MAC POWER
Using Macintosh Software
Allen Munro
Figure 10.1 is reproduced with the permission of T/Maker Graphics Company.

The following figures are reproduced with the permission of Microsoft Corporation:
4.2, 4.4, 4.5, 4.6, 4.8, 4.9, 4.10, 4.12, 4.13, 4.14, 4.15, 4.16, 4.17, 4.18, 4.19, 4.20, 4.21, 4.22, 4.23, 4.24, 4.25, 4.26, 4.28, 4.29, 4.30, 4.31, 4.32, 5.2, 5.3, 5.4, 5.5, 5.6, 5.7, 5.8, 5.9, 5.10, 5.11, 5.12, 5.13, 5.14, 5.15, 5.17, 5.18, 5.19, 5.20, 5.21, 5.22, 5.23, 5.24, 5.25, 5.26, 10.14, 10.16, 10.17, 10.18, 10.19, 10.20, 10.21, and 12.3

Figure 1.1, a Screen Print from Penguin Software’s Transylvania, is reproduced with the permission of Penguin Software, Inc.

Figures 12.1 and 12.2 are reproduced with the permission of DB Master Associates, Inc.

Trademarks

Apple, Apple II, IIe, and III are registered trademarks of Apple Computer, Inc. The Apple logo, the Macintosh logo, Lisa, The Finder, MacDraw, MacPaint, Macintosh Pascal, Macintosh Assembler, MacProject, MacTerminal, and MacWrite are trademarks of Apple Computer, Inc. Macintosh is a trademark licensed to Apple Computer, Inc.

ClickArt is a trademark of T/Maker Graphics. Compuserve is a registered trademark of Compuserve Information Services, Inc. Corvus Systems, Omninet, and OmniDrive are trademarks of Corvus Systems, Inc. DB Master and Stoneware are trademarks of DB Master, Inc. DEC, VAX, VT100, and VT52 are registered trademarks of Digital Equipment Corporation. Filevision is a trademark of Telos Software Products. Framework is a trademark of Ashton-Tate. IBM is a registered trademark of International Business Machines Corporation. Lotus 1-2-3, Jazz, and Symphony are trademarks of Lotus Development Corporation. MacForth is a trademark of Creative Solutions, Inc. Mac the Knife is a trademark licensed to Miles Computing, Inc. Microsoft, Chart, Multiplan and Word are registered trademarks of Microsoft Corporation. Motorola is a registered trademark of Motorola, Inc. Osborne 1 is a trademark of Osborne Computer Corporation. Penguin Software’s Transylvania was written by Antonio Antiochia. Macintosh version by Robert Hardy, with graphics by Holly Thomason. Transylvania is copyrighted 1984 by Penguin Software, Inc. All rights reserved. Penguin Software is a trademark of Penguin Software, Inc. Transylvania is a registered trademark of Penguin Software, Inc. SmoothTalker is a trademark of First Byte, Inc. Softech Microsystems is a registered trademark and The MacAdvantage is a trademark of Softech Microsystems, Inc. The Source is a service mark of Source Telecomputing Corp., a subsidiary of the Readers Digest Association, Inc. Teemar is a trademark of Teemar, Inc. ThinkTank, ThinkTank 128, and “The first idea processor” are trademarks of Living Videotext, Inc. UCSD p-System is a registered trademark of the Regents of the University of California. Visicalc is a registered trademark of VisiCorp. XEROX Star is a registered trademark of XEROX corporation.
for Pam
Acknowledgments

Many people generously contributed their time and skills to this project. Application developers took time from their schedule-driven lives to explain their applications, and to read and comment on the portions of *Mac Work* that treat their products. Friends, relatives, and colleagues critically read the book and offered suggestions. I thank Greg Ames, Suzette Elgin, Eileen Fussner, Bob Foster, Larry Gorbet, Ricky Jacobs, Jan Josephs, Mark Johnson, Guy Kawasaki, Scott Love, Dorothy Munro, Joseph Munro, David Murray, Andrew Singer, Roz Schwartz, Doug Towne, Charles Ulrich, and Dave Winer for their help. Floyd Flanagan worked through the book chapter by chapter as he wrote his own book on MacPaint. Most of all, I thank Pam Munro for encouragement, support, and heroic editing through many drafts.
## CONTENTS

AKNOWLEDGMENTS vii

**INTRODUCTION: WHAT MAKES YOUR MACINTOSH SPECIAL** 1

- Special Features of the Mac 2
- Using Mac Power 9

**CHAPTER 1: PICTURES** 11

- The Mac's Graphics Capabilities 12
- MacPaint's Features 14
- Techniques for Using MacPaint 15
  - Title Boxes 23
- Transferring Material Between Documents 23
- The MacPaint *Print Catalog* Feature 27
- Some MacPaint Shortcuts 27
  - Fats Bits Shortcuts 27
  - Show Page Shortcuts 29
  - Other Shortcuts 29
- Problems and Techniques with MacPaint 30
  - Using the Grid 31
  - Tiny Text 35
  - Recovering from Error 35
  - Making Room for Documents 35
CHAPTER 2: WORDS

Word Processing with the Mac 40
Word Processing Applications—Choices 42
Using MacWrite 43
MacWrite Features 44
Techniques for Getting the Most from MacWrite 51
One-Drive MacWrite 51
Dummy Text Documents 53
Printing 55
Long Texts 56
Accidental Text Selections 58
Disasters 59
Keyboard Commands 61

CHAPTER 3: MANAGING YOUR MACINTOSH DISKS

About the Mac’s Disks 63
The Disk Directory 64
Different Views of a Directory 68
The Importance of Backing Up 69
One-Drive Disk Copying 71
Two-Drive Disk Copying 72
Backing Up Your Own Work 72
When the Party’s Over 77
Printed Directories 78
CHAPTER 4:  SPREADSHEETS  
Using Multiplan—An Example 81
  The Base Canard Widget Project Budget 83
  A Budget Summary 99
Showing Formulas 103
Charting in Multiplan 104
Other Kinds of Spreadsheets with Multiplan 106
Multiplan’s Command Keys 108
Books About Multiplan 108

CHAPTER 5:  CHARTS  
Making a Pie Chart 111
Charts with More than One Set of Data 120
Using Multiplan and Chart Together 127
Further Reading About Microsoft Chart 129

CHAPTER 6:  OUTLINES  
Using ThinkTank 131
Printing Outlines 140
Moving Outlines to Text Documents 142
Things to Do with ThinkTank 143

CHAPTER 7:  PICTORIAL DATA BASES  
Features of Filevision 147
Manipulating the Display 150
  Controlling the Appearance of Objects 150
  Highlighting 150
  Complex Highlighting Specifications 152
Printing Lists 154
The Range of Filevision’s Uses 157
  Intelligent Maps 157
CHAPTER 8: PROJECTS

Managing Major Projects with the Mac 175
Creating a Schedule Chart with MacProject 175
Using a Completed MacProject Schedule 188
Planning and Tracking 193

CHAPTER 9: TALKING WITH OTHER MACHINES

No Mac Is an Island 195
Four Communication Methods 195
    Micro-Floppy Disks 196
    Local Serial Connection 196
    Telephone Connection 196
    Local Area Network (LAN) 196
How to Start Communicating Electronically 197
Using MacTerminal 200
Customizing MacTerminal for the Computers You Talk To 205
Two Local Area Networks 209
Two Styles of Machine Communication 212

CHAPTER 10: MORE PICTURES, MORE WORDS

Help with MacPaint 214
Creating Graphics with a Word Processor 217
Introducing MacDraw 218
Help with MacWrite 226
Word Processing with Multiplan 229
Advanced Word Processing with Microsoft Word 229
Long Documents 234
Wide Documents 234
Paragraph Formatting 235
Automatic Footnotes 235
Working with Two Parts of a Document at Once 236
Editing More than One Document at Once 236
Print Merging 237

CHAPTER 11: PROGRAMMING WITH MACPASCAL 238
Introducing Macintosh Pascal 239
Structured Programming and the Realities of Personal Programming 240
How MacPascal Fills Apparently Competing Requirements 243
Using MacPascal to Program Kinetic Lines 245

CHAPTER 12: THE MAC GOES ON 255
The Standard Macintosh User Interface 256
More Programming Languages 259
Microsoft BASIC 260
MacForth 261
Macintosh BASIC from Apple 261
FORTRAN 261
LISP 262
Logo 262
Modula-2 262
C 262
Upgrading Your Mac 263
A Fat Mac Upgrade 263
Why a 512K Mac? 263
A Hard Disk Upgrade 266
Macintosh Accessories and Paraphernalia 269
Magazines 269
APPENDIX: MOUSE SKILLS

Introducing the Five Mouse Skills 277
Pointing 277
Clicking 278
Double-Clicking 279
Shift-Clicking 279
Dragging 279
Selecting and Unselecting 280
Choosing from Menus 283
Dragging 286
Windows 286
  Opening Windows 289
  Closing Windows 292
Dialog Boxes 294
Keyboard Commands 295
Multiple Selects 299
Manipulating Windows 300
  Scrolling 302
  Moving Windows 305
  Overlapping Windows 306
Problems and Techniques 308
  Ejecting a Disk 308
  The Programmer's Switch 309
  Invisible Icons 310
  Dragging Accurately 311
  Room for Mouse Moves 312
INTRODUCTION: WHAT MAKES YOUR MACINTOSH SPECIAL

SPECIAL FEATURES OF THE MAC 2
USING MAC POWER 9
The Apple Macintosh computer is a revolutionary product. It is easier to learn to use than other computers, and its application programs can be used together more easily and more productively. It is by far the best moderately priced computer for producing presentation-quality printed materials. Your Mac lets you directly manipulate information—particularly non-verbal information—more directly than is possible with any other personal computer. These are some of the reasons you bought a Macintosh. This book will help you get the most from your Mac by showing you how to use APPLICATIONS—major programs—effectively.

SPECIAL FEATURES OF THE MAC

Some of the characteristics that make the Apple Macintosh special are obvious when you look at the machine. Others become apparent only when you see it being used or, better yet, use it yourself. The Macintosh provides a combination of powerful hardware and software features, carefully integrated to contribute to the productive use of the machine.

HARDWARE is the physical, touchable elements of a computer, such as the central processing unit (CPU)—a single large integrated circuit that executes all the instructions. Other hardware elements include the Random Access Memory (RAM) chips and their supporting circuits, as well as the mouse, the disk drive, and the screen. SOFTWARE means the programs and data that make the Mac do everything it does. Some software is always present in the Mac, hidden in special chips called ROM (Read-Only Memory) chips. The software stored in ROM is sometimes called FIRMWARE to distinguish it from the rest of the Mac's software, which is stored on disks.

The innovative use of ROM in the Macintosh contributes to the Mac's success in two ways. First, it is extremely efficient. This means that the program code in ROM runs very quickly. This is an important part of the reason for the Mac's very good graphics display speed. Second, the Mac ROM contains a natural and easily learned USER INTERFACE, a set of consistent methods for letting users interact with applications programs. The Mac's user interface includes the use of simple pictures to represent user-selectable items, descriptive menus for command choices, and simple interactive dialog boxes. These methods are natural and easy to learn. Because the user interface is in
ROM, virtually all Mac applications use it. Macintosh users don’t have to learn a completely new style of interacting with their machine when they start using a new program. Their Mac-using skills carry over to every application.

The Macintosh uses hardware that is unusually powerful for a moderately priced personal computer, and all its software—including the major applications and the underlying ROM routines—is designed to fully exploit that power. The Mac’s combination of hardware and software power provides four benefits to users.

1. The graphics basis of the Macintosh makes it possible for the user to achieve detailed control of the screen and the printouts in ways that were not possible before. With the Mac, you can easily mix text and graphics on the screen, just as they will be printed. You can even compose text documents that use many different character sizes and fonts.

2. It is easy to learn to use. This is true in part because the way users interact with applications is very natural, and in part because almost all applications use the same interaction methods.

3. It lets experts get the most from the system without being slowed down by methods designed for beginners.

4. It provides an INTEGRATED ENVIRONMENT—material created with each application can be used in most other applications.

One of the most apparent characteristics of the Macintosh is its mouse. The Mac was the first computer in the personal price range to offer a mouse as standard equipment. About the size of a bar of soap, the mouse sits on your desk near the computer. As you slide it around in a small area near the Mac, a small symbol called the pointer—usually a slanted arrow—moves correspondingly on the screen. You move the mouse to point to elements on the screen.

If you are a Macintosh user, you are probably already familiar with the uses of the mouse. If you are new to the Macintosh, turn to the Appendix, "Mouse Skills," for an introduction and a few advanced tips on using the mouse.

A second important feature of the Macintosh is its transportability. Some other portable computers are significantly smaller and more portable than the Mac. But no portable computer before the Mac offered such a sophisticated way of interacting with the user.
No other transportable computer offers the powerful internal hardware and software that the Mac does. The Mac uses a very advanced central processing unit, the Motorola 68000 chip. This processing unit runs more quickly than many others, and it can work with larger amounts of data at one time. Therefore, the 68000 can do more in the same time than less powerful processors. In practical terms, this means that the Mac’s CPU has sufficient power to support a sophisticated user interface without running more slowly than computers that use more primitive ways of interacting with users.

A number of other computers use the same 68000 processor in systems that support use by more than one user at a time. All the users have separate terminals, which are connected to the computer with cables. In those systems—which can cost more than ten times as much as a Macintosh—each user must share the single 68000 processor with all the other users on the system. The Mac devotes all the power of the 68000 to one user.

A personal computer should be portable. People don’t ordinarily want to do all their computer use at only one location. Normally, achieving the smallness and lightness needed for portability can be expected to result in sacrifices in performance. The Macintosh has avoided almost all such compromises by using one of the most powerful processors available, by providing enough memory—including ROM chips with software that usually takes up RAM space in other machines—and by using a smaller and higher-capacity disk drive than normal.

A single-sided 3½-inch Macintosh floppy disk can hold more information than most 5½-inch disks. Yet the Mac’s 3½-inch drive mechanism itself weighs only a fraction of the 5½-inch disk drive. This weight saving helps to make the Mac compact and portable.

The size of the base of a computer is called its “footprint.” The Mac’s small footprint is another bonus. Before the Macintosh, “desktop computers” required desks of their own. Adding an ordinary personal computer to an office meant adding furniture just for the computer. Because the main case of the Mac takes only 10-by-10 inches of desktop, it should fit easily into your work space.

When you turn on the Macintosh, a number of special features become obvious. The first such feature is the high-resolution display. The Mac’s graphic display was a real breakthrough in moderately priced computing. The use of a black-on-white display makes the Mac much easier on the eyes than other computers that use green-on-black or amber-on-black screens. The high resolution of the display was another step forward. The Mac display is
512 dots across and 342 vertically. To get an idea of how much more resolution this provides than a traditional microcomputer, take a look at the Mac's screen in the *Transylvania* game from Penguin Software (Figure I.1).

This game was adapted from one developed for the Apple II line of computers. On the Apple II, the entire screen was filled with the picture shown in the upper left corner of the Mac's display. The Macintosh provides more than three times as many viewable dots, or pixels, as did the Apple II.

The Macintosh display resolution appears even greater than its 512-by-342 dots. One reason for this is that the monitor paints each pixel with exceptional precision. Another is that each pixel is square in shape, rather than rectangular, as with most other computers. In addition, the monitor refreshes the entire screen sixty times per second, rather than only thirty, as is more common for high-resolution displays.

![Figure I.1 A screen print from *Transylvania*](image)
Another special Macintosh feature is windowing. A window is a rectangular area of the screen that presents information from some source, such as an application. Because several windows can be open at the same time, you can keep material from several different sources on the screen at the same time. For example, you can keep material from the Scrapbook on the screen at the same time that you edit a memo using the MacWrite word processor (Figure I.2). The MacWrite window can be positioned low on the screen, leaving room to display a Scrapbook page above. In this case, the material displayed in the Scrapbook window was created with a different application, MacPaint. Window manipulations are described in the Appendix, "Mouse Skills."

The Macintosh provides a set of "Desktop Accessories," each of which can be used to add another active window to a display. Normally an office desk might have on it a calculator, a digital clock, a notepad, and other

Figure I.2 Screen shared by MacWrite and the Scrapbook
accessories. The Mac provides simple programs, the Desktop Accessories, that replicate these functions (Figure I.3). The Desktop Accessories are available from the first menu, the Apple menu. For example, while you are writing with MacWrite, you can bring up the Calculator to make a quick computation needed for a memo. After you've calculated the answer, you can paste it into the text.

Some of the special features of the Macintosh become apparent only through use. Those who have used other personal computers will appreciate how much easier it is to get things done with the Mac. It is especially pleasant to find a user interface that remains consistent across a variety of applications programs. If you have never used another computer, you will probably wonder why so many people seem to make such a big deal about using personal computers. It's just not very hard with the Mac.

With the Macintosh, you can draw a picture on the screen using MacPaint.
and save it on a disk. Later, you can write a memo using the application program called MacWrite and add the saved drawing to the text. The memo, with its sketch, will appear on the screen just as it looks when printed out. With most other personal computers, there is no way to add a picture created with one program to text made with a different program. The user has to print them out separately—if there is any way to print the graphics—and then put the two printouts together with scissors and paste.

The Macintosh Clipboard makes it possible to move material created with one application program into another program. You can cut or copy material in the first application, which places a copy of the material on the Mac's Clipboard—a storage area reserved for passing information between applications. Then you can paste the contents of the Clipboard when using another application. The cut, copy, and paste features make the Mac an integrated environment, since its major applications can share data.

Most of the Macintosh's advanced user interface features are possible only because the Mac is a graphics computer. Most personal computers cannot easily mix high-resolution text and graphics. The Mac's ability to mix high-quality text and graphics on the screen makes it possible to mix icons—small pictures that represent standard objects and functions—and text labels. The graphics basis of the Mac also makes it possible to have effective windowing, and to let users mix their text and pictures.

It is the graphic quality of the Mac that makes it feasible also to produce text in so many fonts, sizes, and styles. Computers that don't produce text graphically are limited to displaying many fewer styles of text. Such computers require that users put special commands for the printer in the middle of ordinary-looking text. These "embedded formatting commands" permit the printing of some special text effects like boldface or italics, but these styles usually cannot be shown on the screen. The Macintosh's power to display what will be printed is unprecedented in personal computers.

The Macintosh's considerable graphics abilities extend to its printing. Coupled with an Imagewriter printer or a laser printer, it can reproduce all the text and graphics that the screen can display, and more. The Mac's printers are capable of greater resolution and much larger displays. The Macintosh is an especially good tool for preparing presentation-quality graphics. Pie and bar charts, tables, graphs, sketches, organization charts, project schedules, and many other materials that combine text and graphics can be readily created and easily edited with the Mac, using a variety of special-purpose applications. Many Mac applications are specifically designed to support generation of presentation graphics for business.
While the Apple Macintosh has a number of very powerful hardware features, it is not graphics and computational power alone that makes the Macintosh a valuable personal computer. It is the way that the power has been harnessed. The Macintosh is easy to learn to use. It is consistent in how it interacts with the user across applications. It gives the user great control over the appearance of documents, both on the screen and in printouts. Take advantage of the Mac's power by exploring its applications.

**USING MAC POWER**

The focus of this book is the applications that make the Macintosh useful. You can learn how to use major programs for word processing, spreadsheet analysis, project management, and graphics composition, among others.

If you have never used a Macintosh, begin by reading the Appendix, “Mouse Skills.” If you are still getting to know your Macintosh, start by reading Chapters 1–3 in order. They will introduce you to two major applications, MacPaint and MacWrite, and show you how to organize your disks effectively. At that point, you will have a good exposure to the practical elements of Macintosh use.

The remaining chapters deal with other important applications, some from Apple, others developed by major software companies. Chapter 4 describes Multiplan, a spreadsheet application. Chart, covered in Chapter 5 can be used to construct charts of many different types, from scatter plots to pie charts, based on sets of numbers. Chapter 6 covers ThinkTank128, an application that aids the planning process by providing a flexible, interactive outlining system. Filevision, described in Chapter 7, is a visual filing system; it lets the user combine drawings and data filing in new, creative ways. Chapter 8 deals with MacProject, a project scheduling and tracking system that lets you enter project schedule information by drawing it in and presents the information in charts and tables. Chapter 9 presents methods for transferring data among computers, including the Macintosh communications application, MacTerminal. Chapter 10 covers other applications for drawing and writing, including MacDraw and Word, as well as a number of tools designed to work with MacWrite and MacPaint. Macintosh Pascal, a major breakthrough in Pascal programming environments, is treated in Chapter 11. Chapter 12
discusses Macintosh applications with non-standard user interfaces, other programming languages, and probable future forms for the Mac.

If you are an experienced Macintosh user, you can read the chapters as they appeal to you. You might want to check out the tips on using MacPaint and MacWrite at the ends of Chapters 1 and 2, and a quick skimming of Chapter 3 could turn up something you haven’t yet noticed about disk management.

All the Mac’s strengths—the advanced 68000 processor, the fast high-resolution screen, the efficient ROM routines—are useless without applications. Learn to use major applications, such as the ones described in this book, and you can put the power of the Macintosh to work for you.
PICTURES

THE MAC'S GRAPHICS CAPABILITIES 12
MACPAINT'S FEATURES 14
TECHNIQUES FOR USING MACPAINT 15
TRANSFERRING MATERIAL BETWEEN DOCUMENTS 23
THE MACPRINT PRINT CATALOG FEATURE 27
SOME MACPAINT SHORTCUTS 27
PROBLEMS AND TECHNIQUES WITH MACPAINT 30
MACPAINT RESOURCES 38
THE MAC'S GRAPHICS CAPABILITIES

The Apple Macintosh has amazing graphics capabilities. Action graphics programs like *Through the Looking Glass*, for example, demonstrate the Mac's ability to portray realistically the rapid movement of complex objects in a three-dimensional world (Figure 1.1). One of the most impressive things about this demonstration of graphics power is that the program was not written in carefully tuned assembly code, but rather in Pascal, a language not ordinarily used for fast-action graphics game applications.

Although the Macintosh's graphics capabilities are well displayed by games, it is not primarily because of games that the Mac's graphics are important to most users. The Mac's graphics speed and resolution make it possible to experiment freely in the construction of visual presentations, such as charts, diagrams, maps, and plans. Graphics composition applications like MacPaint are so fast that you can try things out just to see what effect they will have without being concerned about the time it takes to experiment.

Because the Macintosh is a true graphics computer, you can not only use different fonts, styles, and sizes for text in word processing applications, you can also combine graphics with the text. Many different Macintosh applications have graphics capabilities. For example, in Apple's word processing application, MacWrite, you can paste graphics created with MacPaint into a memo. Using the Cairo font, you can add pictorial elements by typing. MacWrite provides graphics manipulation features such as moving and resizing graphic images in a document.

The applications that really let the Mac's graphics capabilities shine, however, are those specifically designed for the preparation of graphic displays. These applications include MacPaint and MacDraw from Apple, and Chart from Microsoft. The Telos Filevision application (see Chapter 7) has graphics construction capabilities mixed with data-base building features.

The Mac has wonderful potential for preparation of black-and-white presentation graphics. Two features make Macintosh graphics applications immensely useful. First, the Mac has unusually good graphics resolution and speed. Second, whatever you create can be printed on paper exactly as it appears on the screen.

The Mac's graphics capabilities are due, in large part, to a collection of very clever drawing routines called Quickdraw, found in the Macintosh ROM. Bill Atkinson of Apple wrote the Quickdraw routines and then followed
that feat by authoring MacPaint. In a sense, MacPaint is Quickdraw made directly available to the Mac user.

Presentation graphics are images that are meant to be shown to other people. Examples include graphs and other figures for inclusion in technical or financial reports, design sketches, and personalized birthday cards.

Graphics construction applications can be divided into two classes. Programs like MacPaint treat pictures as collections of dots on the screen, or pixels. Such applications are called pixel-oriented graphics programs. In other graphics programs, all the elements on the screen are represented in the Mac's memory not only as a collection of pixels but also as objects of types known to the program—this is a circle, that is a rectangle, and so on. Such graphics applications are known as object-oriented graphics systems. Examples include MacDraw and Filevision.
Object-oriented graphics applications have some interesting and useful characteristics. In MacDraw, for example, one object can be moved to overlay and totally obscure another without actually eliminating the overlayed object from the data base that underlies the picture. For example, a large circle could be placed so that it covered a small rectangle completely. Later, however, if the circle is moved aside, the small rectangle will appear in its original location.

There are some disadvantages to the object-oriented graphics systems on the Macintosh. One is that they don’t provide as many features for flexibility and detailed control as pixel-oriented applications do—features such as FatBits in MacPaint. The Mac’s object-oriented graphics systems, MacDraw and Filevision, work best for portraying pictures with regular objects and sharp boundaries, depicting worlds of regular features and straight edges.

MACPAINT’S FEATURES

Pixel-oriented graphics systems are more appropriate for depicting pictures with shadow, texture, and fuzzy edges. MacPaint is such a graphics application. Pixel-oriented graphics have the characteristic that once the graphics designer has made a commitment to placing an object such as an oval or a letter at a particular location on the screen, and has gone on to do something else, the image loses its object status from the application’s point of view. It is no longer an oval or a letter, so far as the application is concerned, but merely a collection of pixels (dots). This characteristic places some constraints on the extent to which a user can undo actions with MacPaint. For example, you can’t overlay five objects in MacPaint and then decide to remove the second and the fourth. MacPaint can’t tell which dots belong to the different objects you layered in the picture.

