sold through direct mail can be much lower than those sold through retail channels. (But remember, if you've tested the offer and the list, you'll likely know how much risk you face.) To be successful in direct mail, you have to make a very good offer to customers—often at a price point well below what you've offered at retail.

If you are doing well in retail or other channels, you should consider that the offer you make directly to customers must be good enough to prompt them to buy through the mail rather than running down to the local store. And that means that you are competing with retailers—and with your own products on store shelves.

Start Slowly. Learn the ropes. In our case, the growth from a few hundred to 60,000 mail pieces a week did not just happen. It evolved. It was a learning process, fueled first by an empty cash register and then by the positive reinforcement of a revitalized
cash flow.

We—Mark Galvin (our former marketing manager) and I—read every book and magazine article we could find on the subject and attended seminars. Also, one of the most useful things we did was to carefully examine every piece of response mail we could find; we scrutinized the design, content, wording, placement of graphics, envelope messages, and even the label placement. Imagine reading all your junk mail every day! In addition, we had to learn about printing costs, postage classes and rates, and how mailing houses and list brokers operate.

Should You Do It Yourself? We did it all ourselves because we didn't have any money. It turned out that we had a natural knack for it. But I would not recommend this approach. Direct mail done the wrong way can be much more expensive than hiring a good consultant. Many very good direct-mail consultants are available.

Give It an Honest Try. Don't hedge: Don't fool yourself into thinking that you can send a letter and a flier with no order form, attempting to direct customers to the retail store that carries your product; if customers want your product and can’t order it via your mailer, when they run to the store to buy your product the salesperson may very well “switch” them to a competing product.

So don’t just make the offer; take the order—give customers a chance to order right then and there. Furthermore, you have to be willing to make a good offer; make one that customers can’t refuse, and you’ll find that the orders will flow in.

Test, Test, Test. With proper tracking procedures, you can test everything. Since you can directly measure the results, use that to help you create the most effective campaign possible. Test your offer. Test your list. Test the format and/or presentation of your offering. Testing is the most important factor in the success or failure of your direct-mail campaign.

Be Realistic About Response Rates. What’s acceptable? The response rate that matters is one that makes money. You may have heard a rule of thumb: 2 percent is the acceptable response. Don’t get hung up on it. If you are offering something for $500, you need fewer orders to make money than if you are offering something for $10.

What determines response? See the “Getting the Right Offer to the Right Person” artcle below.

IS DIRECT RESPONSE RIGHT FOR YOU?

That’s a tough question. Direct mail can more than pay for itself when it works. But when it doesn’t, it teaches some very expensive lessons. (We’ve learned a few good lessons, such as don’t drop 250,000 pieces a week before a war starts; don’t commit to mailing 1 million pieces just before the president says we are in a serious recession; and don’t accidentally send a great Mac product offer to a list of IBM PC owners.)

Also, you must decide for yourself what constitutes channel conflict and what is acceptable, given your circumstances.

In our case, direct mail was the right thing to do; it saved our company. And the feeling of control over our own success that comes from being able to measure and track the market’s response to different things we try has made this business fun—and profitable—again.

Dave Johnson is the president of Working Software, Inc., based in Santa Cruz, California.

****

Getting the Right Offer to the Right Person

What determines the response you'll get to your direct-mail offer? Experts say that 50 percent of the response depends on the list, 35 percent depends on the offer, and 15 percent depends on the perceived quality of the mailing piece (that is, the production quality).

Direct mail is about getting the right offer to the right person. The ideal situation is to find a list of people who have a desire to buy and the ability to buy—people who have been looking for a product just like yours and who get large tax returns the day before your offer arrives. And when you begin using direct mail, you will tend to spend a lot of time trying to make the artwork (the letter, brochure, and order form) look nice and read well. Of course, you want to put your best foot forward, but don't do so at the expense of fine-tuning the offer and the list.

To illustrate how important it is to make the right offer to the right person, consider two scenarios:

•*Scenario 1:* You're a Macintosh developer, all your equipment is Macintosh-oriented, and you really want a laptop. You get a poorly-written letter from a guy who says he wants to sell two PowerBook 170s for $1,000 each, still under warranty—if you call him right away. You *will* call him right away. This demonstrates the power of making the right offer to the right list. (The poor production of the mailer probably won't stop you from taking advantage of the offer.)

•*Scenario 2:* You get the same letter, but the seller is offering Toshiba laptops for $1,000 each. You don't call. Great offer, wrong list. The list made the big difference.

Junk mail is in the eye of the beholder. A piece of direct mail is "junk" only when it makes an offer you aren't interested in. When you're spending a nice chunk of change on a direct mailer, you begin to understand that you have a huge incentive to get the right offer to the right person.

Finally, have you ever noticed that many direct-mail pieces look the same? There is a reason: Direct-mail experts test everything—they find out what works, and they stick to it. For example, almost all direct-mail letters use the Courier font. (Don’t ask me why, but it works, and I don’t want to spend the money to try a different font.)

The Global Macintosh Installed Base

by Joan-Carol Brigham and Wendy Weinstein, International Data Corporation

In contrast to days past when the cumulative installed base of Macintosh products lagged behind the rest of the personal-computer pack, the Macintosh has experienced a visible turnaround on a global scale. All indicators now point to continued healthy unit growth worldwide.

International Data Corporation (IDC) research shows that by 1991, the worldwide installed base of Macintosh products had grown to more than seven times the size it was in 1986.

As figure 1 shows (see below), the total installed base grew from just less than 1 million units in 1986 to almost 7 million units at the end of 1991— a compound annual growth rate (CAGR) of 39.5 percent. In 1990 alone, the installed base grew from 4.94 million to almost 7 million units.

Looking ahead (see figure 2 below), IDC predicts that the impressive growth of recent years will continue for the Macintosh. The growth should be driven by several factors: shorter product development cycles; an increase in networking and communications products that “hook” into other products (that is, products that are less stand-alone); a more cross-platform approach to printing and scanning; and the availability of more products on the market.