On the other hand, because MacPaint doesn’t restrict you to composing pictures from pre-defined objects, you can depict anything that can be depicted by placing dots in any of 414,720 locations. This is the number of pixels that completely fill a display 8 inches wide and 10 inches high, at 72 pixels per inch. The Macintosh has such a display—576 by 720 dots—its printed output.

MacPaint reflects Apple’s commitment to making the Mac a truly useful machine. If the MacPaint had been designed to produce graphics only on the screen, it would have been an interesting toy to play with, but not a serious
tool. Because MacPaint can be used to prepare pictures printed on paper, it can be used to produce technical drawings, sketches, charts, and other types of graphic figures.

A pixel-oriented display can be represented as a bitmap in computer memory. A bitmap is an area of memory in which each pixel is represented by a bit. If the pixel is white, the bit is 1, and if the pixel is black, the bit is 0. The 414,720 pixels of a full-printed MacPaint page require a lot of memory if the page is treated as a bitmap. If you are mathematically inclined (and if you know that there are 8 bits to a byte), you may have already calculated that such a pageful of pixels would occupy 51,840 bytes. Only seven such documents would fit on a single-sided 400K-byte Macintosh disk with nothing else on it—no system files or applications.

Fortunately, MacPaint is more clever than that. It stores the pixel map in a compressed form on the disk and expands only those portions that fit into the drawing area of the screen into a bitmap. For most drawings, the compression technique is very effective. Many simple MacPaint documents may take only 4K to 5K bytes on the disk. (A completely blank MacPaint picture takes 2,048 or 2K bytes.) Even complex documents rarely require more than 10K or 12K bytes.

TECHNIQUES FOR USING MACPAINT

By far the best way to learn the basic MacPaint techniques is to use the application to produce figures of the types that are of interest to you. Experiment with each of the MacPaint tools until you know what they can do. Dip into the lovely little MacPaint manual from time to time. It is quite compact, and most people find that they see something new when they return to it after working with MacPaint.

People interpret new experiences in terms of familiar ones. It is natural to think of MacPaint in terms of the familiar task of sketching with pencil or pen on paper. To an extent, it is possible to use MacPaint in this way. You can select the Paintbrush and draw freehand using the mouse, much as you would with a brush or with pencil on paper (Figure 1.2).

It would be a mistake to limit your use of MacPaint to this primitive analog of ordinary drawing, however. To make effective use of MacPaint, you must learn to use its special features. Indeed, were it not for these features,
only people with well-practiced artistic skills could use it effectively. The special features make it possible for people who don’t think of themselves as artists to produce impressive graphics designs and figures.

Many of the special MacPaint features are made available through its menus, especially the Edit and Goodies menus (Figure 1.3). Other features are exercised by using the mouse in combination with the Option and Command keys.

For one of my first MacPaint projects, I decided to make a Valentine’s Day card. I wanted to create a couple of cartoon-like dancing heart figures. My first thought was to draw a heart freehand, with unimpressive results (Figure 1.4). My problem, of course, was that I failed to take advantage of the special features of MacPaint. When an image has symmetry, you can turn on the Brush Mirrors option and just sketch one side. If the image is more complex, first draw and edit half of the object, make a copy, flip the copy across the axis of symmetry, and then position the copy appropriately.
### File
- New
- Open...
- Close
- Save
- Save As...
- Revert
- Print Draft
- Print Final
- Print Catalog
- Quit

### Edit
- Undo `%Z`
- Cut `%X`
- Copy `%C`
- Paste `%V`
- Clear
- Invert
- Fill
- Trace Edges `%E`
- Flip Horizontal
- Flip Vertical
- Rotate

### Goodies
- Grid
- FatBits
- Show Page
- Edit Pattern
- Brush Shape
- Brush Mirrors
- Introduction
- Short Cuts

### Font
- Seattle
- Los Angeles
- Cairo
- Chicago
- Geneva
- New York
- Monaco
- Venice
- Athens

### Font Size
- 9 point
- 10
- 12
- 14
- 18
- 24
- 36
- 48
- 72

### Style
- Plain `%P`
- Bold `%B`
- Italic `%I`
- Underline `%U`
- Outline `%O`
- Shadow `%S`
- Align Left `%L`
- Align Middle `%M`
- Align Right `%R`

---

**Figure 1.3** MacPaint's menus
To draw a valentine cartoon character taking advantage of MacPaint's features, I decided to create first the left half of a heart shape. I made a circle using the oval option from the Tools menu at the left of the MacPaint page. Holding down the Shift key constrained the tool to produce circles only. I dragged out a circle about an inch in diameter. Then I chose the straight-line tool and added a tangent vertical line. The vertical line was just to mark the destination for a freehand arc, which completed the left edge of a heart shape (see Figure 1.5).

I erased the vertical line and those portions of the circle that don't belong in the left half of a heart shape. I used the eraser from the graphics tool set to remove most of the extraneous lines. Then I used the FatBits option from the Goodies menu to clean up the dots I had failed to get with the eraser (Figure 1.6). The next step was to choose the selection rectangle and drag its box...
around the half heart. I then dragged the selected half heart to the right while holding down both the Shift and the Option keys. Because the option key was held down, the figure stayed in its original location, and only a copy was dragged off to the side (Figure 1.7). Because the shift key was held down, the dragging was strictly horizontal, with no slight skewing up or down.

The next step was to flip the half on the right. Since it was still selected—enclosed by the edit rectangle—I just pulled down the Edit menu and chose Flip Horizontal. To glue the new right half of the heart to the left half, I chose

Figure 1.7 Two identical shapes
the Lasso tool. I circled the right half with the lasso to select it. Then, holding down the shift key in order to restrict movement to the horizontal dimension, I dragged the right half to the left until it mated with the left half (Figure 1.8).

At this point I chose the Spilling Paint Can tool in order to fill the heart with black. I was happy with the shape of the image, but it was larger than I wanted. I enclosed the whole heart in the selection rectangle and held down the command key while dragging in order to shrink the heart to a smaller size (Figure 1.9). (You can also use this technique to expand objects.)

Now I added embellishments to turn the little heart into a cartoon character. I chose appropriate brushes to create the arms and legs. The feet were made by using the Fill-in Oval tool to create one small black oval, and copies of it were placed at the ends of the legs. The hand was built entirely in
the FatBits option, then moved into position at the end of the arm (Figure 1.10).

I made a second copy of the hand, flipped it horizontally, and attached it to the other arm. Then I added some highlighting to give a three-dimensional quality to the feet (see Figure 1.11). Using FatBits, I added highlights to the heart shape as well (Figure 1.12).

To make two facing heart cartoon characters, I dragged a copy of the whole figure to the right and flipped it horizontally. Then I lassoed the new figure and dragged it over to the first character until the hands overlapped. As before, I held down the shift key while dragging to keep the two figures on the same horizontal level. Selecting the text tool, I chose a font, size, and style and gave my little picture a title (Figure 1.13).

To build this picture of two valentine dancers, I repeatedly made use of the special features of copying, moving, and flipping. And I didn't have to stop there. This complex figure can be treated as a component to be manipulated
Figure 1.12 Adding highlights with FatBits

Valentine Dancers

Figure 1.13 Valentine dancers
Repeatedly and put into a different figure. For example, I made many copies of the pair of dancing heart figures, sizing each copy differently. (In order to preserve the proportions of an object you are changing the size of, be sure to hold down not only the Command key but also the Shift key as you drag the image to a smaller size.) Then, using the lasso, I partially overlapped the pairs of dancers to give an illusion of depth. Finally, I added a new banner at the top of the page. I'd produced a new and more complex picture in very little time (Figure 1.14).

To get the most out of MacPaint, look for ways to exploit its special features.

**Title Boxes**

Making shadowed title boxes like the one at the top of the picture of five cartoon heart couples (Figure 1.14) is a simple three-step process (see Figure 1.15). First select the Filled Rounded Rectangle tool and use it to drag out a black box, as in A. Then change the pattern from black to white and, using the same tool, drag out a partially overlapping white box, as in B. Finally, choose a font type, size, and style you like, and use the Text tool to add your title to the white box, as in C. Centering the title is easy if you choose the Align Middle option under style and then place the text insertion point in the middle of the white box. If you want to, you can fill the white box with a pattern from the Pattern menu.

**TRANSFERRING MATERIAL BETWEEN DOCUMENTS**

One of the nicest things about using the Macintosh is that its applications use consistent methods for doing similar things. A common requirement in working with almost any application is that of copying material from one document into another, or from one place in a document to another place in the same document. On the Mac, the usual way of doing this is with the Clipboard. The Clipboard is a magic place—sometimes all in memory and sometimes temporarily on the Mac’s disk—that stores things for pasting into documents.
Valentine Dancers

Figure 1.14 Receding valentine dancers
There are two ways to put something on the Clipboard, copying and cutting. When you Copy something to the Clipboard, the original remains where it was and a copy is put on the Clipboard. When you Cut, the cut material is removed from its original location and put on the Clipboard.

After you've put some material on the Clipboard, it will stay there until one of two things happens: either you put something else on the Clipboard with another Cut or Copy, or you turn your Mac off. To Paste the material on the Clipboard into a document, you simply select an insertion point—usually by pointing and clicking—and then give the Paste command. All three commands, Cut, Copy, and Paste, can either be chosen from the Edit menu or be invoked through keyboard commands.

To make sure you're familiar with the use of the Clipboard in MacPaint, create a new MacPaint document (see Figure 1.16). Use the brush or any of the other tools to create a little picture. Now use the selection rectangle to select the picture. Your screen should show your picture enclosed by the dotted lines
of the selection rectangle. Now **Copy** the selected area onto the Clipboard by typing Command-C. Unselect the picture in the selection rectangle by clicking elsewhere. Now type Command-V to **Paste** the Clipboard contents into your document. The newly pasted image appears in its own selection rectangle. Drag it to a location near the original and drop it.

Of course, within a single MacPaint document it is easier to use the Option key to make copies of selected areas of a document, so long as the two areas are in the same drawing window. If they are on widely separated parts of the full-page MacPaint document, the Clipboard method is best. The same copy-and-paste method can be used to copy material from one MacPaint picture to another. In fact, as Chapter 2 describes, you can even use the Clipboard to copy a graphic design into a MacWrite document.
THE MACPAINT PRINT CATALOG FEATURE

Even if you make a special system disk for working with MacPaint—one with only the necessary system folder documents, the empty folder, and the MacPaint application—you can keep only a small number of MacPaint documents on it. Pictures you aren't working on at the time should be kept on DATA DISKS, non-system disks on which you keep backup copies of your work.

A 400K-byte Macintosh disk can hold a great many documents. But how can you remember which pictures are on what disk? One way, of course, would be to keep a written record for each disk, adding to it or deleting as you add and delete documents from the disk. Unfortunately, most of us aren't that well-organized. Fortunately, there is an easier way. MacPaint lets you print out a catalog of the pictures you have on a disk (Figure 1.17). It would have been fairly easy to include a feature to print out the names of the documents, but MacPaint goes one better. After all, a name is not always as good a clue to the contents of a document as you hoped it would be when you thought it up. MacPaint's Print Catalog option—available in the File menu—actually prints a miniature version of each MacPaint picture on a disk.

SOME MACPAINT SHORTCUTS

Thanks to MacPaint's commitment to the design of the Macintosh user interface, the beginning user doesn't have to know any tricks to be able to use all of MacPaint's tools. Once you have a little basic experience with MacPaint, however, you'll find you can work faster and more easily by using the convenient shortcuts. Shortcuts give you ways of doing some of the same things you can do with the menus, but faster.

FatBits Shortcuts

FatBits gives you a close-up view of the pixels in a small area of the screen. You're likely to find yourself doing lots of work with FatBits, so it makes sense to use the fastest ways to get into and out of this mode. The Pencil is the most
MacPaint documents on disk AM backup

Allen's Dino  JP's Bronto  Two Vals  touchscreen  touchscreen2

cursors  Valentine  old Valentine  Keys  F1.12windw

Finger  Pam Loves  Hot Spots  window  ESAS EdWinds

ESAS AfEd  VCR wAfEd  VCR wCEd  VCR  NewValDence

Figure 1.17 A catalog of MacPaint documents
useful tool for FatBits work. After you have selected the Pencil, position it in the area where you want to do FatBits work, hold down the Command key, and click the mouse. You can get back to the regular edit page by again clicking with the Command key held down.

If you hold down the mouse button while drawing with the pencil, it will retain the color (black or white) that it starts drawing with—the opposite color of the dot it was pointing to when you first depressed the mouse button. You can draw straight lines—horizontal, vertical, and 45° diagonal—by holding down the Shift key while drawing with the Pencil.

When you work in FatBits, you usually want to do a good deal of scrolling—pushing the work area around under the FatBits window with the Grabber. (Note that the Grabber—the hand-shaped tool—plays much the same role in MacPaint that scroll bars do in other applications.) Unfortunately, it’s awkward to have to keep moving over to choose the Grabber tool, repositioning the window, then choosing the Pencil again. Fortunately, there is another shortcut. Just hold down the Option key in FatBits, and the mouse pointer turns into the Grabber. Move the window to where you want it, release the Option key, and you’ll get your pencil back.

**Show Page Shortcuts**

The quick way to get to the Show Page view of your document is to double-click the Grabber. From this global view of the picture you can do some very powerful things. By positioning the mouse pointer outside the current drawing window, you can drag the whole picture to a new position on the page. You can even drag some or all of the image off the page entirely.

This view of the entire page is also the most convenient way of making major changes in the location of the drawing window. To move to a different part of the document, put the pointer inside the window and drag it to the part of the page you want to work on next.

**Other Shortcuts**

Sometimes you want to do something to the whole graphics drawing window, such as place a copy on the Clipboard. To select the entire window, just double-click the dotted-line selection rectangle tool. The shimmering dotted
lines of the selection rectangle will appear at the edges of the window. At this point you can choose any of the operations that apply, such as copying to the Clipboard, flipping, or resizing by dragging with the Command key down.

To choose a new brush shape, just double-click on the Paintbrush tool. You'll be presented with the brush shape selections, just as though you had selected Brush Shape from the Goodies menu.

To erase the entire drawing window, double-click the Eraser. This shortcut has its dangerous side. Be sure not to double-click when you merely want to choose the eraser for a little selective erasing in the current window. Don't panic if it does happen by accident, however. Just choose Undo from the Edit menu. The shortcut for Undo in MacPaint is the ^key at the top left corner of the keyboard. Undo can take back only the last action you make. If you accidently erase the drawing window, for example, and then take any other action—such as clicking on one of the MacPaint tools—you will not be able to undo the erasure.

PROBLEMS AND TECHNIQUES WITH MACPAINT

MacPaint does have more serious uses than making valentines. Consider the simple chart of projected sales over time in Figure 1.18. Such charts can be quickly prepared. First select a tight grid pattern from the Pattern menu at the bottom of the screen. Using the Filled Rectangle tool, drag out the chart background using this grid pattern. Then type the words "Sales" and "Time" at the bottom of the grid. Use the select box to select the word "Sales," and choose the Rotate option from the Edit menu. Move the word (still in its select rectangle) to the left side of the grid. Now choose an appropriate brush size and shape. Choose the black pattern and draw the "Their Company" line. Then choose the all white pattern and use the brush to add the "Our Company" line. Finally, choose a smaller text size and use the text tool to add the line labels in the grid area.

MacPaint is not the best application for making sales charts. A much better choice would be Microsoft Chart, which is discussed in Chapter 4. Still, if all you want to do is create some visual presentations to convey a general idea, such as the one shown in Figure 1.18—"Our company’s sales will eventually surpass theirs"—then you can use simple MacPaint methods to do so.
You can also easily make technical drawings with MacPaint. In the example in Figure 1.19, the front panel of the Spurious, Inc. videocassette recorder is displayed. Many of the same techniques used to prepare the valentine dancers figure were also used to build this one. Look for evidence of copying, forced alignment (keeping items on the same horizontal or vertical level by holding down the Shift key while dragging), and grids.

Using the Grid

To get concentric circles, use the Grid (one of the options under Goodies, remember?). As you know, when you drag a circle—the Oval option with the shift key held down—you begin the drag at the corner of a square that would exactly contain the circle. When the invisible grid is turned on, you can move the mouse pointer only to grid intersections. Therefore, you can easily make concentric circles by starting and stopping circle drags at grid points that are on the same 45° diagonal line.

Using the Grid is also important for aligning objects and getting spacing right. Figure 1.20 was designed for a physics lab handout. It portrays the results of an experiment in which a strobe light flashes on two falling balls.
Figure 1.19 A simple technical drawing
Problems and Techniques with MacPaint

Elementary Mechanics, Experiment 1:
Two falling objects
(one with a horizontal velocity component)
illuminated by strobe light

Figure 1.20 Two falling objects illuminated by strobe

To begin with, the cross-hatched background was created by dragging this pattern with the Filled Rectangle tool. Then the legs were added. The ball was created in a clear space on the left, dragged onto the grid background with the Option key held down, so that a copy could be deposited whenever the mouse button was released. The Grid was used to increase the vertical distance between successive balls while keeping the horizontal distances between successive balls the same.

The Grid also makes it easy to create simple forms. The vertical and horizontal lines of the game form shown in Figure 1.21, adapted from a well-known board game, were made with the help of the Grid, which helped the form designer achieve equal spacing between the lines. The strange language on the form is the American Indian language Chickasaw.
<table>
<thead>
<tr>
<th>Hattak</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ishkoboka' Mastat</td>
</tr>
<tr>
<td>Professa' Takgloshi'</td>
</tr>
<tr>
<td>Mr. Okhamali'</td>
</tr>
<tr>
<td>Mrs. Chaloklow-Okhamali'</td>
</tr>
<tr>
<td>Miss Homma'</td>
</tr>
<tr>
<td>Mrs. Tohbi'</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Ishtabi'</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bashpo</td>
</tr>
<tr>
<td>Shoppala' Akmi' Aafokha'</td>
</tr>
<tr>
<td>Tanamposhkolo'li'</td>
</tr>
<tr>
<td>Ishtalakchi'</td>
</tr>
<tr>
<td>Naki' Bila' Cholok Aa'sha'</td>
</tr>
<tr>
<td>Naashtashana' Ishtiwwi'</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Aboha</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aboha Ittintakla'</td>
</tr>
<tr>
<td>Hattak Aabinoli'</td>
</tr>
<tr>
<td>Aaimpa' Aboha</td>
</tr>
<tr>
<td>Aaholhponi'</td>
</tr>
<tr>
<td>Aahilha' Aboha</td>
</tr>
<tr>
<td>Nannaaholokchi'</td>
</tr>
<tr>
<td>Aachokoshkomo' Aboha</td>
</tr>
<tr>
<td>Holissaa'sha'</td>
</tr>
<tr>
<td>Aboha Holissaanisa'</td>
</tr>
</tbody>
</table>

**Figure 1.21** A form created with MacPaint
Tiny Text

It is possible to make letters even smaller than the 9-point fonts. For example, note the VCR front panel in Figure 1.19. I entered the words in 12-point on a blank area of the drawing window. Then I put the block of text in a select box and shrunk it, dragging the box while holding the Command key down. Finally, I did a little cleanup with FatBits.

Recovering from Error

It is easy to make an error that you can't undo. I was dragging my only copy of a complex figure over a complex background when I accidently dropped it where I didn't want it. Then I clicked on the Eraser tool without thinking. If I hadn't started doing something else, I could have simply used the Undo option in the Edit menu.

When something like this happens to you, you may be able to recover. Did you have the figure on disk before you started the current editing session? Or did you save your MacPaint document after you created the complex figure, but not since you messed it up? If so, you can probably recover it. Just choose the Restore option from the File menu. MacPaint will replace your modified document with its last saved version. The restored version should be in the same state that you last saved it.

If you don't have an earlier saved version of the document, you're out of luck. At least you can take a lesson from the experience. It really pays to save different versions of complex MacPaint documents. As you make major modifications to a drawing, use the Save As option from the File menu to put a copy on the disk with a new name. You'll be glad you did if you can pick up the drawing at an intermediate stage rather than have to start from scratch.

Making Room for Documents

You will find that you can't make many complex MacPaint documents on your Write/Paint disk before MacPaint starts telling you that there isn't enough room on the disk. If you have only the built-in disk drive, you should probably make a special MacPaint disk, with just the system folder, the empty folder, and the MacPaint application, as explained in Chapter 3. This will give you a disk with 100K bytes of free space, enough area for a few MacPaint documents.
and the scratch area that MacPaint requires while it's working. MacPaint documents you are not currently working on should be kept on DATA DISKS—disks that have documents, but no applications or system files.

**Recovering from Disaster**

Some night your spouse may find that you are so deep in MacPaint that pulling the Mac's power cord from the socket is the only way to get your attention. If this happens, you may be able to recover some of your work. Start up your Mac again and have a look at your disk window (Figure 1.22). If you have a strange blank icon labeled “Paint 1,” simply open MacPaint directly (not with any of the other Paint documents). If it starts up with a document titled

![Figure 1.22 Directory with remnant of a crashed MacPaint](image-url)
“Rescue,” then MacPaint was able to reopen the temporary version of your document it was working on when so rudely interrupted.

MacPaint saves temporary documents like “Paint 1” when you move to another area of the picture with the grabber or from the Show Page option. If you haven’t done that, chances are the picture will be lost.

**Printing Many MacPaint Documents**

Let’s assume you are the kind of well-organized person who has put a set of related MacPaint documents into a folder in exactly the order in which you would like to print them. Now just open the folder and drag the selection box around them. Pull down the File menu and choose Print (Figure 1.23). After a brief delay, MacPaint is opened automatically and begins printing the documents you selected. The pictures will all be printed in MacPaint’s Print.
Final mode, which is darker and crisper than the Print Draft mode. The order of printing is the order of the documents in the folder, moving left to right and top to bottom. Convenient, isn’t it?

MACPAINT RESOURCES

Publication-Quality MacPaint Output

A company called George Graphics provides the service of printing very high fidelity versions of MacPaint documents. Two levels of quality are provided: 300 dots per inch from a laser printer, or 700 dots per inch from a high-resolution CRT typesetter. To take advantage of the service, all you have to provide is the Macintosh disk with a list of documents to be printed, and money. For further information, contact:

George Graphics
George Lithograph Company
650 Second Street
San Francisco, CA 94107

BOOKS ON MACPAINT

MacCats . . . 99 Ways to Paint a Cat with MacPaint. by Floyd Flanagan.

CHAPTER 2

WORDS

WORD PROCESSING WITH THE MAC 40
WORD PROCESSING APPLICATIONS—CHOICES 42
USING MACWRITE 43
TECHNIQUES FOR GETTING THE MOST FROM MACWRITE 51
For several years, word processing programs for personal computers have made the claim that what you see is what you get. This means that the screen display of a text looks very much like the printed version. Most computers before the Mac could not display text in more than one font, in more than one size, or in more than one or two styles, unless the computer used a special graphics mode that was too slow for word processing. Strictly adhering to the what-you-see claim would severely limit a word processing program’s printing capabilities on earlier technology computers.

Dot matrix printers and laser printers, on the other hand, can usually print text in several sizes—normal, condensed, expanded—and in several styles, at least underlined and boldface. Most advanced word processing programs for other computers compromise the what-you-see principle in some way. They utilize the printers’ capabilities by providing cryptic format control commands for printing in sizes and styles that can’t be displayed on the screen.

**WORD PROCESSING WITH THE MAC**

Word processing on the Macintosh is very different. With the Mac, you don’t have to put a funny command like “^PB” into your text to signify that you want to print a portion in boldface. Instead, you can see the styles and fonts you choose right on the screen, much as they will appear on paper. And the Mac gives you a great deal of control over the appearance of the text, with many fonts in five sizes and five special styles—bold, italic, underline, outline, and shadow—that can be combined with each other (Figure 2.1).

The Macintosh gives authors so much control over the appearance of text that writing with the Mac can be likened to writing with a computer-based typesetting machine. In fact, by the time this book appears on the bookshelves, programs and services will be available to use the Mac’s word processing documents as inputs to automatic typesetters. For those without access to typesetting machines, the new laser printers permit a new level of output excellence with the flexible font control of the Mac. Even Apple’s dot matrix printer, the Imagewriter, makes very impressive page layouts when the high-quality mode is chosen. For example, teachers have found that Apple’s dot matrix printer accepts ditto masters easily, and the finished dittos can display font and style variety they’ve never seen before. Book publishing may become a process accessible to a great many people.
This is New York font in 9-point.
This is Geneva font in 9-point.
This is Toronto font in 9-point. It is the most legible of the 9-point fonts.
This is Monaco font in 9-point.

This is New York font in 12-point. Bold Italic.
This is Geneva font in 12-point. Outline.
Toronto 12-point. Toronto is wider than other fonts.
Monaco font, 12-point. The non-proportional font.
This is Chicago font in 12-point. The system font.