Although the annual growth rate will probably not be characterized by the strong year-to-year changes of the late ’80s and early ’90s, the Macintosh installed-base curve should continue its upward climb.

By 1996, IDC projects that the total worldwide installed base of Macintosh products will be 19.6 million units, or nearly triple the size of the 1991 high.

Also, when the installed base increases, there’s a corollary increase in shipments. Not surprisingly, then, IDC projects that worldwide shipments of all Macintosh products will grow one and a half times between 1991 and 1996.

In fact, thanks to the low-end products, Macintosh CPU shipments nearly doubled between 1990 and 1991, growing from 1.4 million units to 2.3 million units. Obviously, the installed base demonstrated a corresponding leap in total units.

Another phenomenon—the increasing global acceptance of the Macintosh—emerges if we break down the worldwide installed base by geographic region. Since the mid-1980s, Apple has concentrated on building a presence in international markets. Although language barriers and distribution challenges allowed only a gradual increase in sales at first, the proportion of international sales eventually made a sharp upswing.

Between 1986 and 1991, the geographic mix of the Macintosh base experienced substantial change. As shown in figure 3, (see below) the vast majority of the worldwide installed base in 1986 was in the U.S. At that time, only 16 percent of all Macintosh systems were installed outside the U.S. Of that 16 percent, IDC estimates that the bulk of installations were in Europe. During the following five years, the non-U.S. share of the total installed base reached 44 percent.

Several factors contributed to the geographic shifts shown in figure 3:

•In earlier years, Apple’s international marketing strategy was less clearly defined than it is today. The Pacific region, as defined by the company (which includes Japan, the Far East, Australia, Canada, and Latin America), received relatively little attention. Apple’s European efforts, while more advanced, were still in their nascent stages in the mid-1980s.

•As Apple Computer Europe increased its marketing efforts and offered a broader product line, the company’s visibility in that region improved substantially.

•Starting in 1988, the company redefined its marketing efforts in Japan, the strongest-performing country in the Pacific region. Starting with that year, sales in Japan began to double annually, making a corresponding impact on sales for the entire Pacific region.

Apple’s rapidly-increasing installed base in Pacific markets grew from the application of a Japanese business model—a clear case of learning “When in Rome, do as the Romans do.”

•Sales of personal computers in the European and Pacific regions began to increase between 1986 and 1991, yielding a corresponding increase in sales of Apple/ Macintosh products. Despite these increases, though, those regions still have a low per-capita usage of computers, indicating continued market potential.

•Cost is always a factor. As the European markets entered a period of weakening economies, the premium pricing of Macintosh products hindered sales. Apple’s introduction of a more responsive pricing scheme in 1990 and 1991 has been effective in driving sales.

- Sales in the Pacific region are hampered by the extraordinary personal-space constraints in countries such as Japan, where individual workers do not have the luxury of a traditional office or even a “private” cubicle. When an individual’s total working area is limited to the surface of a single desk, effective use of space becomes a major purchasing factor and affects the nature and number of computers purchased.

INTERNATIONAL FUTURES

In addition to increasing its efforts in many non-U.S. markets, Apple has also instituted improved or newly-introduced translations of the Macintosh operating system (including Kanji for the Japanese market), and has begun to market a more diverse product line internationally.

All these elements, combined with a more favorable purchasing environment, have led to substantially increased sales in all markets. The result: By 1991 Apple had increased its non-U.S. installed base from around 154,000 units in 1986 to a respectable 3.1 million units. In 1991, 31 percent of all Macintosh units installed were in Europe, and the Pacific region had grown to 13 percent of the total installed base.

IDC projects that the international diversity of the Macintosh sales mix will continue to gain strength and that shipments in non-U.S. markets will exceed U.S. shipments by 1993. The cumulative effect is that the U.S. installed base, while remaining strong, will be eclipsed by the combined European and Pacific markets by 1996.

Future Success Factors. The continued success of the Macintosh will be due to many factors:

- A broader acceptance of the personal computer in general, and Apple’s products in particular. Cautious purchasing in European markets in many cases has been driven by a conservative approach to technology and by a reluctance to buy noncommodity products.

As computers become more of a commodity product, people are likely to feel more comfortable buying them.

•The Apple/IBM alliance has yielded far-reaching effects. In all markets (U.S., Europe, and Pacific), it has resulted in many corporate buyers giving broader consideration to Macintosh products than they previously would have. Essentially, the relationship with IBM has strengthened the worldwide market opportunities for Apple.

•Pricing is always an issue. By implementing a new pricing strategy, Apple eliminated its old premium-pricing image and in its place created a highly effective price/ performance model. Although it’s too soon to determine the ultimate effects, the fact that the company is structuring its pricing to be compatible with that of comparable Intel-based personal computers is a promising sign. The immediate effect is obvious: By altering the pricing scheme, the company can sell more product, and the installed base will grow. It opens the market to users and corporations that previously may have held back.

•The dynamics of the international markets themselves are changing. Most notably, the integration of Eastern Europe into the European region introduces products to a new group of consumers. The ease of use offered by the Macintosh should drive sales in those new markets.

IDC anticipates similar effects with the opening of many Latin American markets. Many Latin American countries that previously had heavy import restrictions are beginning to allow more-unrestricted imports of computer products. Gray markets previously dominated Latin American sales, but the reduction or elimination of import restrictions will create opportunities to build a distribution network.

GROWTH THROUGH 1996

As the worldwide Macintosh installed base continues to grow, it will create a self-perpetuating side effect: Apple, as a company, will appear stronger and more viable—which in turn will attract more customers.

Of the three regions, the U.S. market will probably remain dominant, growing from almost 4 million installed units in 1991 to 9.54 million units in 1996, or close to three times its current size (see figure 4 below).

Taken individually, Pacific and European markets will outpace the U.S. in compound annual growth (33 percent and 29 percent, respectively). Figure 4 shows that the European market will also triple in size, growing from 2.2 million units in 1991 to 6.9 million units by 1996. The Pacific market displays similar growth rates, and more than triples during that time from .88 million units in 1991 to 3.1 million units in 1996.

As mentioned earlier, the market mix will also change by 1996. Figure 5 shows that in 1991 the United States represents 56 percent of the total installed base, with Europe at 31 percent and the Pacific at 13 percent. By 1996, IDC predicts the installed-base composition to be the U.S. at 49 percent, Europe at 35 percent, and the Pacific at 16 percent.

It’s clear that the Macintosh will continue to gradually become established with an increasingly global customer base, and will experience long-term, solid growth.

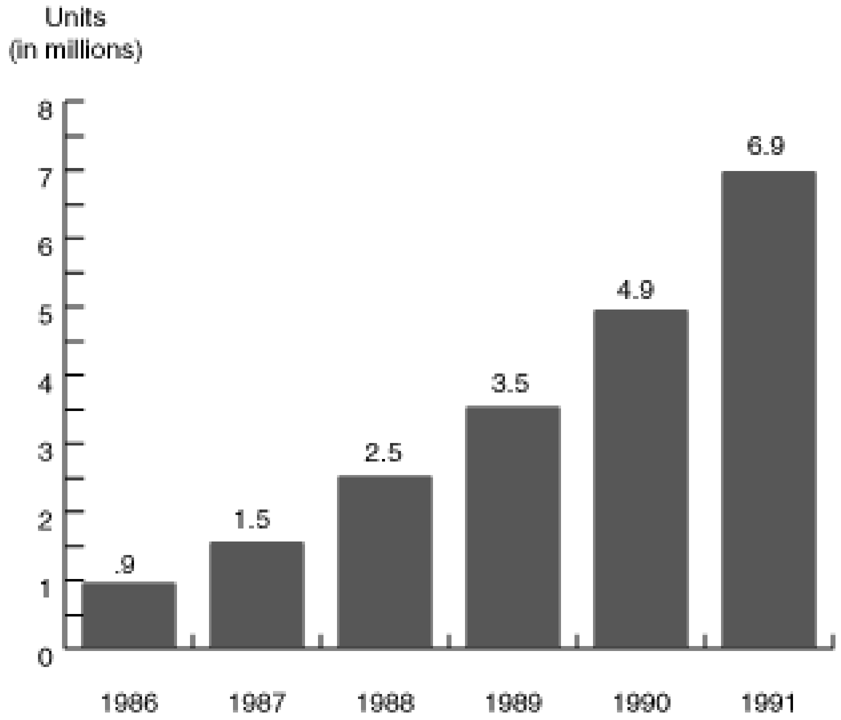


Figure 1: An historical perspective of the worldwide Macintosh installed base, 1986–1991