This is New York in 14-point. Shadow.
Geneva font in 14-point. Underline.
Toronto font in 14-point. Bold.
Venice 14-point is the script font. Underline bold.

Toronto 18 point. Italic, Outline.
London 18-point is a good display font.
Athens 18-point is nice for headings. Bold, Outline.
San Francisco 18-point is a very silly font.

New York 24-point Bold Outline
Geneva 24-point Bold Outline
Toronto 24-point Bold Outline Underline

Figure 2.1 Some Macintosh font, style, and size options
The control that the Mac gives you over the appearance of your document goes far beyond the appearance of the text. You can include graphic materials mixed in with text simply by copying pictures or portions of pictures created with graphics applications. Best of all, you can see all the font selections, all the mixed text and graphics, every aspect of how your document looks, both on the screen and in your printouts.

On the Macintosh, word processing applications act like object-oriented graphics programs. MacPaint is a pixel-oriented graphics program with some text capabilities. Once you’ve gone more than a step beyond the action that places a character or a group of words on the screen, MacPaint doesn’t remember that the collection of dots that appear as a word are anything but a collection of dots, or pixels. In pixel-oriented applications, objects like letters and words lose their object status. In the MacWrite word processing application, letters appear as collections of dots on the screen or on paper, but they never lose their status as letters. Thus, you can always go back to a letter or any chunk of text larger than a letter and change its attributes—from plain to boldface or italic or whatever. This is possible in part because the application stores letters as characters in RAM and on disk and treats them as collections of pixels only for the purpose of presenting them to us on the screen or in printouts.

WORD PROCESSING APPLICATIONS—CHOICES

You have many choices of applications for producing text. If you are making a short document—say, a page—and you want a great deal of flexibility in composing the layout, you could even use MacPaint. Other applications that are not meant primarily as word processors can be used to produce printed texts. Program listings in languages like BASIC and MacPascal are typically produced by the programming application itself rather than with a separate editor/formatter. Even data-base and spreadsheet applications offer some (sometimes quite extensive) word processing capabilities.

To get the most complete facilities for writing and formatting texts, however, you will want to work with a word processor. The original Macintosh word processor, MacWrite, is very easy to learn to use and provides a great deal of control over the form of the text. It is most appropriate for
producing short documents such as memos. Word from Microsoft is the first Macintosh word processing application with design features that support the preparation of really long documents, and it can also be used for creating and printing form letters. For more on Word, see Chapter 10.

Other word processing programs will emerge, each with desirable features. Keep an eye on the reviews in magazines devoted to the Macintosh like MacWorld and The MACazine, as well as in more general-interest publications like InfoWorld, A+, and Byte. While both MacWrite and Word are very impressive applications, you would do well to keep track of emerging developments. The Macintosh may inspire new word processing systems, unlike anything ever seen before on a personal computer.

**USING MACWRITE**

Like Microsoft Word, MacWrite is a disk-based word processor. That means that, during the text editing process, portions of a MacWrite document will be immediately accessible in the Mac's internal Random Access Memory (RAM), while other portions may reside on disk. When you drag the scroll box to a portion of the document that is not currently in RAM, there is a brief pause while some of the document that is currently in RAM is saved to disk and the next portion is brought into RAM from the disk.

The first version of MacWrite for the Macintosh was a RAM-based word processor. This meant that it could produce documents only as long as would fit in RAM. On a Mac with 128K bytes of RAM—most of which is dedicated to crucial operating systems programs, to representing the video screen, or to the MacWrite program itself—RAM-based MacWrite could be used to make documents of only about 20K bytes. That was about eight to ten pages of single-spaced text, depending on margins, font, and font size. You may still hear some people who are not familiar with the current version of MacWrite claim that the Mac's word processing capabilities are suitable only for brief memos.

MacWrite is well suited for preparing memos, letters, and short reports, but it can also be used to write much longer documents. This book was written with MacWrite. When you write very long documents, it works best to put separate sections or chapters into different MacWrite documents. You can
then more easily keep track of revisions of different portions of a book or long report. Each MacWrite document that is to be part of a larger document should start at the top of a page, since the previous document always extends to the bottom of a page.

MacWrite Features

You may already be familiar with many of the editing features of MacWrite. If you have ever edited a document name in response to the prompt of a dialog box, or if you have changed the name of a document, folder, or disk in a directory window, then you have already used many of the editing features of MacWrite. All of the program code that lets you set the point for insertion with the mouse, that makes the Backspace key work to delete selected text, and that lets you cut, copy, and paste textual material is provided not by the MacWrite application itself but by a sophisticated part of the operating system dedicated to text-editing functions. This special operating system module, which is called CoreEdit, makes it easy for applications programmers to provide the basic text-editing functions in their applications. Because Apple makes this set of text-editing capabilities available to applications developers for free, most applications that have any text-editing capabilities implement them in the same way. This consistent approach makes it a lot easier for Mac owners to learn how to use new applications.

The Insertion Point. In a sense, MacWrite has two pointers (Figure 2.2). One is the mouse pointer. When the mouse is moved in the text area of your screen, you see the standard text pointer shape, the I-beam. The mouse pointer takes on its slanted arrow shape when it is moved outside the text area of the screen, into the scroll bar, title bar, menu bar, or a header or footer. The second pointer in MacWrite is a flashing vertical bar, the insertion point. This pointer doesn’t move as you move the mouse. Instead, it moves as you type. You can move the insertion point to any place you can point to with the mouse’s I-beam pointer. Just position the I-beam and click, and the insertion point will appear there.

One of the most common editing techniques in MacWrite is to change the insertion point by pointing to the destination for new text with the I-beam, clicking to set the insertion point, and then typing in the new text. Then you just move the insertion point to the next place you want to enter text. Of
MacWrite is well-suited for preparing memos, letters, and short reports. It can be used to produce longer reports, too, but this works out best if the report can be divided into several sections, each of which can start at the top of a page. You don’t have to add text from an insertion point. You can also delete backwards from the insertion point just by typing the Backspace key. Knowing how to type, how to set the insertion point, and how to backspace is enough to get you started editing with MacWrite.

**Dragging to Select.** Naturally, there are more elaborate ways to edit large chunks of text than insertion and backspacing. One of the nicer things about word processing is that it lets you pick up whole pieces of text and move them elsewhere. To do things like that, you first have to know how to select chunks of text. You already know how to select an insertion point—just point with the I-beam and click. The standard method for selecting a letter or a group of letters is to drag through the letters. If you want to select a couple of words on a line, you just position the I-beam at the beginning of the first word, depress the mouse button, and drag to the right. The text you drag over appears in inverse video—it appears as white text on a black background instead of the usual black on white (see Figure 2.3). When you get to the end of the text you want to select, let up the mouse button. (You can also drag a selection from right to left.)

**Things to Do with Selected Text.** There are lots of things you can do with selected text, the text that appears in inverse video. You can delete it by typing the Backspace key. You can replace it by typing whatever text you want to have replace it. When you strike the first key, the selected text disappears, the
Other ways to select. Dragging is only one way to select text in MacWrite. You can select a single word by pointing anywhere in the word with the l-beam pointer and double-clicking. The word will then appear in inverse video.

![Selected text in inverse video](image)

The following material collapses back to the preceding material, and the new text appears at an insertion point that takes the place of the replaced material. Any letters typed subsequently are entered at this point in the text until you relocate the insertion point.

Another thing you can do with selected text is place it on the Clipboard, that magic part of the Mac's memory that lets you transfer data between documents. After you've selected a block of text by dragging through it, you can simply type Command-C (or pull down the Edit menu and choose Cut) to put a copy of the selected text on the Clipboard. To type Command-C, you hold down the Command key, the one with the cloverleaf-like symbol, just to the left of the spacebar, while you type the letter C. (See the Appendix, "Mouse Skills," for more on the Command key shortcuts.)

Using the Clipboard. Until you get confident about using the Clipboard, you might want to check that the material really got copied. You can do this by choosing Show Clipboard from the Edit menu. The copied material is there, right? You can hide the Clipboard either by closing its window (click its Close Box) or by clicking on some visible part of your MacWrite window and thus putting the MacWrite window on top, concealing the Clipboard window below.

As in MacPaint, Copy is not the only way to put material on the Clipboard. If you want to remove the selected text from its current location, you can Cut it instead. After you've selected the text so it is in inverse video, just type Command-X to remove it and put it on the Clipboard. Or, if you prefer, you can pull down the Edit menu, drag to Cut, and release the mouse button.
Whether you use **Copy** or **Cut** to put material on the Clipboard, you can paste the new material with the **Paste** command, available either from the Edit menu or by typing Command-V. You can paste the Clipboard contents in the same document by scrolling to the new location, clicking an insertion point, and pasting there.

If you want to paste the material into a different MacWrite document, you can **Close** the current document and **Open** the one to which you want to add the material. So long as you don’t turn off the computer, or press the reset button, or put something else on the Clipboard, the material you copied to it will still be there when you go to work on the new document. Just select an insertion point for the material and do the paste.

You can even paste text copied from a MacWrite document into a MacPaint document. After you load the Clipboard with a Cut or Copy, you must quit the MacWrite application and open the MacPaint document you want to paste the Clipboard’s material into. When you do the paste in MacPaint, the text will appear in a selection rectangle, so you’ll be able to position it exactly where you want it. The transferred text will lose its special format attributes—boldface, italics, underlining, and so on—when it is cut and pasted between two applications.

**Other Ways to Select.** Dragging is only one way to select text in MacWrite. You can select a single word by pointing anywhere in the word with the I-beam pointer and double-clicking. The word will then appear in inverse video, ready to be cut, copied, deleted with backspace, or replaced by new text.

If you want to select a large body of text—more than appears on the screen—you can do it by dragging down (or up) through the text. When you get to the bottom (or top) window boundary, the text will just keep scrolling as long as you keep dragging. If you run out of table space, don’t forget that you can lift the mouse up without releasing the button, reposition it on your tabletop, and continue dragging.

An even easier way of selecting lots of contiguous text, however, is to use **Shift-Clicking.** Click the start of the area you want to select. Then use the scroll bar to get the end of the text you are selecting in the window. Position the I-beam at the end, hold down a Shift key, and click the mouse. Behold! All the text between the first click and the Shift-Click is now selected.

**The Two Ways to Delete.** Selected text can be deleted either by typing the Backspace key or by Cutting. From the author/editor’s viewpoint, the main
difference between the two is that a deletion done by cutting is Undoable, while material deleted with Backspace is soon gone forever. Backspace makes sense for deleting small amounts of text. It’s quick and easy. Cutting is preferable when large amounts of text are being deleted; it gives you a measure of fallback safety that most of us find we need from time to time.

You can recover backspaced text immediately after you do the back-spacing (before you backspace somewhere else, for example). If you type Command-Backspace repeatedly, the previously deleted characters will reappear one-by-one, starting with the most recently backspaced character.

**Rulers, Headers, and Footers.** MacWrite is a presentation-oriented word processor. This means that it provides many tools for preparing well-formatted printed texts. Rulers make it easy to set margins, tab stops, line spacing, and features like centering and full justification. You can change these features within a document by inserting a new ruler wherever you want a major format change. To insert a new ruler, simply use the Insert Ruler command from the format menu. A new ruler will be inserted at the last insertion point. Change whatever format features you want to modify for the new section on the new ruler. The following text will all reformat to meet the new specifications, up to the next ruler.

There is another way to change rulers. You can make a Copy of a ruler, such as the one at the beginning of your document, and then Paste it in at the point where the new format should start. This technique is particularly useful if you have several formats you want to switch back and forth between in your text. Keep a ruler for each of the formats you want to use at the beginning of your document. Then you can always move to the beginning, copy a ruler, and then paste the copy in a new location.

Headers and Footers let you specify how many blank lines you want at the top and bottom of every page, as well as giving you the option of specifying header or footer text. You can also put icons in the header or footer that will be printed out as the current date, time, and page number.

**Tabs.** You can move copies of the tab icon to appropriate locations on the ruler in order to set tab stops. As with a typewriter, when you type the Tab key, the insertion point will move to the next tab stop. Unlike a typewriter however, moving a tab on a ruler will affect the text that follows that ruler. Another way that MacWrite tabs are different from most typewriter tabs is that you can untab with one stroke of the backspace key. You don’t have to
type a number of backspaces to get back to the point at which you typed the tab.

**Fonts.** The number and variety of fonts, type sizes, and font styles really give MacWrite-formatted documents a special appearance. But the different font styles and sizes make it difficult to answer some quite ordinary questions your friends might ask about the computer. A conventional question about a business computer is “Does it have an 80-column screen?” If you use a 12-point Geneva font with wide margins on the Mac, the answer to this question is probably “Just about. It varies, but it’s around 78 columns.”

If you use 9-point Geneva, however, the answer would be much different. “Oh, no. The Mac displays over one hundred characters per line of text on the screen.”

Neither answer is the right response to the question. The number of characters per line on the Mac depends on the font and size and style of the letters on that line. Table 2.1 gives an approximate count of the number of characters per inch for a number of commonly used fonts in different sizes.

You can get font changes and style changes as you enter a text, or add the changes later when you edit the whole text. If you want to change fonts or font sizes as you enter text, you must pull down the choices available under the Font and Style menus to make the changes. If you want to change some aspect of the text such as boldface, underlining, italics, outline, or shadow, you can do so either from the Style menu or by the shortcut of typing the corresponding Command key combinations. The Command keys for the style options all make sense—B for bold, U for underline, I for italic, O for outline, S for shadow, and P for plain text. All the style options except Plain can be combined. If you want to write a phrase in bold italics, just type both Command-I and Command-B before typing the word.

When you are editing an existing text and want to change some portion to a different font, font size, or style, you should first select the text to be altered (by double-clicking a word or dragging or shift-clicking the relevant text). Then choose the new font (or size or style), and the entire selection will change. The selection is still selected at this point, so you can make other changes in it without reselecting. This is convenient if you want to change a long segment of plain text to be bold and italic, for example. Don’t type any ordinary text while the selection is in inverse video or it will all be replaced. Once you are happy with the changes you’ve made to the selection, click somewhere to deselect it.
### Table 2.1. Average number of characters per inch, different fonts

<table>
<thead>
<tr>
<th>Font</th>
<th>Size</th>
<th>Characters Per Inch</th>
</tr>
</thead>
<tbody>
<tr>
<td>New York</td>
<td>9 pt</td>
<td>17.0</td>
</tr>
<tr>
<td>New York</td>
<td>12 pt</td>
<td>13.1</td>
</tr>
<tr>
<td>New York</td>
<td>12 pt Bold</td>
<td>11.3</td>
</tr>
<tr>
<td>New York</td>
<td>14 pt</td>
<td>11.4</td>
</tr>
<tr>
<td>New York</td>
<td>18 pt</td>
<td>8.5</td>
</tr>
<tr>
<td>New York</td>
<td>24 pt</td>
<td>6.6</td>
</tr>
<tr>
<td>Geneva</td>
<td>9 pt</td>
<td>17.9</td>
</tr>
<tr>
<td>Geneva</td>
<td>12 pt</td>
<td>13.0</td>
</tr>
<tr>
<td>Geneva</td>
<td>14 pt</td>
<td>11.4</td>
</tr>
<tr>
<td>Geneva</td>
<td>18 pt</td>
<td>9.0</td>
</tr>
<tr>
<td>Geneva</td>
<td>24 pt</td>
<td>6.6</td>
</tr>
<tr>
<td>Toronto</td>
<td>12 pt</td>
<td>11.0</td>
</tr>
<tr>
<td>Chicago</td>
<td>12 pt</td>
<td>12.2</td>
</tr>
<tr>
<td>Monaco</td>
<td>12 pt</td>
<td>11.5</td>
</tr>
<tr>
<td>London</td>
<td>12 pt</td>
<td>13.5</td>
</tr>
<tr>
<td>London</td>
<td>18 pt</td>
<td>9.2</td>
</tr>
<tr>
<td>Athens</td>
<td>12 pt</td>
<td>14.1</td>
</tr>
<tr>
<td>Athens</td>
<td>18 pt</td>
<td>9.7</td>
</tr>
</tbody>
</table>

To elaborate the style of a selected word or phrase, you can type the Command keys appropriate for the style you want. The style commands are “toggles”—if you want to change a phrase that is both underlined and italicized to be italicized only, you can select it and then type Command-U for underline. Because the phrase is already underlined, the underlining will be removed, and the phrase will only be italicized.

**Special characters.** Another feature that distinguishes the Mac from other computers is the number of special characters it can print on the screen. All kinds of special symbols (like £, ™, ©, §, ℘, and ℓ) and foreign language and mathematical characters (such as β, γ and Å) are available from the Mac.
keyboard in many fonts. You can type these symbols by holding down one of
the Option keys, or sometimes both Shift and Option.

Certain of the Option-plus-key combinations seem at first to produce no
effect on the screen. If you type Option-u, for example, nothing appears on
the screen, and the insertion point does not advance. But if you then type
another key, such as the vowel a, you see the German a-umlaut, ä. A key
combination like Option-u followed by another key is called an option-key
combination. This is a key that has no effect when first typed but, when
followed by one of a certain set of other keys, produces an accented character.
Another example is that Option-n followed by n produces ñ.

The Macintosh fonts have room for new characters. If you pull down the
Apple menu and select Key Caps, you can see what letters are provided. Hold
down the Shift key to see the upper-case characters. Hold down the Option
key and you will see many special characters. If you hold down both the
Option and the Shift keys, you will see that many characters are blank. They
appear as the symbol □. These blank characters present an opportunity for
adding new characters to fonts. New fonts and expanded fonts are available to
Macintosh users with special requirements. Special fonts for mathematicians,
for linguists, and for writing in languages that use different alphabets are some
of the optional fonts available.

TECHNIQUES FOR GETTING THE MOST
FROM MACWRITE

One-Drive MacWrite

If you are using MacWrite with one disk drive, you should make a special
working disk to create enough room. As is always the case in working with
computers, the amount of storage you seem to need is often just a bit more than
what you have available. You need lots of free disk space to work with
MacWrite because of the way it handles printing. When you request that a
MacWrite document be printed out, there is a delay in which you are told that
the Mac is “Now saving printed copy to disk.” MacWrite builds a temporary
document full of codes that instruct the printer on how to print the document.
The temporary Print version is a good deal larger than the original, so you must
always have more free space on your disk than the largest MacWrite document you might choose to print out. If there isn’t enough space for the print file, you will get the message “The disk is full. Please try again.” Don’t bother trying again until you have removed some documents from the disk. You can keep MacWrite documents that you are not currently editing on data disks.

Your one-drive MacWrite disk should contain the System Folder, so that you can boot with the disk, the Empty Folder, and MacWrite. When you are first putting the disk together, you should put Font Mover on it. Then open Font Mover and use it to remove any fonts you know you’ll never use. If you aren’t planning on creating the look of a handcrafted ransom note, you can probably do without San Francisco, for example. You may also decide that you won’t ordinarily require London, Los Angeles, or Toronto. In any case, you can copy the fonts you remove to a backup disk, using Font Mover, so that you can easily restore them if you ever must. When you’ve finished removing fonts, toss Font Mover in the trash to free up its space as well. Do this on your working copy only, not on the original disk! That should be stored in a safe place as a complete backup.

Getting rid of unnecessary fonts will give you a lot more space on your boot disk—the one you start your Mac with. A word of warning, though: Don’t remove the large versions of fonts you plan on using for high-quality printed output. You may be tempted to get rid of 18- and 24-point fonts if you plan on using only 9-, 12-, and 14-point in your documents. This would be a mistake if you plan on getting high-quality printouts. MacWrite very cleverly uses the 18-point font, squeezed down to 9-point size, to make a crisper printed 9-point printed text. Similarly, the 24-point information is used to construct high-quality 12-point printout. This technique exploits the fact that the Mac’s printers are capable of higher resolution—more dots per inch—than is its screen. If you remove the larger sizes of a font, then MacWrite will just “double-up” the information in the smaller font sizes to create the high-quality printout, with a less impressive effect.

After you’ve thrown out some fonts, the Font Mover, and MacPaint, you should have a writing disk with 100K bytes or more free space. This is enough room for several related documents you are currently working on, still leaving some free space for the temporary print file that MacPaint will create when you ask it to print out one of your texts.

You have enough space on the disk that you can have a number of text documents on the disk. When you complete a document and don’t plan on working on it again immediately, you should put copies on your backup disks
(see Chapter 3). When you have several related documents on a disk, it makes sense to organize them by putting them into a folder with a name that identifies their class.

To make a new folder, make a duplicate of the empty folder. Select the empty folder, then pull down the File menu and select **Duplicate** or, more simply, type Command-D. You’ll then see a copy of the Empty Folder, named, appropriately, Copy of Empty Folder. This folder is currently selected, so you can simply type in a new folder name appropriate for the topic of the related MacWrite documents. Then drag the documents that belong together into the folder. Storing documents by folders makes it much easier to find them later than if they are lying around all over the desktop.

**Dummy Text Documents**

One way you can prepare to work most productively with MacWrite is by setting up dummy documents. It takes a few minutes to set up the header and footer of a document. If you always use one set of headers and footers for letters, it seems a shame to repeat this effort every time you prepare a new letter. Perhaps you would like all your letters to include a graphic letterhead (Figure 2.4) that you built with MacPaint and to have page numbers at the bottom of the page. Instead of rebuilding the required header and footer every time you start a new letter, edit a dummy text document that already has that header and footer built in.

The first step in making your own “stationery” on disk is to construct the material you want to appear in the MacWrite header. If you want graphic elements in the display, do it in MacPaint. Then **Copy** the graphic to the Clipboard. Leave MacPaint, open MacWrite with a new file, pull down the format menu, and choose **Show Header**. Select the start of the header—position the mouse pointer there and click—and **Paste** the contents of the clipboard there.

If you are willing to settle for an ordinary text letterhead (Figure 2.5), you can just type in the material you want in the font, style, and size you want. You can choose to include the date, the time, and page numbers in the header or footer.

Once you’ve put together your header and footer in the style you like, adjust the top text ruler to the margin and tab settings, line spacing, and justification style you prefer. Finally, choose the font you want to use for the
Figure 2.4 A graphic letterhead

text of your letters by setting the insertion point at the beginning of the letter and then choosing your preferred font, style, and size. DON'T ACTUALLY TYPE THE LETTER YET.

Use Save As to store this almost empty document. You might choose to call it Letterhead or Empty Memo. Later, when you want to write a real letter, you can simply edit a copy of this one. One way to do this is simply to edit your Empty Memo, but do a Save As to store the real memo or letter with a different name.

The problem with deciding to do a Save As with a different name is that it's very easy to accidentally do a Save before you've given the document a different name. A better technique is to make a copy of the Empty Memo or Letterhead document in the disk directory window, position the copy where you want it, change its name from "Copy of Empty Memo" to "Memo to Alice," or whatever you want to call it, and then edit that copy.
You don’t have to restrict your use of the almost-empty document trick to letter and memo formats, of course. If you are writing a series of documents that should all have the same format—the chapters of a report, perhaps—then you should use an “Empty Chapter” document as the basis for each of the documents that make up the report.

Using such dummy documents will help you maintain consistent formats for documents of the same type. They should also save you some time setting up appropriate document formats.

**Printing**

Printing single documents with MacWrite is simple. If the document is open—that is, if you are editing it with MacWrite—you just pull down the File menu and choose the **Print** option (see Figure 2.6). If you want to print a document
report, perhaps — then you should use an "Empty Chapter" document as the basis for each of the documents that make up the report.

Using such dummy documents will help you to maintain consistent formats for documents of the same type. They should also save you some time setting up appropriate document formats.

**Printing**

Printing single documents with MacWrite is simple. If the document is open — that is, if you are editing it with MacWrite — you just pull down the File menu and choose the **Print** option. If you want to print a document from a disk directory window, just select the document and then choose **Print** from the File menu.

![Figure 2.6 The MacWrite printing options](image)

from a disk directory window, just select the document and then choose **Print** from the **File** menu.

A dialog box presents all the print options you have control over, including print quality, number of copies, and printing only part of the document. The initial release of MacWrite had a bug in the partial printout feature. If you tried to print out a document with page numbers starting at, say, 20, you could not ask for the page range 20 to 22 to print out only the first three pages of the document. You had to ask for pages 1 to 3. Not even Apple Macintosh software writers are perfect, but this bug was repaired in the disk-based version of MacWrite.

**Long Texts**

Suppose you have written a report that consists of three MacWrite documents: Report 1, Report 2, and Report 3. In order to make them print out with the correct page numbers, you must use the **Set Page #** option (in the **Format**
Techniques for Getting the Most from MacWrite

menu) while editing the second and third documents. The page numbers for the first one will begin with 1 automatically. When you finish your final editing of Report 1, note the last page number. Then, when you edit Report 2, use the Format option Set Page # to start the page numbers for this document with the next number. Follow the same procedure to make Report 3’s page numbers follow appropriately from Report 2’s.

You should keep closely related documents in an appropriately labeled folder. If the three Report documents are in a folder called My Report, it is very easy to print all three in sequence. When you put the documents into the folder, arrange them in order, with the first document at the top left of the folder window, the next document to the right of that one, and so on. When you want to print them in sequence, you can simply drag a select box around them all. All the selected documents can now be printed in sequence—left to right in the window—by choosing Print from the File menu. That’s not so hard.