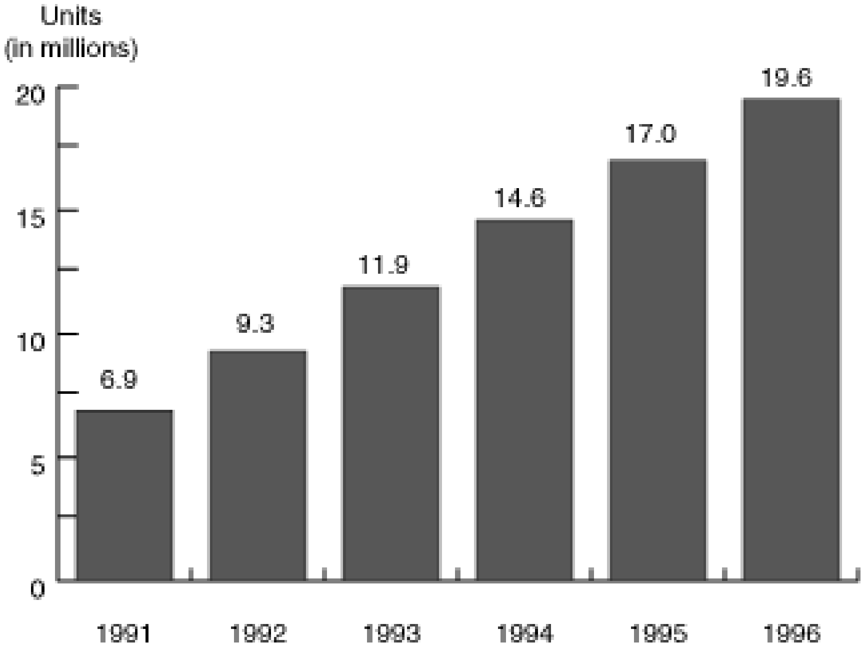


Figure 2 : 1991–1996 forecast of the Macintosh installed base

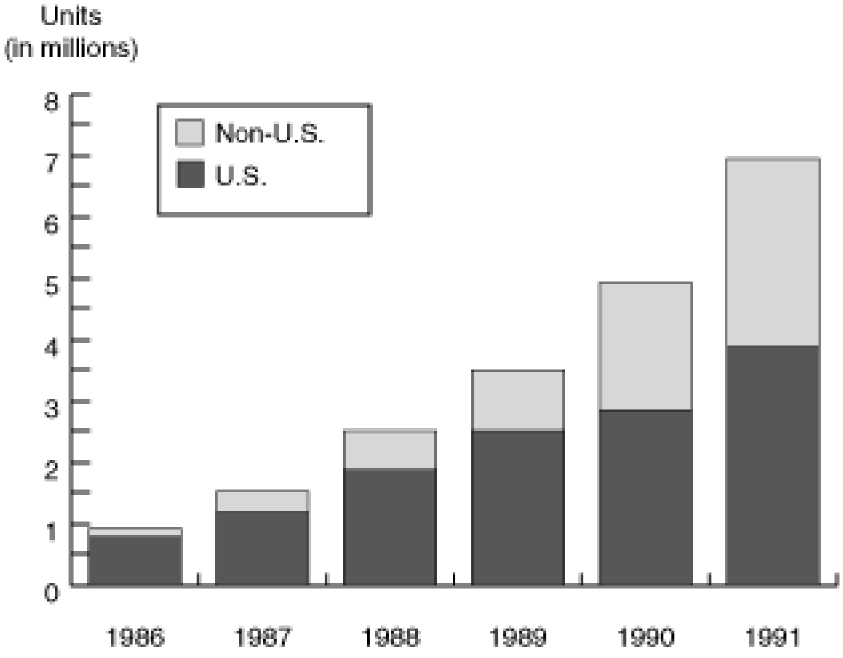


Figure 3: Global distribution of the worldwide Macintosh installed base, 1986–1991

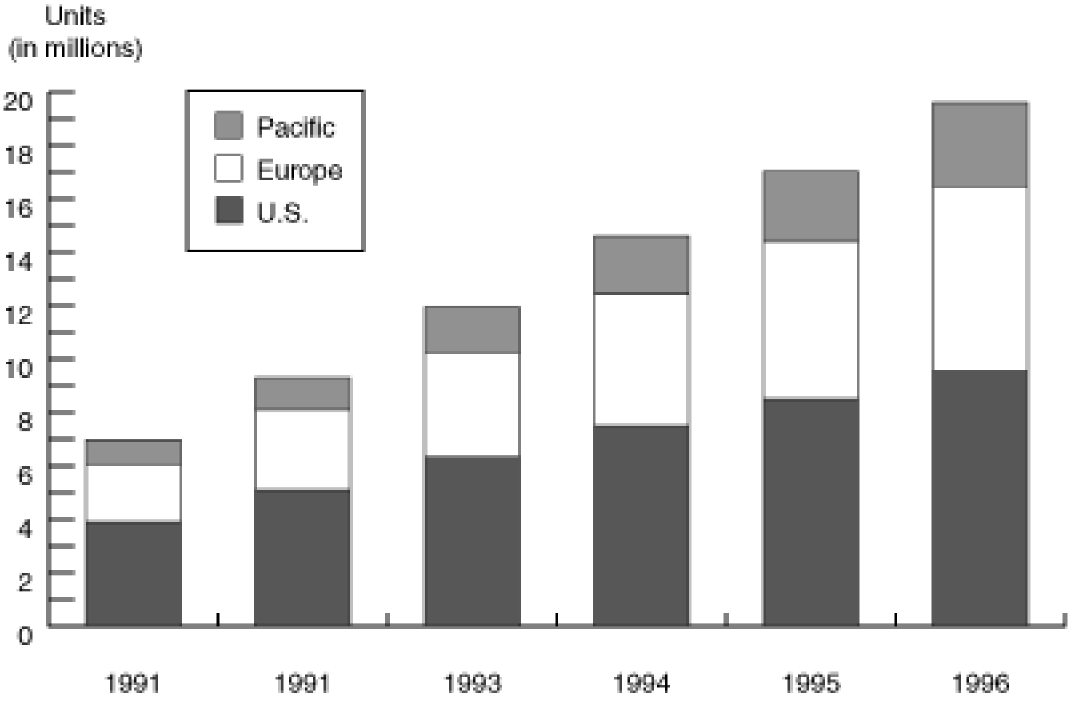


Figure 4: 1991–1996 geographic distribution of the worldwide Macintosh installed base

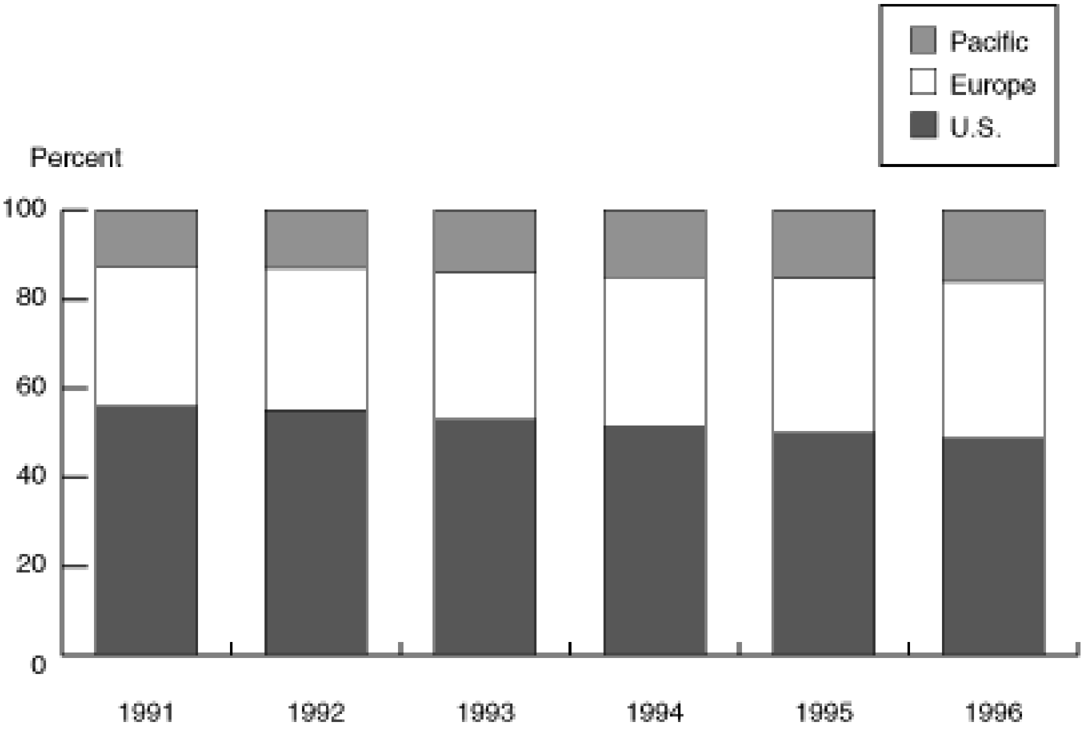


Figure 5: 1991–1996 percentage share, by geographic region, of the worldwide Macintosh installed base

Slicing the Market by Product
A Look at Apple's Product Mix

by Joan-Carol Brigham and Wendy Weinstein,
International Data Corp.

As a developer, you need to know more than installed-base numbers to plan your business. It's also essential to understand how the numbers break down for individual products, product categories, and geographies. With that in mind, let's take a close look at the structure of the Macintosh installed base from a product perspective.

DEFINING FOUR PRODUCT CATEGORIES

International Data Corporation (IDC) has created four categories for Macintosh models—portables, low-end, mid-range, and high-end—and defines them according to product pricing and functionality. Representative products for the categories are as follows (you'll see that the list includes products that Apple doesn't manufacture any longer; these models are important to help you understand the long-term view):

- The *portable* products include PowerBooks and the now-discontinued Macintosh Portable.
- The *low-end* products are represented by the compact devices and nonexpandable units, as well as some of the modular machines. The products included in the 1991 definitions are the Macintosh Classic, Classic II, and LC. (The new LC II, introduced in March 1992, would have belonged to this category.)
- The *mid-range* products are analogous to what IDC calls "business-class machines." These are characterized by a higher degree of functionality and expandability than are low-end systems. They include the Macintosh IIci, IIfx, and IIcx models.
- The *high-end* products are more sophisticated and feature faster, more responsive processors. The Macintosh Quadra systems fall into this category, because of the higher functionality and pricing for the base CPU. Although IDC initially designated the Macintosh IIfx as a high-end product, the introduction of the Quadras, along with pricing changes, moves the IIfx into the mid-range

classification. The Quadras redefined the ceiling for high-end products, and thus the Macintosh IIfx was reclassified.

These designations are subject to change, of course, as Apple redefines its markets and how it positions products.

THE EARLY YEARS: 1987–1991

By 1987 there were two key product categories: low-end and high-end. The product offerings consisted of the Macintosh XL, 512, Plus, SE, and II. IDC assigned these products primarily to the low-end category, based predominantly on pricing at the time. As a result, the 1987 installed-base product mix was 95 percent low-end and 5 percent high-end (see figure 1 below).

By 1991, Apple had substantially broadened the Macintosh product line and had introduced products that fit into the four categories described above. In fact, the company completely redefined the low end of the product line in 1990, with the introduction of the Macintosh Classic. This and subsequent products that were designed for low-end markets have substantially boosted the size of the overall installed base.

In fact, the Classic can claim to be the single most successful product introduction in the company’s history; worldwide unit sales in its first year completely outpaced those of any previous product. As can be seen in figure 2 below, sales of the Classic and of the broader spectrum of Macintosh products yielded two results: a significantly larger installed base and greater product diversification.

Although low-end products had represented 95 percent of the total installed base in 1987, that dropped to 74 percent by 1990. Over time, the low-end share has decreased in size, because of introduction of products in other categories. However, between 1991 and 1996 this category is predicted to retain a majority of all the installed products.

LOOKING AHEAD
TO 1996

The bulk of the installed base in 1991 comprised low-end and mid-range products. However, 1991 was a critical year for Apple, with the introduction of solid offerings in both the high-end and the portable segments. figure 3, below, shows that IDC projects an increasing diversif- ication of the worldwide installed base by 1996, with the high-end increasing to a 2.3 percent share; the mid-range to 20 percent; and the low-end continuing to dominate the installed base at 67 percent (see figure 4 below).

Portables are predicted to represent 11 percent of all installed products by 1996. This is a dramatic change from 1.2 percent in 1990, and even 2.9 percent in 1991. During 1991, Apple introduced notebook computers to a market starved for effective portable Macintosh products. The results are obvious: Even though the notebook computers did not begin shipping until November 1991, Apple's worldwide portable sales jumped to 142,000 units, purely due to the new products.

By 1996, sales will probably climb to more than 470,000 units. This dramatic change will result in a more than tenfold growth of the installed base of portable products, from 199,000 units in 1991 to 2.1 million units in 1996.

Portables sales will be driven by significant enhancements in connectivity and networking, combined with the use of leading-edge technology in the portables themselves.

RISC-Based Products. By slightly adjusting the existing product categories we will be able to add RISC-based products. IDC fully expects that Apple will develop such products for the four Macintosh product classifications.

A GEOGRAPHIC VIEW

Changes in the product mix will evolve differently within Apple's three sales regions. The U.S. market—where the low-end, mid-range, and high-end products (as well as the portables) should all fare well—will continue to be the most diverse.

It will be a different story, though, in other markets. In both the European and Pacific regions, the mix will be more closely tied to
specific product models and categories.

Europe. In Europe, the mid-range business machines should sell quite well, although the lower per-capita usage of personal computers in general will affect sales. IDC expects that low-end systems will be the strongest sellers there, based on lower pricing and smaller footprint of the products.

From the buyers' viewpoint, trade-offs in functionality and expandability in the low-end machines will be well balanced by pricing and ease-of-use considerations.

Additionally, the incorporation of Eastern Europe into the greater European community will spur sales of low-end products. Again, this is an issue of a fair exchange for cost and functionality.

Finally, as Eastern Europe affects the overall European eco- nomy, money supplies will be short until the Eastern European countries become more economically stable and independent.

Pacific. Macintosh sales in the Pacific region will be shaped largely by the buying characteristics and patterns of Japan and other Pacific Rim countries. Japan, especially, is affected by major personal-space constraints that make the smaller, portable products extremely attractive. Interestingly, the Japanese markets also demonstrate a shyness about personal desktop computer usage, which also contributes to the popularity of portables and Macintosh products that combine space/ size advantages with a more user-friendly approach to computing.