At some point you may find that you have to use MacWrite to produce a long text in which you can’t leave blank space at the end of the page that is the end of the first MacWrite document. You’ll wish, at this point, that you had chosen a different Macintosh word processor for this project. Have no fear, though. You can make it work. First create your long text with a series of MacWrite documents. Get the content, fonts, size, and style of each document just right. Don’t make any of the documents the maximum possible length; leave a little room for moving lines between documents.

Once you’ve settled on the final form of your text, you can start moving material between MacWrite documents to make the page boundaries come out right. If the last page of your first document is a partial page (as it almost always will be), then cut it entirely. Save and Close that document and Open its successor. Paste the material you cut from the end of the first document at the beginning of the second. Then go to the end of the second document, cut the partial page there, Save and Close, and Paste it at the beginning of the third document. As you move from document to document in this way, use the Set Page # option to give the MacWrite documents the appropriate sequential page numbers.

This process must be complicated a bit if you are using full justification—both right and left margins are even. If your removal of the last page of a document cuts a paragraph, then the last line of the page before will look funny. The last line will no longer be justified to the right margin, since, so far as MacWrite is concerned, the paragraph now ends there. The only solution is to retype the last word of the paragraph. It will wrap around to make a new last page of one word. Make sure that the next MacWrite document starts with
the same page number. After you've printed the complete set of MacWrite
documents, you can go through them tossing out all the one-word pages you
had to have to make the justification come out right. Whew! Thanks to
MacWrite, you won't have to do this often.

Accidental Text Selections

People sometimes accidentally select text they don't want. The secret is just to
take it in stride. Suppose I want to select the word “difficult” in the text below,
perhaps because I want to replace it with “awkward.” I position the mouse
pointer (the I-beam) at the start of the word and begin dragging to the right.
At first the drag selection is going OK (see Figure 2.7). My hand moves up a
bit, and all of a sudden a lot of text on the line above is in inverse video (Figure

![Figure 2.7 Start of a text selection drag](image)

Q: I'm new to the Mac, and selecting text with the mouse still seems
difficult to me. Do other people never accidentally select text they
don't want?
A: It happens to me all the time. The secret is just to take it in stride.
Suppose I want to select the word “difficult” in the text below,
perhaps because I want to replace it with “awkward.” I position the
mouse pointer (the I-beam) at the start of the word and begin
dragging to the right. At first the drag selection is going OK.

Figure 5.7
Start of a Text Selection Drag

My hand moves up a bit inadvertently, and all of a sudden a lot of text
on the line above is in inverse video. It's no problem, though. I just

Figure 5.72
An Error in Dragging Text
Techniques for Getting the Most from MacWrite 59

Q: I'm new to the Mac, and selecting text with the mouse still seems difficult to me. Do other people never accidentally select text they don't want?

A: It happens to me all the time. The secret is just to take it in stride. Suppose I want to select the word "difficult" in the text below, perhaps because I want to replace it with "awkward." I position the mouse pointer (the I-beam) at the start of the word and begin dragging to the right. At first the drag selection is going OK.

My hand moves up a bit inadvertently, and all of a sudden a lot of text on the line above is in inverse video. It's no problem, though. I just move the mouse pointer back down and to the right a bit, and the word I want to select is in inverse video. Only then do I release the mouse button.

Of course, it sometimes happens that I accidentally make a selection I don't want. Then I just select over again.

People who find it difficult to select single letters by dragging with the mouse can just click after the letter, backspace to delete it, and then type the new material they want to replace it. That can be pretty fast too.

Disasters

If someone thoughtlessly turns off your Mac before you do a Save, there is no way to recover your work without a time machine. Unlike MacPaint, MacWrite doesn't use a temporary disk file to store your document in while
<table>
<thead>
<tr>
<th>File</th>
<th>Edit</th>
<th>Search</th>
</tr>
</thead>
<tbody>
<tr>
<td>New</td>
<td>Undo Copy %Z</td>
<td>Find...</td>
</tr>
<tr>
<td>Open...</td>
<td>Cut %H</td>
<td>Find Next %F</td>
</tr>
<tr>
<td>Close</td>
<td>Copy %C</td>
<td>Change...</td>
</tr>
<tr>
<td>Save</td>
<td>Paste %V</td>
<td>Goto Page %G</td>
</tr>
<tr>
<td>Save As...</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Page Setup</td>
<td>Show Clipboard</td>
<td></td>
</tr>
<tr>
<td>Print...</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Quit</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Format</th>
<th>Font</th>
<th>Style</th>
</tr>
</thead>
<tbody>
<tr>
<td>Insert Ruler</td>
<td>Seattle</td>
<td>Plain Text %P</td>
</tr>
<tr>
<td>Hide Rulers</td>
<td>Los Angeles</td>
<td>Bold %B</td>
</tr>
<tr>
<td>Open Header</td>
<td>Cairo</td>
<td>Italic %I</td>
</tr>
<tr>
<td>Open Footer</td>
<td>Chicago</td>
<td>Underline %U</td>
</tr>
<tr>
<td>Display Headers</td>
<td>Geneva</td>
<td>Outline %O</td>
</tr>
<tr>
<td>Display Footers</td>
<td>New York</td>
<td>Shadow %S</td>
</tr>
<tr>
<td>Set Page #...</td>
<td>Monaco</td>
<td>Superscript %H</td>
</tr>
<tr>
<td>Insert Page Break</td>
<td>Venice</td>
<td>Subscript %L</td>
</tr>
<tr>
<td>Title Page</td>
<td>Athens</td>
<td>9 Point</td>
</tr>
</tbody>
</table>

Figure 2.9 The MacWrite menus
you work on it. Your whole document is in the Mac’s RAM—Random Access Memory. That’s one reason you can go so quickly from the beginning to the end of a MacWrite document, with no disk accesses. If you turn your Mac off without saving, no trace of the document exists.

If you did routine Saves from time to time while editing, then the last saved version is still out there. If you didn’t, you will probably remember to do so next time!

**Keyboard Commands**

Figure 2.9 presents the menus of MacWrite. The menu options displayed with trailing command symbol-letter combinations can be chosen without using the mouse. You can issue these commands from the keyboard by holding down the command key and typing the letter. Learning to use these shortcut commands will increase your fluency with MacWrite.
CHAPTER 3

MANAGING YOUR MACINTOSH DISKS

ABOUT THE MAC'S DISKS 63
THE DISK DIRECTORY 64
DIFFERENT VIEWS OF A DIRECTORY 68
THE IMPORTANCE OF BACKING UP 69
ONE-DRIVE DISK COPYING 71
TWO-DRIVE DISK COPYING 72
BACKING UP YOUR OWN WORK 72
WHEN THE PARTY'S OVER 77
A data-base management system would be worthless if there were no way to store information for later retrieval. Every application for productivity—word processing, spreadsheet systems, project managers, whatever—would be of no use if documents could not be stored and then later retrieved and modified.

When you use almost any application program on the Macintosh, you enter new information (or edit old) that is then stored on a disk. Your goal in using the application is usually not just to store information, however. You want a printed document or the display of information on the screen. Storage of information on a disk is a necessary intermediate goal, particularly if you plan on using the information again.

ABOUT THE MAC’S DISKS

The built-in disk drive on the Macintosh was designed by Sony to meet special requirements established by Apple that ensure reliable storage of large amounts of information. Encased in the rigid plastic shell is a flexible plastic disk about 3½ inches in diameter. Although there is nothing floppy about the protective case, the disk inside is very similar to 8-inch floppy disks and 5½-inch mini-floppy disks. They are therefore often called MICRO-FLOPPY DISKS.

One of the most important things the Mac operating system is responsible for is managing the use of the disk. It provides all the techniques that applications need to store or write information to the disk, along with all the information needed to retrieve or read information from the disk. Computers cannot work quickly and directly with information stored on disk. They must first read the information into Random Access Memory (RAM), where it can be readily manipulated. The altered information can then be written back on the disk, either replacing the original version or as a new set of information.

If a computer is turned off before altered information is stored on the disk, only the original version will be available. This is the reason that computer users learn to save their work regularly. It exists only as an ephemeral state in RAM until it is saved to disk.

The information is stored on a disk in FILES. A file is a set of data that are stored together on disk, something like the way a file in a filing cabinet holds a set of related data. A disk file is a set of information elements such as characters (letters), numbers, or machine instructions. MacWrite and MacPaint docu-
ments are files, as are the MacWrite and MacPaint applications themselves. In fact, every application, document, and set of data on the disk is stored in one or more files.

THE DISK DIRECTORY

A special type of file, the computer's Disk Directory, has information about the disk—how much it can hold, when it was initialized—and a list of all the files on the disk. Directories contain not only the names of the files on disks but also information about where the files are physically located. See Figure 3.1.
When an application directs the Macintosh to read from a particular file, the disk operating system uses the directory to find out where that file is. As you might imagine, if anything happened to ruin a disk directory it would ordinarily make the entire disk useless, since the operating system would not be able to find any of the disk's files. The operating system Apple developed for the Sony disk drives cleverly puts into each block on the disk information about what file the block belongs to. This makes possible the development of utility programs for reconstructing the directories of damaged disks. The level of care shown in the design of the Mac's disk operating system is a reflection of the Macintosh development team's commitment to producing a very reliable computer system.

In fact, the Mac's ROM (Read-Only Memory) already has a hidden utility for reconstructing certain types of bad disks. Suppose that some day you insert a previously good disk and your Mac spits it back out as bad. If the disk was to be your boot disk—a system disk that is used when you start up or reset the Mac—then first press the reset button or turn the machine off and then on again. Then hold down the Command key and an Option key at the same time and reinset the problem disk. Keep the two keys depressed until the drive stops whirring. This process can repair certain types of damaged directories. Any folders you had on the disk will disappear, but the documents and applications should all still be present, and the disk should once again be usable. You can then recreate the folders by making and renaming copies of the Empty Folder and putting the files in them.

When you put a disk into a Macintosh disk drive, the Mac reads information from the disk directory into RAM. This makes it possible to get other files from the disk and lets the Mac know where the unused portions of the disk are, so that it can store new information on the disk. One sign that the Mac has read a disk's directory is the appearance of a disk icon on the desktop. When you open the disk, either by selecting it with a click and choosing Open from the File menu, or by double-clicking it, you open a disk window that reflects information in the disk's directory (Figure 3.2).

The directory window doesn't actually show you everything in a disk directory. In fact, it doesn't even list all the files on the disk. Files that are not meant to be directly accessed by users aren't visible in directory windows. One example of such invisible files is the font files that are stored on the disk. To see what font files you have on a disk, you must use the Font Mover application.

A Macintosh disk directory window displays the contents of the disk—
applications, documents, system files, and folders. **Applications** are programs that you can use, such as MacWrite and MacPaint and the Font Mover. **Documents** are the data files you create through the use of applications. One of the clever aspects of the standard Macintosh user interface is that documents know what their applications are. This means that you can simply open a document and that its application (if it is present) will automatically be loaded to let you work on the document.

**System files** are special data and code files that the operating system needs to make the Macintosh as easy to use as it is. Applications use parts of the system files routinely, but you don’t have to worry about that. Just don’t try to remove system files from the disk you use to start your Mac with—the Mac won’t work without them.

**Folders** provide a very useful way of organizing a disk. Just as related documents in a file cabinet may be grouped in labeled folders, you can use Macintosh folders to group related documents. You can even use folders to create hierarchical groupings by putting some folders inside other folders (see Figure 3.3).

In the disk directory window in Figure 3.3, there is a folder called “Using Mac-1” that represents the first part of this book. That folder is open, revealing several other folders, one for each of a number of chapters. Two of those folders are also open, for Chapters 3 and 4. Within each chapter folder are the MacWrite text documents that make up the chapter.
Folders can be opened to create folder windows in the same way that disks can be opened to create disk windows. You can double-click the desired folder, or you can select a folder with a single click, and then choose the Open command from the File menu.

Folders aren't documents themselves. They act more like directories, subdirectories of the main disk directory. Using folders makes it a lot easier to keep track of your work, especially on backup disks, where you store copies of many different documents. The disk window in Figure 3.4 represents a disk with twenty-four documents on it. Seeing all these documents at once is a bit confusing, and understanding their relationships to each other requires paying careful attention to their names. When they are sensibly organized into folders, it is much easier to find what you want (Figure 3.5).
DIFFERENT VIEWS OF A DIRECTORY

When you first booted your Macintosh system, it represented files as icons in your disk directory window. If you feel more comfortable with lists of files with no little pictures, you can choose to have files displayed as names, ordered alphabetically or by date, size, or kind (Figure 3.6). Simply pull down the View menu to the choice you want. People with prior computer experience sometimes feel that Real Computer Users Don’t Use Icons. Before you set about making your Macintosh display files in the same way that old-fashioned computers do, try working with the icon representations for a while.

Supposedly the two halves of the human brain have significantly different information-processing abilities. In most people, the left half of the brain is particularly adept at understanding and working with words and numbers. The right half is the one that is skilled at certain pictorial and spatial tasks.
Perhaps using the icon representation of files lets both halves of the brain cooperate in the task of recognizing your documents and other files. I’ve found that I find the documents I want to work on in a directory window much faster when I’m using the icon representation than when I view the directory in some other way. Try more than one method yourself, but give the icon view a chance.

THE IMPORTANCE OF BACKING UP

The documents you create using Macintosh applications are stored on your disks as data files. The hard-shelled disks used in the Mac are extremely reliable. It is much less likely that anything will happen to data stored on such a
 disk than if the Mac used ordinary floppy disks. On the other hand, there are ways for disaster to strike even the reliable Mac floppies.

Mark King’s three-year-old son wanted to see whether one of his dad’s Macintosh disks would make a good raft for the action figures he played with in the bathtub. It didn’t. It also made a very bad data disk afterwards, even when it had dried out. King had used his Mac to create the entire technical manual for Spurious Widget’s newest product. Because it was an urgent job, he had worked on it using the Macintosh at work and, in the evening, on his home Mac. It had taken him four weeks to write, and the only copy had been on the disk/raft that wouldn’t float. Mark wasn’t very happy, and neither were his bosses, since his manual was supposed to go to press the day after the disaster. The story doesn’t have a happy ending, except that Mark learned how to make backup copies of his work.

Two months later, Mark’s sixth-grade daughter demonstrated her science project on electromagnetism while Mark sat at his Mac in his home office. She put a giant electromagnet on his desk next to his box of working disks and turned it on. Disks store information in the form of many tightly packed magnetic fields, and they can easily become very mixed up when exposed to
strong magnetic forces. All the files on the disks closest to the magnet were destroyed.

Fortunately, Mark had learned his lesson after the earlier ruined-disk catastrophe. He had recent backup copies of all his work on a nearby shelf, as well as a slightly older set of backup disks at his office at Spurious Widget.

Because the Mac's micro-floppy disks are encased in a tough plastic shell, they are a good deal less subject to catastrophe than conventional floppy disks. Still, you should protect them from liquids, excessive heat, and strong magnetic fields. Try not to spill coffee on them or drop them in the dishpan. Don't leave them on the dashboard of your car in the sun. Avoid placing them on televisions or other devices that generate strong magnetic fields. And be prepared for disaster despite your precautions.

Take Mark's experiences as a warning. Learn now how to make copies of your work, and do so in a regular and organized fashion. There are several different kinds of backup disks you should make. First, you should have a backup of any application disks. If two disks are provided in the application package, consider keeping one some distance from your machine—in a building other than the one in which you do most of your Macintosh work, if possible. If a plumbing leak were to occur directly over the area where you kept your disks, you would be very pleased to have a backup somewhere else.

**ONE-DRIVE DISK COPYING**

Although it is possible to copy a disk by dragging its icon into another disk's icon, the process is time consuming on a Macintosh without an external drive because it requires that you exchange disks many times. Instead, use the Disk Copy utility application provided on your system disk. By using every available byte, it lets you copy a disk with only four disk swaps on a 128K Mac.

Adding a second disk drive to your Macintosh is one of the best ways to encourage yourself to back up your disks. Having the external drive increases the usefulness of many applications programs, as well. When you are using applications that are not limited to available space in RAM, such as Microsoft Word (described in Chapter 10) or Telos Filevision (see Chapter 7), you can create much larger documents when you use a second disk drive. You can also deal with larger sets of documents when you use an external Macintosh disk
drive. For example, you can select a larger group of MacWrite documents for sequential printing.

TWO-DRIVE DISK COPYING

It is very easy to copy a complete disk if you have a Macintosh with two drives. Boot with the application disk in the built-in drive and a blank disk in the second drive. A dialog box will prompt you through the actions required to initialize and name the blank disk. At this point, icons for both the application disk and the blank backup disk are on the screen. Just drag the icon of the original disk to the icon of the backup, and your Mac will take care of the rest. When the copying process is complete, select the backup disk and eject it. Put it in a safe place. Don't forget to label it first. That will make finding it much easier, should you need it later.

Many commercial applications are not fully copyable. Some applications can be copied, but require that the original disk be inserted once whenever you start up a copy of the application. Other applications programs seem to be copyable, but the copies do not work at all. Copyable software is a great convenience to the user. If you have purchased an application that can be copied, don't make copies of it for others. Distributing commercial software steals from the developers and discourages the production of copyable software.

BACKING UP YOUR OWN WORK

Making whole disk copies is not the usual way of doing backups. You ordinarily work with only a few documents at a time, and your backups should be done on a few-files-at-a-time basis. A good rule of thumb is to back up your work any time you would feel more than merely annoyed to lose your most recent additions or changes.

It would not make sense to back up a whole disk every time you change one or two files on it, and it usually takes less time to back up a file or two than a
whole disk, particularly if you are using a second disk drive. You may find it convenient to organize your documents into folders (see the Appendix, "Mouse Skills") and then back up by folders.

For most Mac users, it makes sense to have three kinds of disks: backup application disks, working application disks, and document storage disks. **Backup application disks** are the copies of your application disks, which you keep in a safe place. **Working application disks** are disks that have system files and applications. In some cases, these disks have to be modified from the form of the original application distribution disks for you to get the most from them. **Document storage disks** are disks without system or application files, where you store the documents you create with applications.

A 400K-byte Macintosh micro-floppy disk has enough room for essential system files, an application, and (very roughly) 100K bytes for document space. For most applications this is a reasonable amount of working space, even though some of it may have to be reserved for temporary print files—such as MacWrite requires when it prints documents—or for temporary files required by some applications—such as those used by MacPaint.

If you use a specialized MacPaint working disk, you have enough free space that you can freely save copies of a work in progress, each with a slightly different name. When, as often happens, you wish you could go back to an earlier version of the picture, you have one stored right there on your Paint-Work disk. If you have such a MacWrite disk, you can work on a thirty-page book chapter, stored as three or four documents on your Write-Work disk.

To make an application work disk like Paint-Work or Write-Work, just make a copy of the release disk—the original one you purchased—and remove from it everything you don’t need. Then put applications that aren’t required for using the main application, such as Font Mover itself, in the trash. Don’t do this to the original application disk or to its stored backup. You may want to use such minor applications some day.

Some application disks, such as the one that Microsoft’s Multiplan is released on, use a protection scheme that requires that the original disk be present when the application is opened. It is easy to copy the contents to another disk, but when you open the copied Multiplan application, one is prompted to insert the original disk as a check. This approach to copy protection makes it easy to use Multiplan on a Macintosh hard disk drive, such as Tecmar’s MacDrive or the Corvus hard disk.

Working application disks are the ones you keep with your Mac. Aside from essential system and application files, they contain only a few docu-
ments, usually a group of related documents you are currently working on. When a document or small set of documents is in reasonably good shape and you want to go on to work with other documents on the disk, it's time to put copies onto Document Storage Disks.

To make a document storage disk, initialize a blank disk and give it a name that reflects the nature of the documents you plan to put on it. If it will contain all the files you are developing for use in your company's annual report for 1985, you might call it "Annual Rep '85." This disk may eventually contain documents made with a number of different applications, including MacPaint pictures, MacWrite text files, and Multiplan spreadsheets.

If you are a cautious computer user, you will create a second, duplicate storage disk for backup of your documents and store it in a separate location. In the case of the disk with the annual report, you should probably initialize another document storage disk and call it "Annual Rep '85 BACK."

Mark King used two document storage disks after his son's disk-rafting experiment taught him the importance of being backed up. When he began work on the user's guide for a new Spurious product, he initialized one disk as "NewUsersGuide" and the other as "NewUsersGuide BACK."

Mark began his work on the new user's guide by using a Macintosh program called ThinkTank, described in Chapter 6. ThinkTank is an application that helps users develop complex, flexible outlines. King's work with ThinkTank led him to decide that the manual would have a number of chapters. Most chapters would consist of several Macintosh documents. Table 3.1 is only the top level of planned organization for the user's guide. Each of the "major elements" has its own suboutline and set of notes that Mark made with ThinkTank.

Mark's plan called for a separate folder for every major element except the cover. The entire project called for the use of twenty document files—eight MacWrite documents and twelve MacPaint pictures. A disk directory window with twenty such documents jumbled together would look a bit confusing. Instead, the disk directory for his disk labeled "NewUsersGuide" had only seven items—a MacPaint document called "Cover" and six folders, each labeled as a chapter or appendix (Figure 3.7). This organization made it a lot easier for Mark King to find the documents he wanted.

Mark's home Macintosh has only one disk drive, so he made separate working disks for MacPaint and MacWrite. When he wanted to edit the text files from a section of the user's guide, he put those files on his "Write-Work" disk.
First he booted the system with Write-Work. To boot with a new system disk, you first eject the currently inserted disk. Then press the Reset key you installed on the left side of your Mac. Insert the new system disk, such as Write-Work, and your Mac will act as though it had just been turned on and had that disk inserted. Naturally, if the Mac were already off, you would simply turn it on and insert the boot disk.

After booting with Write-Work, Mark would eject the disk (by typing Command-E) and insert the disk with the files to be edited—in this case, NewUsersGuide. If he wanted to work on Chapter 2, he would then double-click on the folder icon labeled “Chapter 2.” He then would drag a select box around the two MacWrite files in this folder, “Control Functions” and “Setting Internal Options.” At this point he could drag the selected icons over to the open Write-Work window (see Figure 3.8).

Dialog boxes would then prompt him to make disk exchanges until copies of the two text files were placed on his working disk. After he made the
changes he wanted on these files, he would transfer copies to NewUsersGuide and to NewUsersGuide BACK. When he needed space on his working disk to edit another document, he could simply put these documents in the trash, since they were safely duplicated on two separate storage disks.

At work, Mark uses a Macintosh with a second drive. On this system it is more convenient to boot with a working disk that has both MacWrite and MacPaint, because it isn’t necessary to leave as much space on the boot disk for working documents. Mark puts NewUsersGuide in the second drive, opens its disk directory window, and opens the folder window that contains the documents he wants to work with. Then he just double-clicks the document he wants, and the appropriate application (MacWrite or MacPaint) is automatically loaded from the internal disk into the Mac’s RAM.

As with the single-drive method, after Mark has modified several documents he copies their new versions onto the appropriate backup document disk. Sometimes Mark isn’t certain which version of a document is most recent—the one on his ordinary document storage disk or the one on the backup disk. When this happens, he opens windows for the folders that contain the two versions of the document. Then he selects one of the documents in question and types Command-I to Get Info on it. He notes the time that the document was last modified, closes the info window, and then
selects the other document and gets info for that one. The most recently modified version is now identified.

WHEN THE PARTY'S OVER

Most projects, thankfully, come to an end. When the actual new user's guide—"Spurious Ultimate Widget Manual"—was published, Mark King breathed a sigh of relief. But he didn't throw away all the documents associated with the project. Instead, he renamed them, calling NewUsersGuide "UWIdgMan-Archive" and making NewUsersGuideBACK "UWIdgManArchiveBACK." He put the former on a shelf in his home office and the latter in a file folder at

![Diagram of file structure]

Figure 3.8 Drag selected files to a working disk
the office. An ARCHIVE DISK is one that stores documents that are no longer under development but in a finished state.

Sometimes when you finish a major project you will feel that you never want to see any of the documents connected with the project again. Don't throw them away. A month or a year after the end of the project you may find you want to produce a drawing very much like one of the figures you did for the current project. Think how nice it will be to take a copy of the old drawing from an archive disk and modify it for the new purpose, rather than have to recreate the whole thing from scratch!