However, it is important to remember that the European and Pacific regions display some similar characteristics. Price/performance issues exist in both regions, with consumers looking to get the best value for their money. This means that low-end products will do well in the Pacific and European markets.

Buyers in each region will also appreciate the convenience of the portables, which should sell well in Europe and Japan. However, in these two regions we won't see the phenomenon of customers purchasing portable products as second machines. That characteristic will remain strongest in the U.S. market.

Sales Will Differ by Product Category. By focusing on two key product areas, it's possible to see differences in sales between the three major geographic regions. The U.S. market is predicted to remain dominant in the low-end category between 1991 and 1996 (see figure 5 below), with sales ranging from 55 percent of all products in 1991 to 44 percent in 1996. Europe's share of low-end sales will grow from 30 percent to 33 percent by 1996, and the Pacific will reach 23 percent in 1996, from a base of 15 percent in 1991.

However, the three markets will show markedly different growth rates, with the Pacific region demonstrating the strongest growth: Its compound annual growth rate (CAGR) will be 31 percent, compared with an 18-percent CAGR for Europe.

Portables should also exhibit marked differences in sales between geographic regions, as shown in figure 6 below. In 1991, sales of portables were split as follows: U.S., 59 percent; Europe, 25 percent; and Pacific, 16 percent.

By 1996, the geographic mix will probably have changed to reflect some regional differences: U.S., 44 percent; Europe, 34 percent; and Pacific, 22 percent. The highest growth is predicted to be in the Pacific region, with annual sales increasing more than 4.5 times between 1991 and 1996, from 22,500 units to 103,425 (a CAGR of 54 percent).

With the total size of the installed base predicted to increase from 7 million units in 1991 to 19.6 million units in 1996, it is apparent that all product categories should experience strong growth. Apple is actively developing products to respond to market demands. Now that the company has a broader product line, all indications are that the diversity will be maintained. RISC-based Macintosh computers will add to the product-line differentiation, giving Macintosh users a full range of power and functionality options.

Joan-Carol Brigham is director of graphics and Macintosh research at International Data Corporation. Wendy Weinstein is a consultant to International Data Corporation.

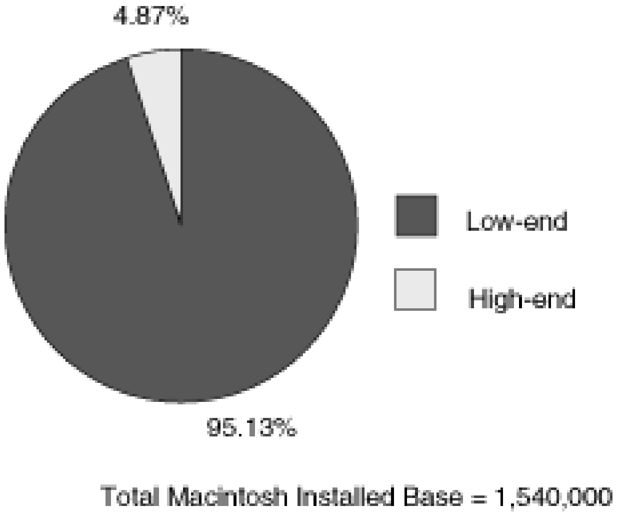


Figure 1: 1987 worldwide product distribution of the Macintosh installed base

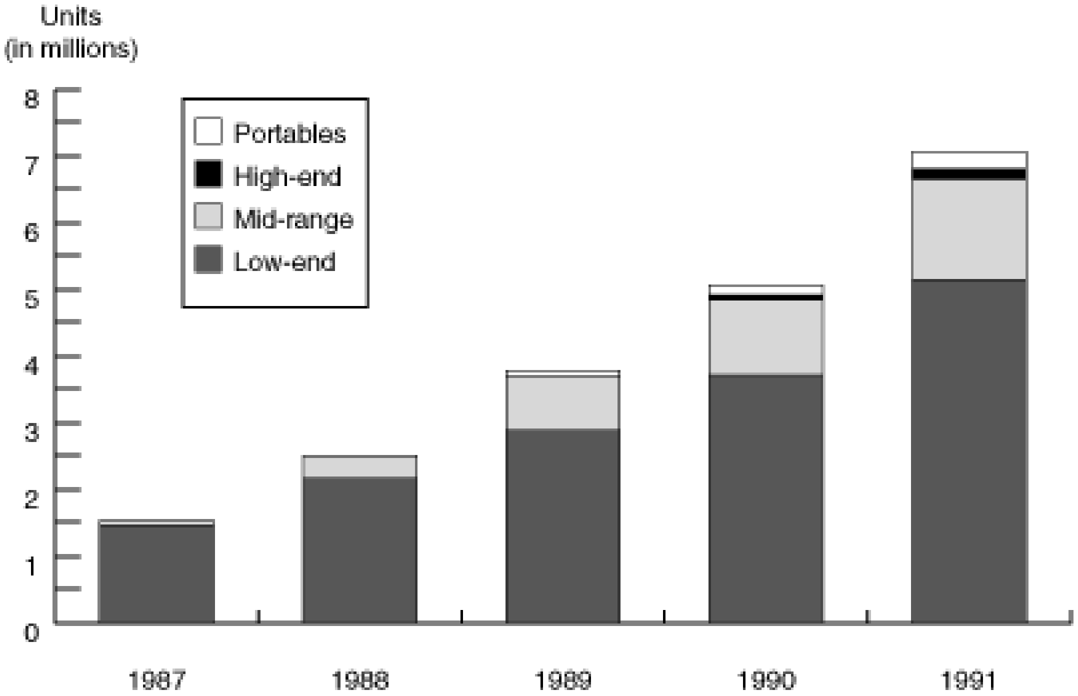
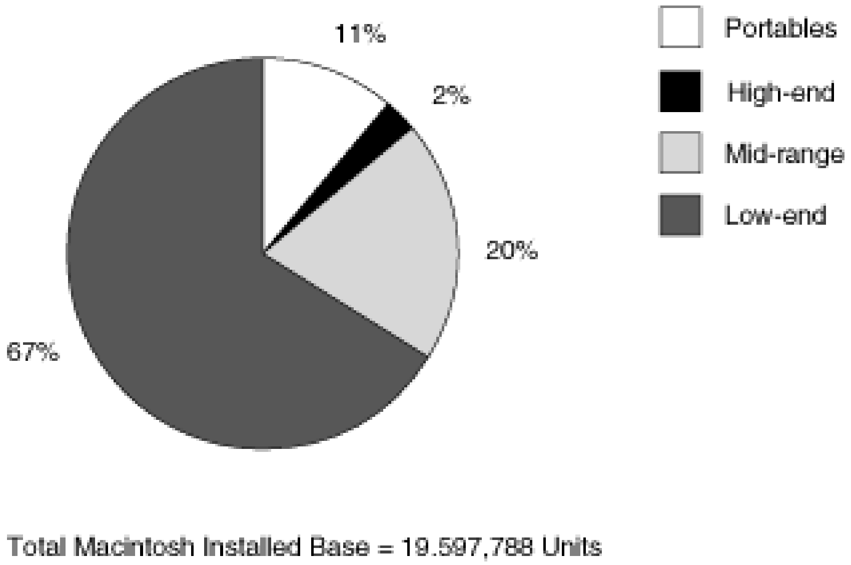