Printed Directories

One problem with making effective use of the material stored on archive disks is that it's very hard to remember what is on them a few weeks or months after you've put them on the shelf. The solution is to keep printed directories of disk contents (see Figure 3.9). You can produce printed directories of different sizes and levels of detail using your Mac. I like to print two kinds of directories for my archive disks. One squeezes the top-level information about a disk onto a small area of paper. With a pair of scissors, I trim this printout and then fold it

<table>
<thead>
<tr>
<th>Size</th>
<th>Name</th>
<th>Kind</th>
<th>Last Modified</th>
</tr>
</thead>
<tbody>
<tr>
<td>0K</td>
<td>Appendix 1</td>
<td>folder</td>
<td>Wed, Mar 21, 1984</td>
</tr>
<tr>
<td>45K</td>
<td>Appendix 2</td>
<td>folder</td>
<td>Wed, Mar 21, 1984</td>
</tr>
<tr>
<td>50K</td>
<td>Chapter 1</td>
<td>folder</td>
<td>Mon, Mar 5, 1984</td>
</tr>
<tr>
<td>71K</td>
<td>Chapter 2</td>
<td>folder</td>
<td>Wed, Mar 21, 1984</td>
</tr>
<tr>
<td>45K</td>
<td>Chapter 3</td>
<td>folder</td>
<td>Tue, Mar 6, 1984</td>
</tr>
<tr>
<td>43K</td>
<td>Chapter 4</td>
<td>folder</td>
<td>Sat, Mar 17, 1984</td>
</tr>
<tr>
<td>3K</td>
<td>Cover</td>
<td>document</td>
<td>Sat, Feb 18, 1984</td>
</tr>
<tr>
<td>0K</td>
<td>Empty Folder</td>
<td>folder</td>
<td>Tue, Mar 6, 1984</td>
</tr>
</tbody>
</table>

Figure 3.9 Printed mini-directory, viewed by name
so that I can slip it into the little plastic bag that holds the 3½-inch disk itself. Now I can flip through my box of stored disks and immediately get some idea of what’s on them. If you use this method, you won’t have to put each disk into your Mac and open its directory window when you are searching for a particular document.

To make such a printed directory, simply bring open the directory window on the screen. Choose an appropriate view. If there are many documents on your disk, you should change the view to one of the non-icon displays—viewing by name, by size, by date, or by kind—so that all the document names will fit in the window. Size the window so that it displays everything. That is, drag the size box so that the window displays all the names. To print this display of the disk documents, type Command-Shift-4. This prints the active window—the one with horizontal lines in the title bar.

You can use the same technique to print directories for individual folders on a disk. Or you can open several folder windows on the display, positioning them so that they present a tiled rather than overlapped appearance. To have them all printed together, depress the Caps Lock key before typing Command-Shift-4. That will make the whole display print out. You can fold up the whole screen print and keep it with the disk.

As you’ve already learned in Chapter 1, you can print a special directory for the MacPaint documents on a disk. All you have to do is choose Print Catalog from the MacPaint application. If you have many MacPaint pictures on storage disk, this printout will be too large to fold up and keep with the disk. You might want to consider keeping a file folder of directories for your disks. Such a folder can contain both the special MacPaint pictorial directories and screen prints of ordinary windows. When you need to find an old document, you can quickly flip through your printed disk directories to find it.
CHAPTER 4

SPREADSHEETS

USING MULTIPLAN—AN EXAMPLE 81
SHOWING FORMULAS 103
CHARTING IN MULTIPLAN 104
OTHER KINDS OF SPREADSHEETS WITH MULTIPLAN 106
MULTIPLAN’S COMMAND KEYS 108
BOOKS ABOUT MULTIPLAN 108
A spreadsheet application lets a user create and edit a spreadsheet—a large display of rows and columns that can contain words and numbers (Figure 4.1). Such spreadsheets are widely used in business for budgeting and for other forms of financial planning. They can be adapted for a much wider range of uses, however, including making tables of text items or even recording students' grades in a flexible computer-based gradebook.

You can use the keyboard to enter values for particular locations in the spreadsheet. The real power of spreadsheet programs, however, is that they let you specify the values of some locations by rules. For example, you can tell the program that the value to be displayed in a certain location should always be the sum of the values in the eight locations above it in the column.

Financial worksheets have long been used in business for planning and forecasting. The work was done using large sheets of paper divided into rows and columns, and arithmetic operations were performed using a calculator. When this manual technique was the only means for doing spreadsheets, making any change to the worksheet was a painful process. Because subtotals, totals, and other figures typically depended on the altered value, many cells on the worksheet would have to be recalculated by hand every time there was a change.

A computer-based spreadsheet makes changing values very easy. Recalculations can proceed automatically, showing all the new values within a few seconds.

**USING MULTIPLAN—AN EXAMPLE**

The first spreadsheet application created for the Macintosh was Microsoft's Multiplan. Multiplan is a powerful program—an earlier version won *Infoworld* magazine's award for best software package of 1982. The Macintosh version has a number of impressive new features that exploit the graphics capabilities of the Mac and let the user construct formulas by pointing with the mouse rather than by typing references to cells. Consider an example of how a simple budget can be prepared with Multiplan.
<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SPURIOUS WIDGET</strong></td>
<td><strong>PROJECT BUDGET — CANARD WIDGET</strong></td>
<td><strong>TOTALS</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Year One, January — June</strong></td>
<td><strong>One, July — December</strong></td>
<td><strong>Month</strong></td>
<td><strong>% on Proj.</strong></td>
<td><strong>Monthly Rate</strong></td>
<td><strong>Amount</strong></td>
<td><strong>% on Proj.</strong></td>
<td><strong>Monthly Rate</strong></td>
<td><strong>Amount</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Project Director</strong></td>
<td>50.00%</td>
<td>$3,744.00</td>
<td>$13,470.40</td>
<td>60.00%</td>
<td>$3,968.64</td>
<td>$14,287.10</td>
<td>27,795.50</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Department Manager</strong></td>
<td>53.00%</td>
<td>$3,729.31</td>
<td>$13,782.84</td>
<td>52.00%</td>
<td>$3,470.56</td>
<td>$12,570.13</td>
<td>18,440.25</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Senior Designer</strong></td>
<td>50.00%</td>
<td>$3,352.62</td>
<td>$10,986.00</td>
<td>50.00%</td>
<td>$3,563.72</td>
<td>$10,991.16</td>
<td>20,557.16</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Staff Designer 1</strong></td>
<td>100.00%</td>
<td>$2,850.00</td>
<td>$17,160.00</td>
<td>100.00%</td>
<td>$3,031.60</td>
<td>$18,186.00</td>
<td>35,365.60</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Staff Designer 2</strong></td>
<td>100.00%</td>
<td>$2,300.00</td>
<td>$13,800.00</td>
<td>100.00%</td>
<td>$2,436.00</td>
<td>$14,626.00</td>
<td>28,462.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Draftsperson</strong></td>
<td>80.00%</td>
<td>$2,040.00</td>
<td>$9,792.00</td>
<td>80.00%</td>
<td>$2,162.40</td>
<td>$10,379.52</td>
<td>20,171.52</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Clerk</strong></td>
<td>100.00%</td>
<td>$1,779.00</td>
<td>$10,674.00</td>
<td>100.00%</td>
<td>$1,885.74</td>
<td>$11,314.44</td>
<td>21,988.44</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Total Salaries</strong></td>
<td><strong>$83,941.98</strong></td>
<td></td>
<td></td>
<td><strong>$88,978.50</strong></td>
<td></td>
<td></td>
<td><strong>$172,920.48</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Year Two, January — June</strong></td>
<td><strong>Two, July — December</strong></td>
<td><strong>Month</strong></td>
<td><strong>% on Proj.</strong></td>
<td><strong>Monthly Rate</strong></td>
<td><strong>Amount</strong></td>
<td><strong>% on Proj.</strong></td>
<td><strong>Monthly Rate</strong></td>
<td><strong>Amount</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Project Director</strong></td>
<td>50.00%</td>
<td>$3,968.64</td>
<td>$11,905.92</td>
<td>50.00%</td>
<td>$4,206.76</td>
<td>$12,620.28</td>
<td>24,526.20</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Department Manager</strong></td>
<td>33.00%</td>
<td>$4,792.26</td>
<td>$9,488.67</td>
<td>33.00%</td>
<td>$5,079.60</td>
<td>$10,059.00</td>
<td>19,558.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Senior Designer</strong></td>
<td>50.00%</td>
<td>$3,563.72</td>
<td>$10,691.16</td>
<td>50.00%</td>
<td>$3,777.54</td>
<td>$11,332.63</td>
<td>22,000.17</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Staff Designer 1</strong></td>
<td>100.00%</td>
<td>$3,031.60</td>
<td>$18,186.00</td>
<td>100.00%</td>
<td>$3,213.50</td>
<td>$19,280.90</td>
<td>37,494.40</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Staff Designer 2</strong></td>
<td>75.00%</td>
<td>$2,438.00</td>
<td>$10,991.16</td>
<td>75.00%</td>
<td>$2,584.28</td>
<td>$11,626.25</td>
<td>22,400.53</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Draftsperson</strong></td>
<td>75.00%</td>
<td>$2,162.40</td>
<td>$9,792.00</td>
<td>75.00%</td>
<td>$2,292.14</td>
<td>$10,379.52</td>
<td>20,045.45</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Clerk</strong></td>
<td>100.00%</td>
<td>$1,885.74</td>
<td>$11,314.44</td>
<td>100.00%</td>
<td>$1,998.88</td>
<td>$11,993.31</td>
<td>23,302.75</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Total Salaries</strong></td>
<td><strong>$82,291.59</strong></td>
<td></td>
<td></td>
<td><strong>$87,229.09</strong></td>
<td></td>
<td></td>
<td><strong>$169,520.69</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Fringe</strong></td>
<td>24.50%</td>
<td>$20,397.90</td>
<td></td>
<td></td>
<td>$21,621.78</td>
<td></td>
<td>$42,019.68</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Subtotal</strong></td>
<td>33,553.97</td>
<td>33,596.17</td>
<td>33,553.97</td>
<td>33,596.17</td>
<td>33,553.97</td>
<td>33,596.17</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Overhead</strong></td>
<td>72.00%</td>
<td>$102,829.70</td>
<td></td>
<td></td>
<td>$114,413.40</td>
<td></td>
<td>$217,243.10</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Consultant</strong></td>
<td>14.40%</td>
<td>$8,400.00</td>
<td></td>
<td></td>
<td>$9,595.00</td>
<td></td>
<td>$18,995.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Equipment</strong></td>
<td>14.40%</td>
<td>$8,400.00</td>
<td></td>
<td></td>
<td>$9,595.00</td>
<td></td>
<td>$18,995.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td><strong>$478,163.51</strong></td>
<td></td>
<td></td>
<td><strong>$440,884.54</strong></td>
<td></td>
<td></td>
<td><strong>$919,048.05</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Figure 4.1 A sample spreadsheet**
The Base Canard Widget Project Budget

Mark King had spent several weeks developing the plans for a major new research and development project at Spurious Widget, Inc. The new project, the "Base Canard Widget," would take about two years to develop, using his current staff. Mark developed a preliminary project proposal, outlining the effort and the expected benefits. At an early afternoon meeting he presented the concept to his department manager and the divisional vice-president of Spurious. They agreed that the idea was a timely one, offering significant new growth opportunities for Spurious. They told him to have a project budget drawn up for a ten o'clock meeting the next day.

Naturally, they expected him to stay up half the night, working out the budget and totaling and checking his figures. Mark produced the preliminary budget shown at left in just over an hour. Here is how he did it.

He began by starting up Multiplan, double-clicking the Multiplan icon (Figure 4.2). Since he hadn't selected an existing Multiplan document, the application opened a spreadsheet labeled "Untitled." Mark decided to begin his budget by listing the job titles of the people who would be involved in the project. To put a name into a cell of a Macintosh Multiplan spreadsheet, all you have to do is position the mouse pointer at the cell you want to use (Figure 4.3). Then, to put a label—an alphabetic string—into the cell, click the cursor to select the cell, and type in the name.

As Mark typed the name of the first label, it appeared in the formula bar at the top of the desktop as well as in the selected cell. When the name he entered, "Project Director," grew too large for the cell, the new letters continued to appear only in the formula bar as they were typed (Figure 4.4). Mark accepted the contents of the formula bar as the contents of the cell by typing the Return key. At that point, the cell was displayed in normal rather than inverse video, and the cell below became the selected cell.

Mark successively entered the titles of all the staff members associated with the planned project by typing the text of their titles and typing return (Figure 4.5). After all seven members of the team had been listed, the selected cell was at Row 11, Column 1 (shown as R11C1 at the beginning of the formula bar). Then Mark wanted to put some titles across Row 3. He positioned the pointer and clicked the mouse button at Row 3, Column 3 (R3C3). He typed "% on Project" for the text of this cell, but then typed the Enter key rather than the Return key to accept the title. This kept R3C3 as the selection. Now typing Command-H made the title right-justified in the cell.
Command-H may sound like a strange mnemonic for right-justification. The reason the key command is not Command-R is that that key combination is used in Multiplan to execute another common function, Fill Right. The three cell-justification key commands are Command-F, Command-G, and Command-H for justifying the text left, center, and right, respectively. The way to remember these keys is that the G key is in the center of the keyboard, with F to its left and R to its right. Naturally, Mark could have pulled down the Format menu to choose the Align Center option, but like most experienced Mac users, he takes advantage of the greater speed of key commands when they are available.

After centering the first column title, Mark moved to the cell to the right by typing the Tab key (Figure 4.6). He could just as well have used the mouse.
### Figure 4.4 Entering text in a cell

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

3 Project Director

### Figure 4.5 Staff titles in column 1

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

3 Project Director

5 Department Manager

6 Senior Designer

7 Staff Designer 1

8 Staff Designer 2

9 Draftsperson

10 Clerk

11

12

13

14
to select the next cell for entry, but it is faster to use the keyboard when entering a sequence of items in a row or column. The Return key selects the cell below the current cell, and Tab selects the cell to the right. Shift-Return selects the cell above the current cell, and Shift-Tab selects the cell to the left (see Figure 4.7).

As he entered the “Monthly Rate” column title, Mark realized that the standard column width (10 numerals) was narrower than what he wanted, so he selected all cells, and used the Column Width option from the Format menu to increase the column width to 12 (Figure 4.8). The new column size looked about right, but Column 2 was larger than he needed, since it was being used only for overflow text from Column 1. Mark put the pointer on the vertical line dividing the headers of Columns 2 and 3, depressed the mouse button, and dragged a bit to the left. When he released the button, Column 2 had a more modest width.

After getting the new column width, Mark hit the Tab key to select R3C5 and entered the word “Amount,” which he centered. Next, he put the text
“Year One, January–June” in cell R2C4, typed the Enter key to accept the text, and typed Command-G to center the text.

Mark planned on treating each year of the project in two parts, before and after the annual Widget salary adjustments, which take place on July 1. To the right of the three columns reserved for the January–June data, he needed three columns for the corresponding data for July–December. Rather than type in
the three headings again, he selected the first three by dragging from R3C3 to R3C5 (Figure 4.9). All three appeared in inverse video. Mark copied them to the Clipboard by typing Command-C. Then he used the Scroll Bar to move the display to the right on the spreadsheet. He selected three cells in Row 3, from Column 7 to Column 9, and typed Command-V to paste the Clipboard contents into that location (Figure 4.10).

Mark decided the blank column (C6) wasn't necessary, so he selected several cells from Column 6, including R3C6, and used Command-X to cut the cell to the left of the "% on Project" title. The three new column titles shifted to the left to fill in the deleted cell (see Figure 4.11). If there had been material in the cells below the deleted area as well as to the right, Multiplan would have presented a dialog box to ask Mark whether he wanted the deleted cells to be filled from below or from the right.

Mark decided that he wanted to put a title on the worksheet, but he needed more room at the top. He put the mouse pointer on the header for Row 1—the little box at the left of the first row—and clicked the mouse button
Figure 4.10 Pasting a copy of the three cells

Figure 4.11 The screen after a cut
while holding down a shift key. Doing this twice added two new rows to the top of the page, shifting everything he'd typed so far two rows down. He then added title captions at R1C5 and R2C5, both centered by typing Command-G (see Figure 4.12).

Since he would want to get total salary costs for each period, Mark put the text title “Total Salaries” at R14C1. He also labeled R15C1 with the word “Fringe,” aligning this one on the right side of the cell. Then he put the fringe benefit rate, 24.3%, at R16C2.

Now that the basic layout of the first portion of the spreadsheet was set up, Mark could finally start entering the numbers. First he entered the proportion of time that each project member would put in on the project in Column 3, Rows 6–12. Then he entered the monthly salaries of those employees (Figure 4.13). He wanted to show the proportions of time committed to the project as percentages, so he dragged Column3 from Row6 to Row 12 and then pulled down the Format menu, selecting the Percent option.

![Spreadsheet](image-url)  
**Figure 4.12** New caption lines inserted
He also chose the **Commas** option from this menu, to make commas appear in numbers of more than three digits, for greater readability. This option always applies to the entire spreadsheet. Then he dragged through the monthly rate column (C4) to select it and chose the **Dollar** option from the Format menu (see Figure 4.14).

The next step was to compute how much each employee would cost for the six-month period in question. Mark selected Row 6, Column 5, which was to display the Amount to be budgeted for the project director (Mark himself) for this period. All the values Mark entered in the “% on Project” and “Monthly Rate” columns so far had been constant values, not dependent on other cells in the spreadsheet. The values of Column 6, “Amount,” however, would depend on the values in these columns.

Mark needed to define a formula that would control the value of the selected cell. He typed “=”, which told Multiplan that he was about to enter a formula, rather than an absolute value, such as a text or a number. Such formulas give rules that make the value in the cell with such a formula be a
function of the values of one or more other cells. The rule Mark wanted to use was that the proportion (in the cell two columns to the left) should be multiplied by the salary (to the immediate left of the cell being defined) and that that monthly cost should be multiplied by six, the number of months in the period. Mark was able to enter the required formula with just two mouse clicks and three key presses. First he clicked R6C3. The formula bar showed "=RC[-2]", showing that he had referred to a cell on the same row, two columns left. He typed "*", the multiplication sign, which then appeared in the formula bar. He then clicked the cell R6C4, and finally typed "*6". When he hit the Enter key, Multiplan calculated the result, $13,478.40 (see Figure 4.15).

Of course, Mark could have used his desk calculator or the Desktop Accessory Calculator to find out that $3,744.00 * 6 is $13,478.40 and then entered that number as an absolute value. The advantage to using a formula is not only that Multiplan performs the calculation but also that it does so automatically, no matter what the value of the cells the formula refers to. Therefore, if Mark’s superiors like his proposal so well that they award him an
unscheduled merit pay increase, he can modify the number $3,744 to the new value, and the cell R6C5 will be recomputed appropriately.

The next step was to create similar formulas for the amounts to be allocated for other members of the project. First, he dragged the pointer through Column 5, Row 6 to Row 12. Then he typed Command-D to instruct Multiplan to fill the selected area below R6C5 with the same formula (Figure 4.16). That formula does not refer to absolute locations, but only to cell locations relative to the cell whose value is being computed. This means that the filled area will not contain the same value as the first cell, but rather the appropriate values, the ones dependent on the two cells to the left.

The next step was to compute the total salary budget for the period. Mark selected the cell at R14C5 to hold this value. He typed "=" to let Multiplan know that he was going to enter a formula. To get the sum of the salaries, he simply typed the word "SUM" followed by the left parenthesis, and then dragged the mouse from R6C5 to R12C5. The area he had dragged out was enclosed in dotted lines. The dragging caused the formula bar to display a

**Figure 4.15** A cell value determined by formula
relative reference to a range of values, "R[-8]C:R[-2]C" (see Figure 4.17). The input to the SUM function was thereby defined as the range from the eighth cell above the current cell to the second cell above the current cell. To finish off the formula, Mark typed a right parenthesis—marking the end of the input to SUM—and then typed the Enter key. The screen then displayed the total salaries for the first six months (Figure 4.18).

To determine the amount of fringe benefits for this period, Mark selected a new cell, R16C5, and defined it to be the product of the Fringe Rate (in cell R16C2) times the Salary Total (in cell R14C5). Again, Mark created the formula by using pointing inputs, so that all the references were relative to the current cell rather than absolute (Figure 4.19).

At this point, even though less than a fourth of the entire spreadsheet had been created, Mark had done most of the work. Everything else could be built on the foundation he had already laid. To fill in the "% on Proj." column for the second six months, Mark didn't type in the figures again. Nor did he choose to use Copy and Paste. He reasoned that if he decided to change someone's percent effort on the project in the first half of the year, he would probably want it to carry over to the second half. So he simply set up the simplest
### Figure 4.17 Entering the formula for total salaries

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>% on Proj.</td>
<td>Monthly Rate</td>
<td>Amount</td>
<td>% on Proj.</td>
<td>Mon</td>
</tr>
<tr>
<td>6</td>
<td>60.00%</td>
<td>$3,744.00</td>
<td>$13,476.40</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>33.00%</td>
<td>$4,521.00</td>
<td>$8,951.58</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>50.00%</td>
<td>$3,362.00</td>
<td>$10,086.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>100.00%</td>
<td>$2,860.00</td>
<td>$17,160.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>100.00%</td>
<td>$2,300.00</td>
<td>$13,800.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>80.00%</td>
<td>$2,040.00</td>
<td>$9,792.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>100.00%</td>
<td>$1,779.00</td>
<td>$10,674.00</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Figure 4.18 A sum entered by formula

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>% on Proj.</td>
<td>Monthly Rate</td>
<td>Amount</td>
<td>% on Proj.</td>
<td>Mon</td>
</tr>
<tr>
<td>6</td>
<td>60.00%</td>
<td>$3,744.00</td>
<td>$13,476.40</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>33.00%</td>
<td>$4,521.00</td>
<td>$8,951.58</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>50.00%</td>
<td>$3,362.00</td>
<td>$10,086.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>100.00%</td>
<td>$2,860.00</td>
<td>$17,160.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>100.00%</td>
<td>$2,300.00</td>
<td>$13,800.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>80.00%</td>
<td>$2,040.00</td>
<td>$9,792.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>100.00%</td>
<td>$1,779.00</td>
<td>$10,674.00</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

=SUM(R[-8]:C[-2]:C)

$13,941.56
possible relative formula for R6C6: he set it equal to the cell three columns to the left. He then filled this formula down through the next six cells, so that the same percentages are used in both halves of the year (Figure 4.20). Still, he can always go back and change any of these values, replacing it with an absolute value or a different formula.

To determine the values for the monthly rates in the second half of the first year, Mark assumed that those on the project would receive salary increases averaging 6 percent on July 1. He therefore defined cell R6C7 to be equal to the earlier monthly rate (given three columns to the left) times 1.06 (Figure 4.21). He filled this formula down through the next six cells as well, so that all monthly salaries were 6 percent larger than in the first half of the year. Following the model he used for the first three columns, he made the formulas for the salary amounts for each member of the project and for the total salaries and fringe benefits (Figure 4.22).

At this point, Mark moved down the page and entered some titles for the second year of the proposed project. He assumed that the same people would work on the project, so he defined a formula for the title of the first participant
Figure 4.20 A column defined by reference to another column

<table>
<thead>
<tr>
<th>5</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>% on Proj.</td>
<td>Monthly Rate</td>
<td>Amount</td>
<td>% on Proj.</td>
<td>Mon</td>
</tr>
<tr>
<td>6</td>
<td>60.00%</td>
<td>$3,744.00</td>
<td>$13,478.40</td>
<td>60.00%</td>
<td>60.00%</td>
</tr>
<tr>
<td>7</td>
<td>33.00%</td>
<td>$4,521.00</td>
<td>$8,951.58</td>
<td>33.00%</td>
<td>33.00%</td>
</tr>
<tr>
<td>8</td>
<td>50.00%</td>
<td>$3,362.00</td>
<td>$10,086.00</td>
<td>50.00%</td>
<td>50.00%</td>
</tr>
<tr>
<td>9</td>
<td>100.00%</td>
<td>$2,860.00</td>
<td>$17,160.00</td>
<td>100.00%</td>
<td>100.00%</td>
</tr>
<tr>
<td>10</td>
<td>100.00%</td>
<td>$2,300.00</td>
<td>$13,800.00</td>
<td>100.00%</td>
<td>100.00%</td>
</tr>
<tr>
<td>11</td>
<td>60.00%</td>
<td>$2,040.00</td>
<td>$9,792.00</td>
<td>60.00%</td>
<td>60.00%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$83,941.96</td>
</tr>
<tr>
<td>12</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$20,397.90</td>
</tr>
</tbody>
</table>

Figure 4.21 A formula that computes wage increases

\[
=RCl-31*1.06
\]
Figure 4.22 Second half of first year, complete

Figure 4.23 Participant titles for the second year
in the second year. Again, he only had to select the new location (R21C1), type "=" to start defining a formula, and point to the first title for Year One. Multiplan automatically built the formula that refers to the value fifteen rows above ("=R[−15]C") (Figure 4.23). Note that you can create formulas that produce text values, not only numeric values. As before, the Fill Down option let Mark copy a formula for a selected portion of the worksheet, so that all the participants' titles referred to those used for the first year.

Mark went on building the Year Two portion of the spreadsheet, referring to relevant cells in the Year One section as he created the formulas that determined the new values. When both sections were complete, Mark set up a new column of annual totals in Column 10. Here he was again able to create a single formula that referred to the two Amount columns to the left, and use Fill Down to use that formula for the whole column (Figure 4.24).

A Budget Summary

Mark created a new portion of the spreadsheet, below the Year One and Year Two portions. The new section summarized the figures from the first portions, together with estimates of the non-salary contributions to the project budget. Mark inserted text labeling the summarized salaries, fringe benefits, and other expenses such as travel and equipment.

He realized that there would not be enough room on a printed page to print all the cells he had created already, along with the new ones below, so Mark selected cell R32C1 as a good spot for a printed page break. Then he pulled down the Options menu and chose Set Page Break (Figure 4.25). That separated the new lower portion of the spreadsheet from the upper. When he had finished, Mark had the bottom-line figure for his proposed project. It would cost $919,148.05, including overhead (Figure 4.26). He printed out the complete spreadsheet and pasted the four pieces of printout paper together. A company photocopier with reduction capability let him copy the whole spreadsheet to a single 8½-by-11-inch piece of paper (Figure 4.27).

The printed spreadsheet served as the focus of Mark's presentation to his department manager and the division vice-president the next morning. After some discussion, the department manager convinced Mark that he could do the job with some reductions in staffing. They agreed that the junior staff designer could participate on a half-time basis on the project. Furthermore, the clerical duties for this project could be handled by a Level 2 clerk as well as by a Level 4.
Figure 4.24 The column of totals

Figure 4.25 Setting a page break
"How long will it take to get the new budget figures based on those assumptions, Mark?" asked the department manager.

"Less time than it will take me to walk down the hall to my office and back," he told her. "Why don't you two take a coffee break, and I'll be back with the new figures in five minutes."

Mark opened his Canard project Multiplan document and changed the "% on Proj." value for staff designer 2 to 50 percent. He also edited the Clerk designation to read "Clerk, Level 2" and changed the salary level appropriately (Figure 4.28). He made the Year Two change in time on project and then took a look at the new bottom line displayed in R49C7. Before his bosses' coffee had started to cool, he had the new budget total back to them (Figure 4.29).

Mark and his department manager went on to make extensive use of Multiplan in preparing more detailed budget plans. One feature they found very useful was Multiplan's ability to incorporate links from one spreadsheet to another. This made it possible to develop separate sheets with the breakdown of equipment expenses, for example, and to link the totals from those sheets to references in the budget summary section. Whenever a change
<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**SPURIOUS WIDGET**

**PROJECT BUDGET -- CANARD WIDGET**

<table>
<thead>
<tr>
<th></th>
<th>Year One, January - June</th>
<th>Year One, July - December</th>
<th>TOTALS</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>Project Director</td>
<td>60.00%</td>
<td>$3,744.00</td>
</tr>
<tr>
<td>7</td>
<td>Department Manager</td>
<td>50.00%</td>
<td>$4,581.00</td>
</tr>
<tr>
<td>8</td>
<td>Senior Designer</td>
<td>50.00%</td>
<td>$3,563.72</td>
</tr>
<tr>
<td>9</td>
<td>Staff Designer 1</td>
<td>100.00%</td>
<td>$2,860.00</td>
</tr>
<tr>
<td>10</td>
<td>Staff Designer 2</td>
<td>100.00%</td>
<td>$2,300.00</td>
</tr>
<tr>
<td>11</td>
<td>Draftsman</td>
<td>80.00%</td>
<td>$2,040.00</td>
</tr>
<tr>
<td>12</td>
<td>Clerk</td>
<td>100.00%</td>
<td>$1,779.00</td>
</tr>
<tr>
<td>13</td>
<td>Total Salaries</td>
<td>75.90%</td>
<td>$83,941.98</td>
</tr>
<tr>
<td>14</td>
<td>Fringe</td>
<td>24.30%</td>
<td>$20,397.90</td>
</tr>
<tr>
<td>15</td>
<td>Year Two, January - June</td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>Project Director</td>
<td>50.00%</td>
<td>$3,968.61</td>
</tr>
<tr>
<td>17</td>
<td>Department Manager</td>
<td>33.00%</td>
<td>$3,998.26</td>
</tr>
<tr>
<td>18</td>
<td>Senior Designer</td>
<td>50.00%</td>
<td>$2,535.72</td>
</tr>
<tr>
<td>19</td>
<td>Staff Designer 1</td>
<td>100.00%</td>
<td>$2,300.00</td>
</tr>
<tr>
<td>20</td>
<td>Staff Designer 2</td>
<td>75.00%</td>
<td>$2,162.40</td>
</tr>
<tr>
<td>21</td>
<td>Draftsman</td>
<td>75.00%</td>
<td>$1,779.00</td>
</tr>
<tr>
<td>22</td>
<td>Clerk</td>
<td>100.00%</td>
<td>$1,885.74</td>
</tr>
<tr>
<td>23</td>
<td>Total Salaries</td>
<td>75.90%</td>
<td>$82,291.59</td>
</tr>
<tr>
<td>24</td>
<td>Fringe</td>
<td>24.30%</td>
<td>$21,621.78</td>
</tr>
</tbody>
</table>

**BUDGET SUMMARY -- CANARD WIDGET**

<table>
<thead>
<tr>
<th></th>
<th>Year One</th>
<th>Year Two</th>
<th>2-Year TOTAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>36</td>
<td>Salaries</td>
<td>$172,920.48</td>
<td>$169,520.69</td>
</tr>
<tr>
<td>37</td>
<td>Fringe</td>
<td>$167,829.70</td>
<td>$167,215.20</td>
</tr>
<tr>
<td>38</td>
<td>Materials &amp; Supplies</td>
<td>$13,833.64</td>
<td>$13,561.65</td>
</tr>
<tr>
<td>39</td>
<td>Travel</td>
<td>$4,325.01</td>
<td>$4,236.02</td>
</tr>
<tr>
<td>40</td>
<td>Subtotal</td>
<td>$233,096.81</td>
<td>$232,243.34</td>
</tr>
<tr>
<td>41</td>
<td>Overhead</td>
<td>$167,215.20</td>
<td>$167,215.20</td>
</tr>
<tr>
<td>42</td>
<td>Consultant</td>
<td>$14,400.00</td>
<td>$6,400.00</td>
</tr>
<tr>
<td>43</td>
<td>Equipment</td>
<td>$13,833.64</td>
<td>$13,561.65</td>
</tr>
<tr>
<td>44</td>
<td>TOTAL</td>
<td>$487,163.51</td>
<td>$440,984.54</td>
</tr>
</tbody>
</table>

**Figure 4.27** The pasted-together printout
Figure 4.28 Modifying a spreadsheet

is made in a source spreadsheet, the effects of that change are automatically carried over into the spreadsheet to which it is linked.

SHOWING FORMULAS

After you have built a Multiplan spreadsheet, you should archive it in the same manner that you archive other documents, as discussed in Chapter 3, "Managing Your Macintosh Disks." It also makes sense to create two paper archive versions of your spreadsheets. One archive copy is a printout of the entire spreadsheet with its final values, just like the ones given above in this chapter. The other type of paper archive is a printout of the spreadsheet when the Show Formulas option is selected. Show Formulas has the effect of replacing values with the formulas that underlie them. Having a copy of the formulas of each of your spreadsheets handy helps make you ready to answer
questions about how the values of particular cells were computed (Figure 4.30).

When you choose this option, the spreadsheet columns are doubled in size, to give enough room to present most formulas. Of course, you can still change column widths, so you can make the columns a little narrower if they don't have to be twice as wide as normal. The spreadsheet can be printed with its formulas rather than its values displayed. Paste together the formula printout, make a reduced photocopy, and keep it with the printed spreadsheet that shows values.

**CHARTING IN MULTIPLAN**

For advanced charting, Microsoft’s Chart program is clearly the best method for creating and editing charts related to Multiplan spreadsheets. However, Multiplan itself has some simple charting capabilities. You can choose to
<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>35</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>36</td>
<td>Year One</td>
<td>BUDGET SUMMARY -- CANARD WIC</td>
<td>Year Two</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>37</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>Materials &amp; Supplies</td>
<td>+R[-2]C[+0.08]</td>
<td>+R[-2]C[+0.08]</td>
<td></td>
<td>+RC[-2]RC[-4]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>41</td>
<td>Travel</td>
<td>+R[-3]C[+0.025]</td>
<td>+R[-3]C[+0.025]</td>
<td></td>
<td>+RC[-2]RC[-4]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>43</td>
<td>Overhead</td>
<td>0.72</td>
<td>0.72</td>
<td>0.72</td>
<td>0.72</td>
<td>0.72</td>
<td>0.72</td>
</tr>
<tr>
<td>44</td>
<td>Consultant</td>
<td>14400</td>
<td>8400</td>
<td></td>
<td>+RC[-2]RC[-4]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>45</td>
<td>Equipment</td>
<td>62637</td>
<td>33125</td>
<td></td>
<td>+RC[-2]RC[-4]</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Figure 4.30** The formulas of a spreadsheet
display some cells in bar graph format by selecting them and choosing Bar Graph from the Format menu. Mark King wanted to give a quick visual impression of the relative contributions of the major categories to the budget total. He first used the split bar to create two separate "panes" in the Multiplan window. On the left he displayed Columns 1 and 2. In the right panel he displayed the totals by category for the project. He increased the size of Column 8. Then he set up a formula for R38C8—the first relevant cell of the new column. The formula set this value to the value of the cell to the left, divided by 35,000. He filled this formula down through the column, and then chose the Bar Graph option. By typing Command-F, he aligned the bar charts to the left of the cells (Figure 4.31).

OTHER KINDS OF SPREADSHEETS WITH MULTIPLAN

Multiplan can be used for many things other than budget planning, of course. As you might imagine, one can easily prepare a spreadsheet form for preparing a financial statement. Multiplan can also be used for a variety of
Figure 4.32 The Multiplan Command keys
other financial applications. A number of books on Multiplan describe methods for cash management, debt management, fixed asset management, working capital management, quote preparation, inventory management, billing, commissions calculations, and so on. Two examples of such books are referenced at the end of this chapter.

Multiplan has other uses, including many that have nothing to do with business or finance. Writers may choose to construct text tables using Multiplan rather than MacWrite or Word. This process is described below, in Chapter 10.

MULTIPLAN'S COMMAND KEYS

Figure 4.32 presents the Multiplan menus which mark the special key presses that can be used to issue commands from the keyboard rather than pulling down a menu for every step.

BOOKS ABOUT MULTIPLAN

This book presents ten different exercises for students of Microsoft Multiplan, ranging from "Invoicing from Inventory" to "Calculating Commission Using the Iteration Option." The book does not assume a Macintosh, so the reader must translate the book's instructions to type, for example, "R11C5" to mean to point to the cell at Row 11, Column 5.

This book presents models for eight major Multiplan application areas: cash management, debt management, fixed asset management, working capital management, financial statements, planning and budgeting, and quote preparation. This book contains a great deal more explanatory text than the Williams book does. Like that book, it does not include Macintosh-specific instructions.
CHAPTER 5

CHARTS

MAKING A PIE CHART  111
CHARTS WITH MORE THAN ONE SET OF DATA  120
USING MULTIPLAN AND CHART TOGETHER  127
FURTHER READING ABOUT MICROSOFT CHART  129
Figure 5.1 Several kinds of charts
Microsoft’s Chart application is a specialized application for the production of presentation graphs. With Chart, you can quickly and easily create area charts, bar charts, column charts, pie charts, line charts, and scatter plots (see Figure 5.1). Chart takes advantage of the Macintosh’s crisp, high-resolution display and rapid graphics capabilities to automatically construct graphs on your screen and in printouts, based on sets of numbers you enter. In Chart’s terminology, each such set of numbers is called a series or a data series. Chart lets the user enter a series and graph it in any one of a number of standard Chart styles.

MAKING A PIE CHART

At the second presentation of Mark King’s proposed project, described in Chapter 4, his division vice-president casually remarked that the proposed budget was still rather large. Mark decided to produce a couple of pie charts to convey quickly what proportions of the budget were going to different categories. To begin, he started up his Mac with the Chart disk (Figure 5.2).

Mark opened Microsoft Chart by double-clicking its icon (Figure 5.3). Two windows are open whenever you begin working with a new chart. One is the chart window itself, here labeled “Untitled.” This is where the chart you create will appear. The other is a window labeled “New Series.” This is where you will enter the series of numbers that are to be charted.

Mark chose to begin his Chart session by specifying the form of the data to be entered. He pulled down the Data menu and selected the Text option. This means that the series of numbers to be plotted will each have a name. Other data options include entering a numbered series or a series based on the days of the week. A dialog box labeled “Text Series” opened on the screen (Figure 5.4). The default name of the series is simply the word “Series” with the current time.

Mark used the standard text-editing techniques to give the new series the name “Canard Budget Categories.” He also set the X-axis label to “Major Category” in place of X, and “Dollars” in place of Y (Figure 5.5). After he clicked the “OK” button in the dialog box, a new window appeared on the screen with the title “Canard Budget Categories” (Figure 5.6).

Before actually entering the values to be plotted, Mark used the Format window to set up the formats to be used for the categories and values he
Figure 5.2 Mark's chart

Figure 5.3 A new chart opened
Figure 5.4 The text Data dialog box

Figure 5.5 Dialog box with series labels
planned to enter (Figure 5.7). For the Categories, he chose to align the names on the left. For the Values, he accepted the default alignment, on the right, but specified that the text "$" should precede each value (Figure 5.8). The reason Mark didn’t choose the “Dollar” number format is that he didn’t want to include the two decimal places for cents.

After clicking the “OK” button in the Values dialog box, Mark entered the names and rounded off dollar values of the major budget categories in the Canard project budget. The process consists of simply typing in the first name, hitting the Enter key, typing in the first value, hitting Enter again, then typing the second name, pressing Enter, and so on (Figure 5.9).

At this point, Mark had to choose which type of pie chart he wanted to display. One of his goals was to show that almost half the project costs were indirect. For this reason, he had listed the two major indirect expense categories, overhead and fringe benefits, next to each other in the series. Because he wanted to show the relative contribution of each category to the total expense of the project, he chose to present the data in a pie chart. He pulled down the Gallery menu (Figure 5.10) to Pie and was presented with six

![Figure 5.6 “Canard” series window](image-url)
Figure 5.7 Format Categories dialog box

Figure 5.8 Format Values dialog box
Figure 5.9 The completed series

Figure 5.10 The Gallery menu
options for different types of pie charts (Figure 5.11). He chose the fifth one, which presents the segments of the pie with different shadings and their category labels.

Clicking the "OK" button closes the Pie Chart gallery, and the new selection is active. To actually see the chart, Mark finally clicked the Plot Series box in the "Canard Budget Categories" window. The resulting display shows the relative contributions to the Canard project budget of each of the major budget categories, such as salaries, equipment, overhead, and so on (Figure 5.12).

Mark felt that, while this presentation was fairly effective, he needed a supplementary presentation to really impress his vice-president with the contribution of indirect costs to the budget. He would create a second pie chart with only two segments, one for direct costs—salaries, equipment, materials and travel, and the consultant—and the other for indirect costs—overhead and fringe benefits.

The first step was to do pull down the menu and choose Save as. Mark named the chart "Canard Pies." Then he chose the Data menu’s Text option to

![Figure 5.11 The Pie Chart gallery](image-url)
set up the format for the new series. This series he named “Canard Direct/Indirect” (Figure 5.13).

Then Mark entered the names and values of the new categories, “DIRECT” and “INDIRECT.” From the Pie Chart gallery he selected the sixth type, in which pie segments have different shadings and are labeled with their percentage contributions to the total (Figure 5.14). Then he clicked the Plot Series box and got a new pie chart (Figure 5.15).

Now it was time to print out the two charts for use in his presentation to the vice-president. Mark pulled down the File menu to choose the Print option, then selected the first series and printed it out in the same way. The two charts are shown in Figures 5.16A and 5.16B.

One of Chart’s great advantages is that it is very easy to learn the basics so that you can easily produce simple charts of your own. You can quickly learn how to input a data series and have the values plotted in a format of your choosing. Chart also has many more advanced capabilities, including many format control options. Practice and experimentation will reward you with the ability to produce high-quality charts easily.
Figure 5.13 Naming a new text series

Figure 5.14 Selecting a chart type from the gallery
Eldon Johnson is director of marketing research for a manufacturer of video arcade games. His department studies consumer preferences and performance on the firm's arcade games. They recently completed a major study of user behavior in early games. When his research staff presented him with the raw data on user performance during the first eight novice games, Eldon decided to use Microsoft Chart to get a visual representation of the data.

Thirty novice players had played a sequence of eight games on the company's new arcade game. Using Chart, he quickly set up a data series called "MEAN SCORES BY GAME" and entered the eight mean scores on the games. He chose to plot the series as a column chart (Figure 5.17). As he expected, the chart seemed to show a relatively steady progression of game scores. Did this mean that every user progressed relatively steadily, he wondered? What if only a few users were really getting better? He asked his
assistant to give him three sets of figures—the average scores for each game for three groups: the lowest-scoring ten users, the highest ten, and the middle ten.

Eldon entered these three sets of game scores as three separate series in Chart. After entering the Low Group, he clicked the mouse pointer on the window labeled “New Series.” This made a new window active, which he labeled “Mid Group.” After entering the eight average game scores for the
members of this group (Figure 5.18), he clicked on the New Series window again to create a High Group window. He filled in the game scores for that group, and then chose one of the Scatter Plot options from the Gallery menu. Chart automatically plotted all three sets of data (Figure 5.19).

Looking at the scatter plot, Eldon immediately saw that his fears were well founded. It looked as though only a third of the users were actually improving. Was it possible that two-thirds of novice arcade game users would be so discouraged by their failure to improve that they would not continue to feed their spare change to the machine? This was clearly important enough to discuss in his weekly meeting with the manager of the Product Development Group.

Eldon wanted a larger copy of the chart to use in the meeting. He clicked on the chart window to bring it to the fore, and dragged the title bar to the upper left corner of the display (Figure 5.20). Then he used the size box to enlarge the screen. The next step was to select the whole chart for resizing in the newly enlarged space. In the Chart application, each of the parts of a chart is an object that can be manipulated. An object such as the vertical axis can be

Figure 5.17 Column chart, mean scores by game
Figure 5.18 Adding a new data series—the Mid Group

Figure 5.19 A chart with three series of data
selected by clicking on it. It changes appearance to show that it is the active object, by displaying two small circles at either end of the axis.

Since Eldon wanted to enlarge the whole chart, he typed Command-A, the shorthand command for the Select All menu item in the Chart menu. Eight small black rectangles appeared around the borders of the chart area. These rectangles are "handles" that the mouse pointer can grab onto to change features of the selected item. Eldon pointed to the handle at the lower right corner of the chart and dragged it down and to the right (Figure 5.21). When he released the mouse button, the chart was replotted in a larger size. Part of the chart had disappeared off the top of the screen, so he pointed between two of the handles and dragged the whole chart down.

Looking over the enlarged chart (Figure 5.22), Eldon decided he wanted a larger title and more prominent axis labels. He clicked the mouse pointer on the Three Groups title, and it was enclosed by eight handles. From the Format menu, he selected the Text option (Figure 5.23). An elaborate dialog box offering control of the appearance of text objects appeared (Figure 5.24). He chose to change the chart title to New York font, bold, and medium size. Using
Figure 5.21 Dragging a size handle

Figure 5.22 The enlarged, repositioned chart
Figure 5.23 A selected text item

Figure 5.24 The chart text options dialog box
the same procedures, he selected the two axes titles and changed them to the Chicago font. The resulting chart (Figure 5.25) looked about right to Eldon, and he used a printout of it in the meeting to present his point quickly.

**USING MULTIPLAN AND CHART TOGETHER**

It is an easy matter to use Chart to present important results from a Multiplan spreadsheet. From Multiplan, select a portion of a row or column to be treated as a data series in Chart. Drag through these cells to select them, and then Copy them to the Clipboard. Each series must be a set of contiguous data. If the row or column you want to take the series from has gaps or unwanted values, construct a new, contiguous section of the spreadsheet for the values you want to use in Chart, making sure to set the value of each cell with a reference to the old spreadsheet cell, rather than with a particular numerical value. That way
Figure 5.26 The Chart menus
you will be able to update the spreadsheet later, and the cells that Chart references will change appropriately.

Leave Multiplan and start up the Chart application. Specify the data type for the series, then Paste and Link the Clipboard contents into the new chart series. Thereafter, whenever that Chart document is opened, you will be asked to insert the disk with the Multiplan document. The plotted chart will always reflect the most recent changes to the spreadsheet.

Chart is a powerful tool, the basics of which are easily learned. Within a few minutes of starting your Chart disk, you can easily create usable charts of many different types. If you will be using Chart frequently, or if you must make presentation-quality graphics, set aside some time for experimenting with Chart's many formatting functions, which are distributed through the Data, Gallery, Chart, and Format menus.

As you grow in expertise, you should find yourself making extensive use of the command keys in place of some of the pull-down menu selections. To help you graduate to the use of these key combinations, Figure 5.26 presents the set, as they are presented in the menus themselves.

**FURTHER READING ABOUT MICROSOFT CHART**


Lambert had access to the Microsoft applications programs during their development and probably contributed, through his comments to its programmers, to its final form.
CHAPTER 6

OUTLINES

USING THINKTANK 131
PRINTING OUTLINES 140
MOVING OUTLINES TO TEXT DOCUMENTS 142
THINGS TO DO WITH THINKTANK 143
ThinkTank 128, from Living Videotext, is one of the easiest-to-learn practical applications for the Macintosh. ThinkTank helps you create and manipulate outlines. Of course, you could use a word processing application like MacWrite or Word to make outlines, but ThinkTank makes it easy to manipulate outlines in ways that are not possible with word processing programs.

**USING THINKTANK**

To start a new outline, simply open the ThinkTank icon, either by double-clicking it or by selecting it and using **Open** from the File menu (Figure 6.1). If you want to examine or edit an outline that already exists, just open its icon and the ThinkTank application will be opened automatically.

When you begin from the ThinkTank application, you are presented with a minimal outline—one that consists of only a single heading, the word “Untitled” (Figure 6.2). You will notice that the word is prefaced by a minus sign. This symbol means that a heading has no subheadings. If a heading is preceded by a plus sign, then the heading is understood to have subheadings.

I used ThinkTank to create an outline for this book. The first step was to enter the main heading, or title, of the book (Figure 6.3). The heading “untitled” was already selected, so I just backspaced over it to erase it and then typed in the title of the book. A ThinkTank outline can have only one main heading. After you type the main heading and the Return key, ThinkTank automatically indents the next line. This happens only after you enter the main heading. Thereafter, typing **Return** creates a new line at the same indentation level as the one above. You have to let ThinkTank know if you want the new line to be indented farther or less than the previous heading.

I typed in tentative titles for the chapters planned for the book (Figure 6.4), following each one with the Return key. As I did so, ThinkTank showed that the heading I was currently entering was the selected one by enclosing it in a rectangle. None of the main subheadings have subheadings themselves at this point, so they are all preceded by minus signs.

ThinkTank uses the Mac’s Monaco font, which is a non-proportional font, like the type produced by ordinary typewriters. Every letter takes up the same amount of space on the line. The FontSize menu lets you choose how large you want the letters to be. In these examples, a 12-point font is used, but the 9-
Figure 6.1 Opening ThinkTank

Figure 6.2 The empty ThinkTank outline
Figure 6.3 A main heading

Figure 6.4 Second-level headings entered
point Monaco font can also be used. Different font sizes permit different amounts of information to be displayed in the ThinkTank window (see Figure 6.5).

In ThinkTank, the text insertion point is shown by a vertical bar. In the above figures, this bar is at the right of the selection rectangle that encloses the current heading. In these figures, the insertion pointer shows up as a thick line on the right side of the selection rectangle (Figure 6.6). Naturally, ThinkTank lets you insert anywhere in a heading line. All you have to do is point to the letter you want to insert in front of and click. The insertion pointer appears as a vertical line before the letter. You can now type in new text at this point, or backspace over old text.

ThinkTank doesn't have all the text-editing features you are familiar with from other Macintosh text applications. You can't select a word or phrase by dragging through it, for example. Only headings can be selected, not letters or words. Most Mac users will find that this restriction takes a little getting used to. The hardest thing to remember is where to click to get a new insertion point. Don't point to just before the letter you want to insert in front of, as you would with other applications. Point to the letter itself when you click. The new insertion point will be just before the letter.

When I had established my chapter headings, I went back through the outline, adding subheadings for each chapter. I first selected the Introduction's heading, "What Makes Your Macintosh Special," by clicking on the line with that heading. I pressed the Return key, and a new, blank heading line, preceded by a minus sign, appeared immediately below. This line was at the

<table>
<thead>
<tr>
<th>Font Size (Points)</th>
<th>Characters/Line</th>
<th>Lines/Window</th>
<th>Characters/Window</th>
</tr>
</thead>
<tbody>
<tr>
<td>9</td>
<td>82</td>
<td>25</td>
<td>2050</td>
</tr>
<tr>
<td>12</td>
<td>70</td>
<td>17</td>
<td>1190</td>
</tr>
</tbody>
</table>

Figure 6.5 ThinkTank window capacity
same level as the chapter headings. Since I wanted it to be a subheading under
the “Introduction” line, I typed Command-R to specify that the new
(currently selected) line should be indented one stop to the right. At the same
time that the new line was indented to the right, the minus sign preceding the
chapter heading line above turned into a plus (Figure 6.7). I typed the new
heading—“Techniques for Using MacPaint”—and created three subheadings
for it. Here I typed Command-R when entering the first of these subheading
lines (“Transferring Material Between Documents”) to set up another level of
indenting. When I entered the next line, I didn’t have to indent again, since I
wanted “The MacPaint Print Catalog Feature” to be at the same level.