Figure 2: Product distribution of worldwide installed base, 1987–1991



(Note: Chart uses 1991 product category designations)

Figure 3: Product distribution of worldwide installed base, 1991–1996

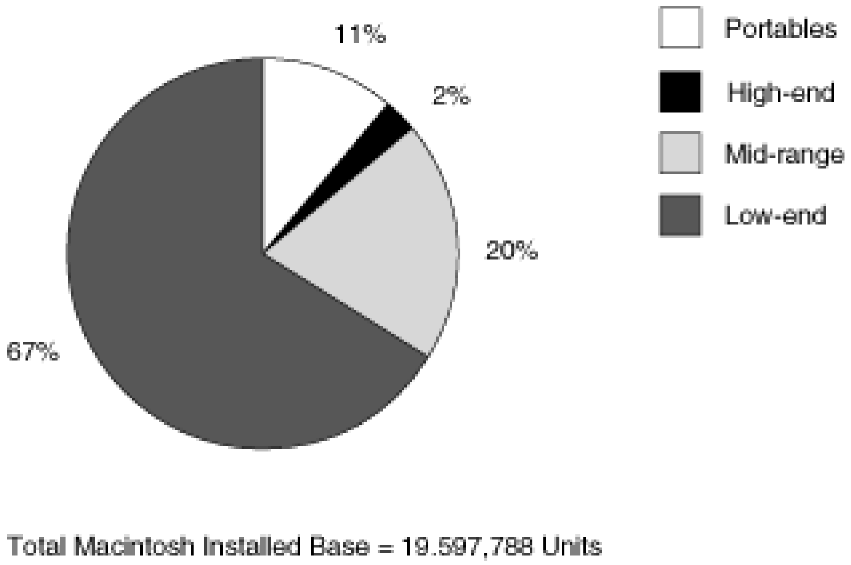


Figure 4: Product distribution of worldwide installed base, 1996

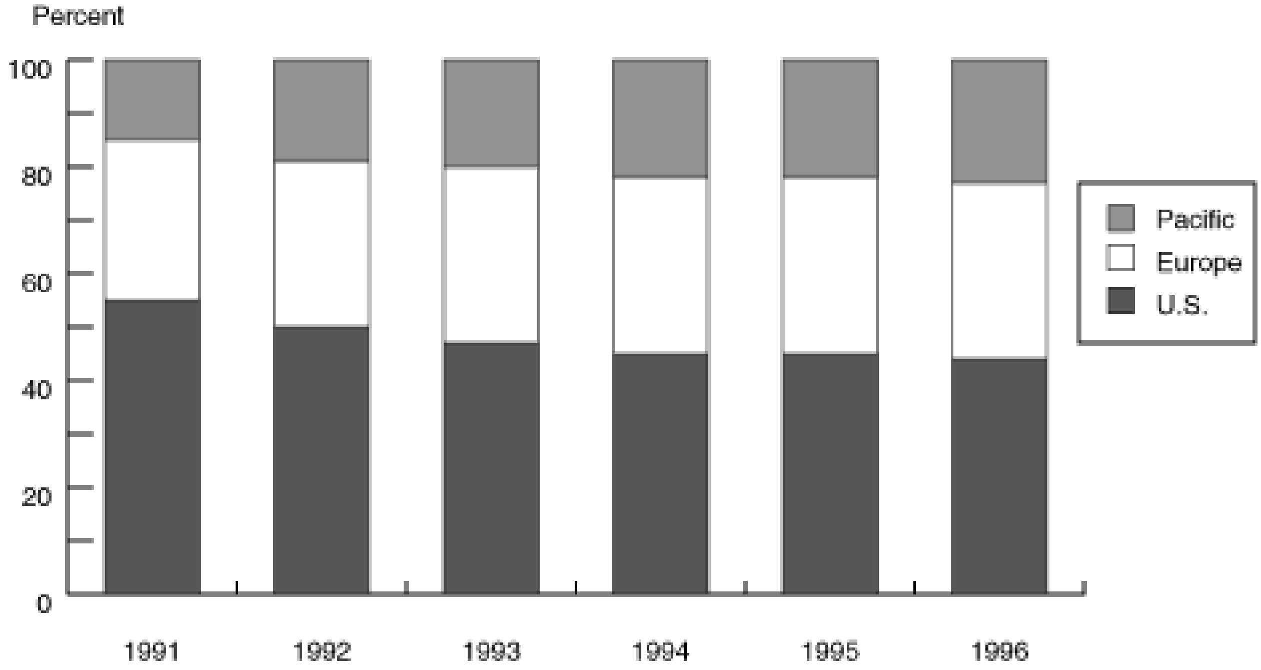


Figure 5: Regional distribution by percentage of low-end Macintosh shipments, 1991–1996