To reverse-indent, one can simply type Command-L, specifying that the
selected heading will be positioned one stop to the left. One way in which
outlining with ThinkTank is different from outlining with a word processing
application is that the subheadings are all tied to the more major heading
above. For example, when a heading is moved left or right, all its subheadings
move with it.
Another way that a ThinkTank outline behaves differently from an ordinary text outline is that the ThinkTank user can selectively hide or "bury" portions of an outline. After I had finished outlining the Introduction, I wanted to conceal that level of detail while I worked on the structure of Chapter 1 in the outline. I pointed to the Chapter 1 heading line ("Pictures") and double-clicked. Immediately, the detail levels below that heading disappeared (Figure 6.8). Of course, the subheading information was not lost. To see it again, all I had to do was double-click on the heading line again. The ability to close and open levels of topic detail make ThinkTank an appropriate tool for intellectual planning. The user can alternately pan back to get the big picture of the problem and zoom in to focus on details.

Inserting and deleting are performed in ThinkTank in much the same way that they are in other Macintosh applications. To insert or delete letters, as we have already seen, you simply select a text insertion point and type in new text or backspace out the old. If you want to delete a complete heading and all its subheadings, you select the heading (by clicking anywhere on its line) and use the Cut option from the Edit menu. The keyboard command shortcut, Command-X, is available for cutting. The deleted heading (and all its
Using ThinkTank 137

subheadings) is on the Clipboard at this point, and you can move it to a
different part of the outline if you want to, using the Paste function.

In ThinkTank, moving around through your outline document is a little
different from other applications we have examined. Instead of using scroll
bars, ThinkTank has eight distinct scroll areas around the document window
(Figure 6.9). Clicking in any one of these areas will move the window one
“unit” in that direction. In the case of the vertical movements—clicking in the
gray area at the top or bottom of the window—a unit is one line of text.
Clicking in the gray areas to the left or right of the window moves the display
one “heading stop” in the chosen direction. Clicking in the gray corners makes
the display move diagonally, one unit horizontally and one vertically. To
move a whole page in a chosen direction, point to the appropriate scroll area
and double-click.

It is easy to revise an outline by changing the indentation of headings or by
adding or deleting headings. After entering the subheadings for Chapter 2,
“Words,” I decided that I wanted to add to the outline the acknowledgments
and other materials that would precede the Introduction. I simply scrolled
back to the beginning, and inserted the new material (Figure 6.10).

Figure 6.8 A “closed up” heading
Figure 6.9 The scroll areas

Figure 6.10 The new heading: "O. Preliminaries"
The ThinkTank Extra menu gives you some simple sorting and searching capabilities. To sort all the subheadings of a heading, you must first select the heading. Then pull down the Extra menu to Sort and release the mouse button (Figure 6.11). Now the subheadings are rearranged in alphabetical order (Figure 6.12). Sorting works about as you would expect, with numbers being sorted after letters. Because the sort is alphabetic rather than numeric, a heading that begins with the number 6 will be sorted after one that begins with the number 193, because 1 precedes 6 in the alphabet. If you plan on sorting headings that begin with numbers, be sure to include the same number of digits for every number and use leading zeros before the smaller numbers. Then your “006” will be sorted before “193.”

The Extra menu’s Search option lets you carry out operations similar to the Find and Replace options of MacWrite. You can find every instance of a word in an outline segment—a portion under a heading. If you choose, you can replace each instance with something else. See Figure 6.13.

There is more than one way to move a heading to a new position. In addition to cutting and pasting, you can simply drag a heading (and its subheadings) to a new location in an outline. First, just point to the heading you want to move to another place in the window. Depress the mouse button
and leave it down. A gray border will surround the heading and its subheadings as a unit, and the pointer will change to a slim pointed horizontal shape. Position the pointer between the two headings where you wish to put the enclosed material and release the mouse button. Bingo! The selected portion of the outline has been moved.

**PRINTING OUTLINES**

To print a portion of an outline, first select the heading for the part of the outline you want to print. If you want to print the whole outline, select the main heading. Then pull down the File menu to the **Print** option (Figure 6.14). A dialog box will open to offer print setup options (Figure 6.15). When you've finished choosing print options, click the “OK” button. The selected portion of the outline will be printed.
Figure 6.13 A and B Searching for instances of "mouse"
MOVING OUTLINES TO TEXT DOCUMENTS

After creating an outline, you may want to transfer all or part of it to a word processing application document. Perhaps you would like to begin the document with the outline, or you might want to use the outline headings as headings in your text. The first step is moving the outline or outline portion to a text document. While in ThinkTank, select the heading you want to copy into a word processing document file. Naturally, if you want to copy the whole document, you will choose the main heading. Then type Command-C or choose Copy from the Edit menu. At this point, the Clipboard will contain the selected heading and its subheadings. (If you don't believe it, choose Show Clipboard from the Edit menu to have a look at the copied material.)

Quit ThinkTank, and start the word processing application by opening the document into which you want to insert the outline segment. Select an insertion point and type Command-V or choose Paste from the Edit menu. A copy of your outline will appear (Figure 6.16). You can use the word processing application to add multiple fonts, styles, and sizes to your outline if
you want. You can even paste an outline into a MacPaint document and elaborate it graphically (Figure 6.17).

THINGS TO DO WITH THINKTANK

ThinkTank is not limited to constructing outlines for books. Unlike many other Macintosh applications, ThinkTank's strength lies not in creating finished products—printed outlines. Instead, it is a tool for encouraging organized thought. ThinkTank encourages its users to separate a problem into components. It makes it easy to expand an issue into a very detailed outline or to collapse it back to the major headings in order to keep track of the big picture.

Project managers who do not need the full panoply of project planning and tracking features provided by MacProject may want to consider using ThinkTank for project planning. You can outline the major stages of a project,
11. Programming with MacPascal
   - Intro
   - Structured Languages and the Realities of Personal Programming
   - MacPascal Fills Apparently Competing Requirements
   - Programming Kinetic Lines
     - Specification
     - Stepwise Refinement
       - Single line
       - Elaborating dummy elements in program
       - Adding user control

Figure 6.16 Outline pasted into a MacWrite document

Figure 6.17 An outline elaborated in MacPaint
with subheadings for all the contributing activities for each stage. Re-editing the outline as your project goes on gives you a simple means for tracking project progress.

Anyone who must plan and give oral presentations should consider giving ThinkTank a try. Writing out the complete text of an oral presentation in advance is not usually the best way to prepare a talk. In fact, having a complete transcript of an oral presentation to read from is a certain way of making the presentation colorless and hard to listen to. On the other hand, the planning that must accompany writing can improve a talk immeasurably. Using ThinkTank can help you plan an oral presentation at many different levels of detail without actually having to write it out. The printed outline can even serve as your lecture notes.
CHAPTER 7

PICTORIAL DATA BASES

FEATURES OF FILEVISION 147
MANIPULATING THE DISPLAY 150
PRINTING LISTS 154
THE RANGE OF FILEVISION'S USES 157
BUILDING A DRAWING FILE 161
LINKING DRAWING FILES 170
The Macintosh’s computational power and graphics capabilities made possible new kinds of software not previously available on other computers. Mac users have been given the ability to manipulate information, especially non-verbal information, more directly than is possible with other computers.

FEATURES OF FILEVISION

One of the best examples of such innovation is Filevision from Telos Software Products. Filevision makes it possible to store data in files that refer to pictures you create. Put another way, with Filevision you can draw pictures that have objects with associated data.

The documents you create with Filevision are called DRAWING FILES. These pictorial data bases include both drawings and data forms that describe the elements in the pictures. Menu options let you instruct Filevision to pictorially highlight objects in your drawings that have certain data characteristics. Or you can print out complete or selective lists of data about drawing file objects. If the stored data includes names and addresses, you can even do mailing labels with Filevision.

Filevision drawing files are a useful way to enter data you would like to arrange topographically—as though it were part of a map. If you have a collection of objects stored in a particular arrangement, you can use Filevision to record both where the objects are located and what their alphanumeric data characteristics are. For example, Filevision can be used to store an organization chart together with facts about the departments represented and their managers.

Figure 7.1 is a drawing of a map of the Los Angeles area I made with Filevision. It includes objects of four types: background objects (such as the coastline and the mountains), freeways, cities, and Olympic Games sites. Unlike a picture created with MacPaint, each object depicted in this scene is recognized as an object by Filevision. If you point to an object and click, its name appears just below the window with the drawing.

In Figure 7.2 a different Filevision drawing file is on the screen, depicting a closeup of the area of Los Angeles in which the Olympic swimming competition was held. A rectangular object has been selected by pointing and clicking, and it shows up with nine small white rectangular “handles.” Just as in MacDraw, a graphic object’s handles let the user reposition or stretch the
Figure 7.1 A Filevision Olympic map

Figure 7.2 The USC 1984 Olympic site
object using the mouse. The text at the bottom of the screen informs us that the object is the natatorium. The type of object is a Sports Facility.

Each object in a drawing file has associated with it a set of alphanumeric data that the user can access by clicking the Info button on the bottom of the screen or by double-clicking the object. Each type of object has its own standard information form that gives the user access to information about objects of that type. When a Filevision user defines a new type, the layout of the information form for all objects of that type is determined. Once the natatorium has been selected, if the user clicks the Info button, the form appears, showing that three types of events took place at the facility: swimming, diving, and synchronized swimming (Figure 7.3).

Filevision has built-in facilities to support two major kinds of work with drawing files—creating and editing the drawings and their associated data forms, and using the completed drawing files to access and understand the stored information. Like MacDraw—described in Chapter 10—Filevision is an object-oriented graphics composition application rather than a pixel-oriented application like MacPaint. Unlike either MacDraw or MacPaint, it is also an information-filing system.

Figure 7.3 The information form for the natatorium
Consider a few of the capabilities Filevision provides to support information browsers—those who use preexisting drawing files to get information. Eight menus give access to one of the largest sets of information manipulation commands available for a Macintosh application (Figure 7.4). Two menus of special interest to the information browser are the Tinker and File menus. The File menu provides many selective print options. The Tinker menu provides for temporary changes in the appearance of a drawing to bring out some aspect of the data. It lets you tinker with the drawing. The Tinker menu gives the browser control over object-oriented displays in ways that other applications don’t provide.

MANIPULATING THE DISPLAY

When authors create a drawing file, they define the types to be used in the file. Every pictorial object must belong to one of these defined types. Filevision lets users manipulate information based on type or on information stored in data forms.

Controlling the Appearance of Objects

All the objects of a particular type can be hidden in a drawing. First you make that type the currently selected type, either by pointing to an object of that type and clicking, or by choosing it from the Types menu. Then you simply choose the Hide These option from the Tinker menu (Figure 7.5), and all the cities shown on the drawing disappear (Figure 7.6). Naturally, a Show These option then becomes available to make the objects that represent cities reappear in the drawing.

Highlighting

Control over the appearance of objects is not limited to making all the objects of a type appear or disappear. One can also Highlight objects. Highlighting is a process that “grays out” unselected objects in a drawing and adds special shadowing to the objects that are highlighted. The result is a display in which
Figure 7.4 Filevision's menus
the highlighted objects stand out quite strongly. You can choose to highlight all the objects of a type, but a much more useful feature is the ability to highlight only those objects that have certain data characteristics which you specify.

If you choose the Highlight Some option when the City type is selected, you can specify the constraints you want to place on the highlighted objects. A special form fills the screen, asking you to describe the conditions for highlighting a city. In the example in Figure 7.7, only those cities with two or more Olympic sites are to be highlighted. As soon as the user clicks the Done button on this form, the drawing is redisplayed, but with almost all the objects grayed out. According to the data stored with this drawing file, only the cities of Los Angeles and Mission Viejo had more than one Olympic site in the 1984 summer games, so only those two objects are highlighted (Figure 7.8).

Complex Highlighting Specifications

The Highlight Some option permits more complex statements of conditions. If you wanted to know which Olympic sites were in use throughout the period of the Olympics, you could ask to see those for which the first day is July 29 and
Figure 7.6 The same drawing, with cities hidden

Figure 7.7 Filling out the Highlight Some "Conditions" form
the last day was August 12. The process of filling out a conditions form does not require that you type in the conditions phrases (such as "First Day is less than or equal to") seen in the condition boxes on the right in Figure 7.9. These English phrases are composed by clicking the selections in the box to the left. The only items you have to type are the limiting values that the condition refers to, such as "840812."

In Figure 7.9, dates are expressed as numbers consisting of the last two digits of the year, followed by the month (here "08") and then the day of the month. Expressing dates this way, as single numbers that represent year, month, and day, makes it easy for Filevision to sort on date fields. If you express dates with strings such as "8/12/84," only the first digit will be used in sorting, because the "/" is understood to mark the end of the number.

**PRINTING LISTS**

Information stored about the objects of a given type can be printed out from Filevision by using the print options available under it’s File menu. Like most other data-base manipulation systems, Filevision stores data in RECORDS. Each record has a number of FIELDS, the predefined parts of a record. The list of printing options of the Filevision application lets you decide which fields should be printed in a list. The formatting page for the list lets you choose which of a record’s fields should appear in the list, where on the line, and how much space to allocate for each one (Figure 7.10). You can independently specify whether the material from the field should be right-justified, left-justified, or centered in its column.

The list-formating options also let the user decide what data field will determine printing order. In this example, Olympic sites are to be printed according to the order of their first day of use. The field that determines the order of printing does not have to be the field in the first column of the printout.

A second printing option lets you print more than one line per record. You can use Print info to print all the information about the objects of a type. A formatting page, similar to the one used to print lists, lets you choose options that control the printout. The Print Info options in Figure 7.11 specify that a set
Figure 7.8 "Los Angeles" and "Mission Viejo" highlighted

Figure 7.9 A more complex conditions form
Figure 7.10 The list-formatting page

Figure 7.11 Formatting options for Print Info
of Olympic swimming records are to be printed in an order based on the record time for each of the events.

Other printing options include printing the display and printing sets of labels based on a drawing file's data.

THE RANGE OF FILEVISION'S USES

Since Filevision is a new kind of software, one not previously available on microcomputers, it is likely that users will come up with uses its developers never imagined. Here are some uses that they have thought of.

Intelligent Maps

The Los Angeles Olympics drawing file described in Figures 7.1 through 7.11 is an example of an "intelligent map." A Filevision map lets the user get a lot more out of it than is possible with ordinary maps. Filevision maps such as

![Figure 7.12 A map of the United States, with twelve states highlighted](image)
those in Figures 7.12 and 7.13 can change their appearance to exclude certain types of information and to highlight others. A Filevision map can be used to generate lists. If your information form includes name and address fields, you can use the Print Labels option to produce mailing labels.

Intelligent Illustrations

Other possibilities for Filevision include "intelligent illustrations." For example, a drawing of the parts of the human eye (Figure 7.14) could employ data forms that let students formulate their own questions about how the objects in a drawing are interrelated, and get the answers in both pictorial and verbal forms. Artistically inclined users can create striking images as technical illustrations, as in Figure 7.15. Technical data for each object is available on command from its information form.
Figure 7.14 The human eye

Figure 7.15 Two flowers
Art Analysis

Filevision can be used for art analysis. For example, a sketch of an artistic work can be stored as a drawing file (Figure 7.16), together with whatever data is desired about the elements of the artistic work. You can use Filevision's power to highlight, to sort, and to compare the features of such elements in the pursuit of new critical insights.

Everyday Uses

On a more mundane level, Macintosh owners may choose to use Filevision to produce drawing files that relate to everyday aspects of their lives. A map of one's apartment may be an aid to organizing the placement of furniture (Figure 7.17). And you've probably been admonished to keep a list of your valuables. With Filevision you can keep a map of your living space in which each valuable has its own data form, filled with all the information that the police and the insurance companies would like you to have on hand. Or, if you

Figure 7.16 Botticelli's Primavera as a Filevision drawing
are a collector, you might choose to keep a drawing file that not only stores relevant data about each item in your collection, be it coins or rare wines (as in Figure 7.18), but also shows where each item is located.

**BUILDING A DRAWING FILE**

Filevision not only provides powerful information-browsing capabilities, it also provides the tools for creating and editing drawing files. Most users will want to create their own drawings, populated with objects of types of their own design. These types have data forms of the users' own devising. To see how it's done, follow my construction of the 1984 Los Angeles Olympics drawing file.

I opened the Filevision icon and was presented with an empty drawing. The default type at this point is always called "Background." I wanted to enter a few background elements for the Los Angeles area—the coastline and the
Figure 7.18 A wine collection and the wine data form
mountains. From the palette of drawing tools at the left, I chose the last one, the Freehand Line tool. I positioned the pointer and depressed the mouse button to sketch out the Los Angeles area coastline. As soon as I released the mouse button, a set of nine "handles" appeared (as in Figure 7.19). These little solid rectangles (and one diamond shape) let the user move an object or change its shape.

I wanted to add several elements made with other tools on the palette, such as the Text tool, to this Background object. I also wanted to make several little waves in the area of the Pacific Ocean. Rather than try to sketch the waves freehand, I decided to make a special wave symbol that I could drop wherever I wanted. I called up the Filevision Symbol Editor from the File menu, chose a symbol that I didn't plan on using, and used Clear to erase it (Figure 7.20). The Symbol Editor will remind you of FatBits in MacPaint. It lets you develop up to twenty special icons or symbols for use in each drawing. After I finished sketching my "waves" symbol (Figure 7.21), I clicked the Done button to get back to the drawing page.

I chose the Symbol tool, the fourth item on the palette. I held down the Shift key as I selected the Symbol tool, thereby telling Filevision that I planned
Figure 7.20 Clearing a symbol in the Symbol Editor

Figure 7.21 Finished waves symbol
to add an element to the current object (the coastline), rather than add a new object. The Symbol tool leaves a copy of the current symbol shape wherever one clicks in the drawing. After adding a few sets of waves to that first background element, I held down the Shift key again while selecting the Text tool, the large capital A. I entered the names “Pacific Ocean” and “Santa Monica Bay.”

An ELEMENT is a portion of a drawing created through a single use of a drawing tool. Elements can be combined to make up drawing objects, although some objects consist of single elements. Because I chose to enter all the coastline elements as parts of one object, they are all closely linked together in the Filevision drawing file. When I click on one of them, all are highlighted (Figure 7.22). Furthermore, since only objects (not elements) can have data forms, there can be only one data form to represent the entire object.

I used the Symbol Editor to make a new symbol to represent mountains. Then I added a new complex object consisting of several elements made with the new symbol and the words “San Gabriel Mountains” (Figure 7.23).

My next step was to pull down the Types menu to **Add another**. Filevision presented me with a type definition form. I called the new type “Freeway”
and gave it five data fields (Figure 7.24). Actually, two of the fields, Name and Link, are always automatically supplied by Filevision when you create a new object type. I provided for up to two freeway numbers because of the strange practice in Los Angeles of giving freeways names without regard to their numbers. (The Hollywood Freeway includes portions of Highways 101 and 170, and the Ventura Freeway consists of the rest of 101 and parts of 134.) A third number tells how many Olympic sites are immediately accessed by the freeway.

I could have included many more fields. A type can have up to thirty data fields, and each data field can have a name up to fifteen characters long. The fields themselves can be much larger, if you desire, up to a limit of 2,000 bytes for a single object, distributed among that object's fields as you desire. You can have up to 999 objects in a drawing file, but they can't all be 2,000 bytes large. The maximum drawing file size is 132K bytes.

Using the Symbol Editor, I created two new symbols to represent interstate highway and California highway symbols. Each freeway in the drawing consisted of a number of elements, including one or more freehand line segments, one or more highway symbols, and one or more text elements positioned right after a highway symbol. In Figure 7.25, clicking on the line
Building a Drawing File 167

that represents the Ventura Freeway has resulted in the line acquiring handles and in the object's two symbols and two text elements becoming highlighted.

The next step in preparing my Olympics drawing file was to create a new object type, the Olympic Site. The data fields associated with an Olympic site would be its name, its link, its event type, the first and last day of its use during the Olympics, and a comment field, as shown in Figure 7.26. To represent the sites, I used symbols that represented the different sporting events that took place at each. I used the Symbol Editor to construct sixteen new symbols, representing sports from archery to tennis (see Figure 7.27). (In case the symbols in Figure 7.27 seem a bit cryptic, here they are in order: Row 1—archery, track and field, baseball, basketball; Row 2—cycling, equestrian, fencing, football (soccer); Row 3—gymnastics, field hockey, water polo, swimming; Row 4—shooting, weight lifting, wrestling, tennis.)

The next step was to add all the Olympic sites to the drawing file, along with the information about each one (see Figure 7.28). By clicking on any object in the drawing, one can learn what the type of the object is and its name. Clicking on the symbol of the wrestlers, we learn that it represents the Anaheim Convention Center. To get more information about an object or to edit information, you simply click the Info button. Figure 7.29 shows the data
Figure 7.25 The freeway objects entered

Figure 7.26 The "Olympic Site" definition form
Figure 7.27 The Olympic sports symbols

Figure 7.28 The drawing file with Olympic sites
form for the Loyola Marymount University site, where the weight lifting competitions were held.

Once a drawing file such as this one has been created and the data has been entered, it can be locked and given to others who need to use the data but who don’t have to edit it. Such users can still Tinker with the drawing file, exploring the significance of the stored data.

**LINKING DRAWING FILES**

Drawing files can be linked to each other. Each site in the Olympic map drawing file can have a link to a more detailed local map for that site. Figure 7.30 shows the drawing file for the University of Southern California site, where the Olympic swimming events were held. This file has not only a map of the USC campus and surrounding area but also a section referencing previous major Olympic swimming records. Each of the swimming figure
symbols is an object with a data form that has information on the current Olympic record for a swimming event. By first selecting the USC site on the global Olympic map and then choosing to Link, the user moves automatically to the new drawing file.

From this new file, it is possible to use the Tinker menu's Highlight some option to ask a question like “What Olympic swimming records are held by athletes from the United States?” Filevision will display a pictorial response (Figure 7.31). If every Olympic site in the higher-level map has a link to such a drawing file with local street information and Olympic records, Filevision can provide the user with a richly interconnected set of drawing files and data.

Since a user is likely to want to get back to the larger map from such a specific drawing file, it makes sense to provide a link in that direction as well. Since nothing in the USC site drawing file serves as a natural representation for the larger map, I added one. I used the rounded rectangle to make a box and added two text elements—a Cairo font arrow and some Geneva font text—to create a box reading “⬆️ L.A. Map” (Figure 7.32). I gave this background object the artificial name “Click Link for Big Map,” which serves as an
Figure 7.31 Highlighted Olympic swimming records

Figure 7.32 A link back to the L.A. Olympics map
instruction to naive users. A complete set of interrelated Filevision drawing files provides a Macintosh environment for exploring the geography and records of the Los Angeles Olympic Games.

Filevision can be used to build richly interconnected files of drawings and data. Although it can also be used as an ordinary filing application—one that can be used to file client data or recipes—its capabilities go beyond ordinary filing. It provides users with new opportunities to use the Mac productively and imaginatively.
CHAPTER 8

PROJECTS

MANAGING MAJOR PROJECTS WITH THE MAC 175
CREATING A SCHEDULE CHART WITH MACPROJECT 175
USING A COMPLETED MACPROJECT SCHEDULE 188
PLANNING AND TRACKING 193
MANAGING MAJOR PROJECTS WITH THE MAC

The Macintosh provides many tools that contribute to the accomplishment of major projects. In a sense, this entire book deals with how you can use the Mac to carry out major portions of projects. Word processing programs contribute to the production of project proposals and reports. Chart, MacPaint, and MacDraw help the project manager develop presentation-quality graphics. MacTerminal lets a project's staff tap into the resources of mainframe computers. Other special-purpose applications, such as accounting and data base systems, can contribute to project progress.

Several Macintosh applications are useful project-planning and project-management tools. Multiplan spreadsheets, as described in Chapter 4, can play a very useful role in planning the financial aspects of a project. A project budget spreadsheet, such as the one presented for the Canard Widget project, does not end its usefulness when the decision is made to fund the project. A project manager can continue to use the spreadsheet to track project expenditures, altering any of the factors that do not conform to the projected expenditure rate. Multiplan will automatically calculate the consequences of the difference for the entire spreadsheet. Be sure to use a copy of the original planning spreadsheet for this process of tracking expenses. That way you can always compare the currently tracked expenditures with the original budget plan.

ThinkTank (covered in Chapter 6) is another tool that can be effectively put to use in project planning. You can outline an entire project with ThinkTank, treating each of the project's major milestones as a separate major outline heading. All the task elements that contribute to each of these milestones can be treated as subheadings. Since ThinkTank makes it easy to collapse and expand outlines, it is a good tool for alternately stepping back to see the big picture and for selectively expanding task descriptions to study particular portions of the project.

CREATING A SCHEDULE CHART WITH MACPROJECT

Apple Computer's MacProject is a specialized tool for project management. Using MacProject, you can draw a graph that portrays all the tasks of a project and their dependencies. You can enter information about who will perform
each task and what other resources the task requires. MacProject uses the data you enter to produce six different types of charts and tables, which present information about the project in different ways.

The printout of a MacProject schedule chart looks like that in Figure 8.1. Each rectangle represents some task that is part of the project. The text in a rectangle gives the name of that task. Earlier tasks are placed to the left of later ones in the chart. Where two tasks are connected by a line, the task to the right depends on the completion of the task at the left end of the line. A line denotes a dependency—the task on the right cannot begin until the task on the left is completed. The thicker boxes, connected by thicker lines, are the project’s critical path. This is the sequence of tasks that determines how long the project

Figure 8.1 A printed schedule chart
will take to complete. Any unexpected delays in this portion of the project will result in a delay of the final project milestone. Small delays along other paths may not result in project delays.

After you have entered a schedule chart, together with some information about the tasks and who will work on them, MacProject can produce many different charts and tables that will help you plan the project and track its progress. To see how to enter a schedule chart and other project information, consider an example.

Dire Wolf Software, a small software development company, used MacProject to develop the chart in Figure 8.1 after being awarded a contract by JPI Aerospace, Inc. Dire Wolf was to produce a major software application for JPI. The chart in Figure 8.1 is a plan for carrying out the first stage of the contract, the software design phase.

Aaron Lightfoot was the Dire Wolf project manager for this contract. His preliminary design for the product, completed as part of the contract proposal process, had divided the design into several partially independent sequences of tasks. He began the project-planning process by entering these component task sequences.

He first selected Invisible Grid from the Layout menu. Like the grid in MacPaint, this feature makes it possible to align graphic objects on the screen. MacProject has only one graphics tool, the rectangle. You drag a rectangle on the screen by positioning the mouse at the point where you want the upper left corner of the task rectangle, depressing the button, and dragging to the lower right. Then release the button, and the box appears, with a centered text insertion point. This box represents a project task. Don't worry if the box is not exactly the size you want or is not in exactly the right place. You can easily adjust it later.

Text entry follows normal Macintosh conventions. If you want to use different fonts, font sizes, or font styles, simply choose from the Fonts and Style menus. You can drag through text to select portions, and you can Cut, Copy, and Paste text, just as in other Macintosh applications.

A box with a rounded rectangle shape is interpreted as a milestone rather than a task. A milestone is an event, such as the start of a project or the delivery of a product. To create a milestone in a chart, first enter the milestone as an ordinary task rectangle. While it is still the selected rectangle, pull down the Task menu to select change to Milestone (or type Command-M). The selected rectangle will become a rounded rectangle, denoting its new status as a milestone rather than a task (Figure 8.2).
A useful method for adding new task boxes to a project is to use the **Duplicate** option in the Edit menu to make a copy of a selected rectangle. To select a task, whether for duplicating, copying, cutting, or whatever, click on the rectangle's border. After you have made a duplicate, position the duplicate in an appropriate location and then make appropriate changes in the text of the task box to label it (Figure 8.3).

Since **Duplicate** is a frequently used function, it has a Command key equivalent, **Command-D**. As you become expert with MacProject, you will find that you make increasing use of the command keys for greater speed in developing new project charts. Figure 8.4 shows the set of MacProject menus. Note the Command key equivalents for options in the Edit, Task, and Style menus.

Sometimes, in the process of entering text in a task box, you will discover that the size of the rectangle is wrong. Select the box by clicking on any edge. The eight handles that surround it are its size handles (Figure 8.5). You can drag these to reshape the rectangle. The four handles in the centers of the sides are for dragging those sides inward or outward. The four corner handles make...
it possible to drag two sides at once, so that you can change the box’s size diagonally.

Moving a task box is as easy as changing its size. First select the box by clicking on the edge of the rectangle. Then drag from a point on an edge that is not a handle. The outline of the task box will follow the mouse pointer to the new location.

To express a dependency, you drag a line from the task that must precede the dependent task. Start by positioning the mouse pointer well inside the preliminary task box and drag the pointer to the inside of the dependent task box. A line will follow the pointer’s path. When you release the button, the line will automatically reposition. Figure 8.6 shows a portion of Aaron’s schedule chart after he drew in some dependency lines.

Aaron chose to enter most of the task boxes before setting up dependencies in his chart, but you can build your schedule chart in whatever order you choose. The scroll bars, at the right and bottom of the window, let you move to any part of the project chart.

If you accidentally overshoot your destination rectangle and release the mouse button when you are past the dependent task box, MacProject will

Figure 8.3 Using Duplicate in building a schedule chart
draw a new task box at that point. To get rid of it and its dependency line, simply select it by clicking on an edge and type Command-X to cut it.

It is possible to select more than one task box at a time, if you want. Just depress the Shift key while dragging out a dotted-line rectangle that encloses the entire area with the tasks you want to select (Figure 8.7). The newly selected tasks will all appear in inverse, as in Figure 8.8. At this point they can be dragged as a group to reposition them without changing their relative positions, just like a group of icons on the Mac desktop.
Creating a Schedule Chart with MacProject

Figure 8.5 Preparing to drag out a larger size for a selected rectangle

Figure 8.6 Drawing dependency lines
Figure 8.7 The shift-dragging method of selecting

Figure 8.8 A set of selected tasks
Creating a Schedule Chart with MacProject

After entering the sequence of tasks in the schedule chart, you can enter task information. You need to tell MacProject how long you expect each task to last and who is responsible for the task. First, however, you must let MacProject know what you mean when you say that a task will take so many working days. MacProject needs to know how many hours of work are in a working day and how many working days are in a month.

If you pull down the Dates menu to the Calendar option, you are presented with a clock dial and a calendar (Figure 8.9). By clicking on the clock dial, you determine how many working hours MacProject will assume you have in the course of a working day. By clicking on the days of the calendar, you can toggle individual dates back and forth between working days (black numerals on white) and days off work (white numerals on black). If you click on a day-of-the-week column header (S, M, T, W, T, F, or S), you toggle all the dates that fall on that day of the week to working or nonworking days.

By clicking on the month and year indicators at the top of the calendar, you can select what month of what year you want to tell the system about. The Dire Wolf contract with JPI Aerospace began in February 1985, so Aaron
made the appropriate adjustments to reflect Dire Wolf's policies for working days and working hours.

After clicking the OK button in the Calendar window, Aaron scrolled the schedule chart back to the left and selected the start milestone, the rounded rectangle labeled "Contract Award." From the Dates menu, he chose Set Earliest Start (Figure 8.10). The new window let him specify this date as February 4, starting at 9:00 A.M. After he clicked the Set button in this window, the window closed up and the schedule chart had the same appearance as before. Now he needed to make the chart display dates.

The Show Dates option in the Dates menu brings up a window that lets you specify up to four dates to be displayed for each task. Aaron clicked buttons that would cause the early start and early finish dates for each task to be displayed at the upper left and upper right corner of the tasks rectangle (Figure 8.11). The date "2/4" appeared at both the left and right top corners of the start milestone, since it is an event of no duration.

To assign duration to each of the other tasks, Aaron used the Task Info window. He selected Show Task Info from the Task menu. This window takes
Creating a Schedule Chart with MacProject

Figure 8.11 Using the Show Dates window

up about one-fifth of the available screen. When you are entering information for a number of tasks, you have to go back and forth between selecting a task in the schedule chart window and entering information about that task in the Task Info window. It makes sense to use the chart window’s size box and title bar to resize and reposition it so that there is room on the screen for both it and the Task Info window.

One by one, Aaron worked through all the task elements he had entered into the schedule chart. As he clicked on a task box, its task information would appear in the Task Info window above. To begin with, the information for each task was all zero (or, in the case of the Resources, all blanks). Typing the Tab key would select the first field of the information for the select task, its duration in days. Aaron entered this number, then hit Tab again to select the next information field for the task, its fixed cost.

In the same way, he entered information about the resources that would contribute to each task. In the case of Dire Wolf software, the resources were the staff members who would carry out the design work. Each was represented in the Resources list by his or her initials and by the percentage of their time that would be devoted to the task during the specified period. (The
Figure 8.12 Entering task information

Figure 8.13 The final milestone date
Creating a Schedule Chart with MacProject

initials CS in Figure 8.12 stand for Carmen Silver, the owner of Dire Wolf and chief technical guru, AL for Aaron himself, LI and PR for Len Ishikawa and Pam Richards, the programmers, PL for Peter Lang, the subject-matter expert, and KM for Kathy Meyer, the writer.)

When the task information had been entered for all the task elements, every rectangle had its start and end dates displayed in the chart (Figure 8.13). MacProject showed Aaron that he should be able to schedule the design project’s milestone—a meeting with the client to get approval of the design—on March 14, 1985.

The critical path was marked on the final schedule (Figure 8.14). Aaron decided that he would have to pay special attention to how the tasks on that path progressed. He was largely responsible for the first two elements (planning the author features and describing the author features)—perhaps he

![Diagram](https://example.com/schedule-diagram.png)

**Figure 8.14** The final schedule chart
could work a few long days and put this portion of the project ahead of schedule at the beginning.

**USING A COMPLETED MACPROJECT SCHEDULE**

As useful as a printed schedule chart is for planning and keeping track of a project, it is only one of the project-scheduling tools MacProject provides. An important concern of project managers is how their resources are distributed. Are the demands of a project too great for the number of people available? The resource chart helps answer this question.

Aaron chose the **Resource Timeline** option in the Chart menu to bring up a screen display of a bar chart showing each staff member’s responsibilities. Figure 8.15 shows the printout he got by selecting the **Print** option from the File menu while in the resource chart display. By glancing up and down the columns, he could check that none of the staff was allocated for more than 100 percent time. To his dismay, he discovered that he himself was scheduled to work 125 percent of the time during two days of the week of February 25. Looking over the resource commitments, Aaron discovered he could ask Pam to shoulder a bit more of the “Sample Author WalkThru” task during this time.

The resource chart is an important tool for checking on the feasibility of a schedule.

MacProject can also be used to plan for project costs and to track those costs during a project. However, first you have to describe the costs of the resources. Aaron used the **Resource Cost Entry** option to bring up a window with the names of all the resources—“CS (50%),” “KM (50%),” and so on. For each staff member, he entered the appropriate daily salary figure, multiplied by the percentage of that person’s time spent on the project. (Fringe benefits and overhead calculations would be performed later with the help of Multiplan.) When Aaron was done, he had a list of staff resources and costs (Figure 8.16).

The **Cash Flow Table** option under the Chart menu brings up a table that shows both periodic and cumulative expenses associated with the project (Figure 8.17). If having an even cash flow were more important than completing the work as soon as possible, Aaron could go back to the schedule chart and set new dates for certain tasks to avoid the high level of activity—and charges—in the week of February 25.
<table>
<thead>
<tr>
<th>Task</th>
<th>Start Date</th>
<th>End Date</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS(50%)</td>
<td>2/4</td>
<td>2/10</td>
<td>Plan User Features, User Menu, User Features Description, Sample User WalkThru, Specification Document</td>
</tr>
<tr>
<td>KM(50%)</td>
<td>2/10</td>
<td>2/25</td>
<td>User Features Description, Author Features Description, Specification Document</td>
</tr>
<tr>
<td>LI(50%)</td>
<td>2/25</td>
<td>3/4</td>
<td>Sample User WalkThru, Prelim Data Structures Description, REVIS ED Data Structures Description</td>
</tr>
<tr>
<td>AL(25%)</td>
<td>3/4</td>
<td>3/10</td>
<td>REVIS ED Data Structures Description, Program Feasibility Analysis</td>
</tr>
<tr>
<td>LI(100%)</td>
<td>3/10</td>
<td>3/15</td>
<td></td>
</tr>
<tr>
<td>AL(50%)</td>
<td>3/15</td>
<td>3/20</td>
<td>Plan Author Features, Coordinate with Library Features, Author Menu, Author Features Description, Sample Author WalkThru, Library Features Description, Specification Document</td>
</tr>
<tr>
<td>PR(50%)</td>
<td>3/20</td>
<td>3/25</td>
<td>Choose Sample Library Elements, Sample Author WalkThru, Library Features Description</td>
</tr>
<tr>
<td>PL(50%)</td>
<td>3/25</td>
<td>3/30</td>
<td>Coordinate with Library Features, Choose Sample Library Elements</td>
</tr>
<tr>
<td>PR(100%)</td>
<td>3/30</td>
<td>3/31</td>
<td>Sample Library WalkThru</td>
</tr>
</tbody>
</table>

**Figure 8.15** A printed Resource Timeline

189
Figure 8.16 Cost of staff resources

<table>
<thead>
<tr>
<th>Resource Name</th>
<th>Cost/Day</th>
<th>Accrual Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>AL (50%)</td>
<td>108.00</td>
<td>Multiple</td>
</tr>
<tr>
<td>CS (50%)</td>
<td>132.00</td>
<td>Multiple</td>
</tr>
<tr>
<td>LI (50%)</td>
<td>84.00</td>
<td>Multiple</td>
</tr>
<tr>
<td>KM (50%)</td>
<td>52.00</td>
<td>Multiple</td>
</tr>
<tr>
<td>PR (50%)</td>
<td>96.00</td>
<td>Multiple</td>
</tr>
<tr>
<td>PR (100%)</td>
<td>192.00</td>
<td>Multiple</td>
</tr>
<tr>
<td>PL (50%)</td>
<td>105.00</td>
<td>Multiple</td>
</tr>
<tr>
<td>AL (25%)</td>
<td>54.00</td>
<td>Multiple</td>
</tr>
<tr>
<td>LI (100%)</td>
<td>168.00</td>
<td>Multiple</td>
</tr>
</tbody>
</table>

Figure 8.17 Cash flow projected for the new applications design project

<table>
<thead>
<tr>
<th>Starting</th>
<th>Costs</th>
<th>Income</th>
<th>Ending</th>
<th>Cumulative</th>
</tr>
</thead>
<tbody>
<tr>
<td>2/4</td>
<td>2292.00</td>
<td>0.00</td>
<td>2/11</td>
<td>-2392.00</td>
</tr>
<tr>
<td>2/11</td>
<td>2697.00</td>
<td>0.00</td>
<td>2/18</td>
<td>-5089.00</td>
</tr>
<tr>
<td>2/18</td>
<td>2336.00</td>
<td>0.00</td>
<td>2/25</td>
<td>-7425.00</td>
</tr>
<tr>
<td>2/25</td>
<td>3048.00</td>
<td>0.00</td>
<td>3/4</td>
<td>-10473.00</td>
</tr>
<tr>
<td>3/4</td>
<td>1296.00</td>
<td>0.00</td>
<td>3/11</td>
<td>-11769.00</td>
</tr>
<tr>
<td>3/11</td>
<td>876.00</td>
<td>0.00</td>
<td>3/18</td>
<td>-12645.00</td>
</tr>
</tbody>
</table>
By selecting the **Project Table** option from the Chart menu, you can get a complete display of all the information MacProject has about a project except the dependencies (Figure 8.18). For most projects, this table will be large enough to span several pages, which you can paste together after you print it.

Another useful scheduling aid is the **Task Timeline** option, again available from the Chart menu. A task chart gives a useful quick summary of the expected order of events in the life of a project (Figure 8.19).

The small project chart we have explored here may be only the start of a much larger MacProject document. The **Set Chart Size** option in the Layout menu lets you increase or decrease the size of the chart (Figure 8.20). Each vertical rectangle in the grid represents a standard piece of printer paper. During construction of a schedule chart, MacProject lets you scroll right over the page breaks between the pages that make up a chart. After you print out such a multipage schedule chart, however, you need to tape it together carefully before you can hang it on your wall.

**Figure 8.18** Viewing the project table
Figure 8.19 A task timeline

Figure 8.20 The Set Chart Size window
PLANNING AND TRACKING

Once the basic information about a project has been entered in a MacProject document, its usefulness has only just begun. You can use the chart to ask “What if?” questions about the project schedule. “What if it takes four more days to choose sample library elements?” “What if we go to ten-hour work days—when will we finish the project then?” Posing and answering such questions is easy with MacProject.

To track an ongoing project, you should always work with a copy based on the original planning document. Keep the original safe, so that you can go back to look at it from time to time and marvel at how optimistic you had been about the schedules.

As a project moves forward, some tasks may take longer to complete than planned. Hopefully, others take less time than was allocated. Using a copy of your original MacProject document, you can find out just what effect each slippage or saving will have on the timing of the project’s milestones. MacProject makes this power available to managers without requiring a long study of an obscure project-scheduling system. By directly manipulating schedule charts and tables, you can use MacProject to track projects of many levels of complexity, including those with schedule charts that would cover a wall of your office.
CHAPTER 9

TALKING WITH OTHER MACHINES

NO MAC IS AN ISLAND 195
FOUR COMMUNICATION METHODS 195
HOW TO START COMMUNICATING ELECTRONICALLY 197
USING MACTERMINAL 200
CUSTOMIZING MACTERMINAL FOR THE COMPUTERS YOU TALK TO 205
TWO LOCAL AREA NETWORKS 209
TWO STYLES OF MACHINE COMMUNICATION 212
NO MAC IS AN ISLAND

An information-processing device would be useless without a way to output the results of its information manipulations. The Macintosh’s two primary means of outputting are its display screen and its printers. Both are extremely useful forms of output. The display of text and graphics on the screen provides an immediate and highly interactive display of information. Printed output provides information in a compact, transportable, and comprehensive form.

A third form of information output from the Macintosh is the link you can establish between the Mac and other computers. Having a link to other computers facilitates collaborative work with your colleagues who use those computers. It is possible to exchange documents with other computer users quickly and easily at remote locations, even thousands of miles away. A link to other computers also makes it possible to distribute your work electronically to people who need access to information you can provide.

A link to other computers is not only an output channel, it is also a new input channel to your Mac. Thus far in this book we have assumed that all information entering a Mac came from either the keyboard, the mouse, or a micro-floppy disk. Receiving information from other computers makes it possible to carry out collaborative work creating or editing documents. If the other computer you are communicating with is a Macintosh, you can exchange any document files, including spreadsheets, project schedules, MacPascal programs, and MacPaint graphics. Two Mac users at opposite ends of the country can jointly develop a project plan, for example, successively modifying a MacProject document and exchanging it for comments and further editing. Two authors working together can send each other revised versions of the same text files. In fact, if the exchange of information is strictly textual, you can easily exchange documents with almost any other computer, not just with Macintoshes.

FOUR COMMUNICATION METHODS

There are four ways in which you can exchange documents with other computers: by micro-floppy disks, by local serial connection, by telephone connection, and by local area network (LAN).
Micro-Floppy Disks

If the other machine is a Macintosh, you can use a micro-floppy disk to exchange information. Any document or group of documents can be copied to a micro-floppy and given to another Macintosh user for his or her use. So long as the other user has the application program that uses that type of document, he or she can use the documents on the disk.

Local Serial Connection

If the other machine is not a Macintosh but is in the same room with the Mac, you can usually connect the two for purposes of communication by using their serial interface ports. In the case of the Mac, this means using the 9-pin connector labeled with the telephone icon. On the other computer you will usually use a 25-pin RS-232-C connector. Naturally, you’ll need a special cable and appropriate software for each machine. More on this below.

If the other computer is an IBM mainframe computer, you will need to use a special piece of hardware called a protocol converter, such as AppleLine. This device lets the mainframe system treat the Macintosh as a special type of terminal (the IBM 3278 Model 2) required by such systems.

Telephone Connection

If the two computers are not in the same room, you can connect them by means of the telephone and a modem for each computer. A modem is a device that converts the electrical signals that a computer sends and receives through a serial port into the signals that a telephone line carries. Each computer must have appropriate software for the communication process (such as Apple’s MacTerminal application), just as in the case of direct connections.

Local Area Network (LAN)

We will all undoubtedly be hearing a great deal about LANs, or local area networks, during the next few years. A LAN is a set of computer workstations and other devices, such as large hard disk drives and fast laser printers,
physically linked by communication cables. Each computer, disk drive, printer, or other device on a LAN is said to be a node in the network. Each has a device (or software) for controlling communications with the other nodes on the network. Usually all the nodes in a LAN are within a few hundred feet of each other.

LANs make a great deal of sense for environments in which groups of people work together using a number of computer workstations. From an economic point of view, their advantage is that they make it possible for each computer user to have access to expensive peripheral devices, such as hard disks and laser printers, that would be too expensive to provide for each computer. From the perspective of collaborative efficiency, a LAN provides the easiest and fastest way for colleagues to exchange documents electronically. As we will see, the Macintosh already offers two approaches to local area networking.

HOW TO START COMMUNICATING ELECTRONICALLY

You already know how to communicate with other Macintoshes by exchanging document disks. If you work in a collaborative Apple 32 system environment or if you belong to a Macintosh users' group, you have probably already exchanged disks with other users. For communicating through direct serial connections or over telephone lines, you need to use an appropriate communications program, such as MacTÉP, a public-domain program, or MacTerminal, a multifeatured communications application from Apple.

To transmit information to or from another computer, you must first be connected to the other computer. There are three ways to do this. First, you can be physically connected to the second machine by a cable. Inside the cable are a number of wires. One of the wires is used by your Mac to send data to the other computer. Another wire receives data from that computer. Still other wires serve other purposes, such as signaling that the computers are connected to each other, or stating their readiness to send or receive data. Figure 9.1 shows a cabling arrangement for connecting a Mac's phone port to a modem or a second computer. If you aren't up to soldering your own cable, you can probably get one from your dealer.
One of the trials of communicating with other computers is that the so-called RS-232 “standard” for communications interfaces does not ensure that two RS-232 devices can work together with a simple cable. The full RS-232 standard provides for up to twenty-five wires inside a cable. Each of those wires is soldered to a small pin in a connector at the end of the cable. The pins fit into holes in the computer’s or modem’s matching connector. The problem is that most of the twenty-five pins can be used in several different ways (or not at all) without violating the “standard.” That means that different computer manufacturers make different choices about which of those pins will actually be used and how they will be used. This often makes it very difficult to connect machines directly. If you must frequently connect your Macintosh to different machines in order to transmit or receive text files, consider purchasing a SmartCable (IQ Technologies, Inc., 11811 N.E. First Street, Suite 308, Bellevue, WA 98005). This clever device automatically
How to Start Communicating Electronically

figures out which pin is sending data, which one is receiving data, and which pins are doing the "handshaking"—telling when the system is ready to send or ready to receive. It has 25-pin connectors at both ends, but you can use it with your Macintosh phone port by using your printer cable or the Mac modem cable to go from the Mac 9-pin connector to the SmartCable.

The second way to be physically connected to a second computer is to use a modem and the telephone system to link up. The telephone network connects your Mac's modem to the modem of another computer, whether that system is another microcomputer such as a Mac or an IBM PC, a minicomputer like the DEC VAX 11/750, or a large mainframe computer. Of course, the connection isn't entirely physical if the telephone link happens to make use of satellite transmission, but the principle is the same.

The third way to connect to another computer is to use the AppleLine protocol converter. This device allows you to connect your Mac to an IBM mainframe computer. With AppleLine and MacTerminal, your Mac can emulate an IBM 3278 terminal. The AppleLine peripheral device connects the IBM mainframe's communications controller to the Mac or to a modem that connects to the Mac. The AppleLine converts the signals that IBM computers use into the kinds of signals that most other computers use, including the Macintosh and Lisa systems.

There are several advantages to using a Mac in place of an IBM 3278 Model 2 terminal. One is that it takes up much less room on your desk. More important, the Mac can be used to do a great deal of work without constantly interacting with the mainframe system. Once you've used the Mac to get information from the IBM mainframe, you can manipulate it independently of the big computer, perhaps by editing the text with MacWrite or by manipulating the numbers in a Multiplan spreadsheet. When you've got the results you want, you can copy them to the Clipboard and then paste them in the MacTerminal window, sending them in a burst to the mainframe.

Many mainframe users are very happy to do as much of their work as possible on the Mac because its response time can be so much better than a big computer's. The mainframe must divide its processing time among all the users currently sharing the system. When many users give the computer many tasks at once, its response time can slow down tremendously. Sometimes it can take many seconds to respond to each key press. If you've already entered all the work on your Mac, you can quickly dump it to the big IBM. If response time is terribly slow at that moment, you can take a coffee break and let the Mac worry about sending each line to the mainframe when it is ready.
USING MACTERMINAL

The MacTerminal application provides software control that makes all three of these kinds of machine communication—direct serial connection, telephone links, and connection through the AppleLine protocol converter. MacTerminal is an application that lets you use your Mac as though it were a computer terminal. A computer terminal has a keyboard and a screen, like a personal computer, but it usually doesn’t have very much internal memory or any form of mass storage such as a disk drive system. Terminals let a user type information that is immediately sent to computers, and they display information that the computer sends back. MacTerminal lets you do more than type things to other computers, however. It also lets you send documents direct from Macintosh disks and receive whole documents and put them on disk.

Different terminals have different performance characteristics. For example, a Hazeltine 1520 terminal uses a set of digital codes to transmit and receive certain instructions which is different from that used by a Lear-Siegler ADM 3-A terminal. The standard emulation mode provided by the MacTerminal program makes your Mac look like a Digital Equipment Corporation (DEC) VT100 terminal, the terminal type in widest use today. Most large time-sharing computers, such as the ones you can access by telephone, can interact with a VT100.

MacTerminal controls are provided through its five pull-down menus (Figure 9.2). MacTerminal users should also remember to make appropriate use of the Apple menu’s Scrapbook and Note Pad options too. It can be very convenient to load the Scrapbook or Note Pad with the elements you want to transmit from MacTerminal.

In MacTerminal, you are presented with a window capable of displaying twenty-