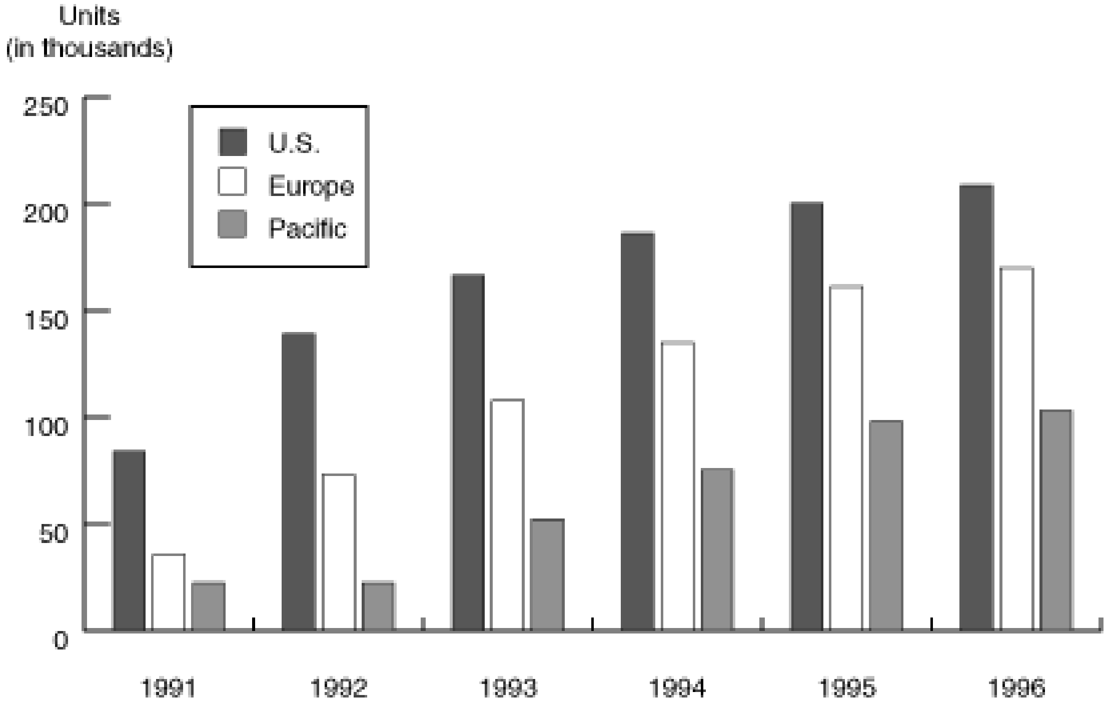


Figure 6: Regional distribution of portable Macintosh shipments, 1991–1996

In-Depth Market Research Data Now Available in *Apple Direct*

Welcome to the debut of Market Research Quarterly. This section, which will appear every three months, will bring you in-depth market-research information that can help in your business planning.

The information in this section is supplied by International Data Corporation, a veteran provider of market data and guidance to the information industry and MIS vendors.

Once each quarter, you can expect to see useful market-research data and accompanying analysis. The section will usually be two pages long; however, this special premier edition gives you four pages of market research information. In other months, look for our regular Market Insight in the Business & Marketing section of *Apple Direct.*

To help you get a more solid and clear understanding of the opportunities in the Macintosh market, this issue of Market Research Quarterly focuses on the size of the worldwide Macintosh installed base, the installed base product mix, and a snapshot of what is predicted for coming years.

For future issues, we would like to hear from you regard-ing what kinds of market research information would be helpful.

APPLE DIRECT

Apple Direct is Apple’s monthly developer newspaper, covering business and technical issues for decision makers at development companies. It is published by Apple Computer, Inc.’s Developer Support Systems and Communications (DSSC) group.

Publication team:

EDITOR-IN-CHIEF:
Dee Kiamy (AppleLink: KIAMY)

TECHNICAL WRITER/EDITOR:
Gregg Williams (GREGGW)

PUBLICATIONS AREA MANAGER:
Hartley G. Lesser (H.LESSER)

FORMATTER:
Lisa Ferdinandsen (LISAFERD)

CONTRIBUTORS:
Jessa Vartanian, Suzanne Dills, Sharon Flowers, Monica Meffert, Ana Wilczynski, Lisa Raleigh, Liz Westover, Jennifer Highes, Pat Calderhead, Marian Djurovich, Paul Dreyfus, Juan Bettaglio

MANAGER, DSSC:
David A. Krathwohl

CONTENT GROUP MANAGER:
Greg Joswiak

FILM:
Aptos Post, Aptos, CA

PREPRESS:

Prepress Assembly,
San Francisco, CA

PRINTER:
Wolfer Printing Co., Inc.
Los Angeles, CA

Mention of products in this newspaper is for informational purposes only and constitutes neither an endorsement nor a recommendation. Apple assumes no responsibility for the accuracy of the information supplied by International Data Corporation.

All product specifications and descriptions were supplied by the respective vendor or supplier. Apple assumes no responsibility with regard to the selection, performance, or use of the products listed in this newspaper. All understandings, agreements, or warranties take place directly between the vendors and prospective users. Limitation of liability: Apple makes no warranties with respect to the contents of products listed in this newspaper or of the completeness or accuracy of this publication. Apple specifically disclaims all warranties, express or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose.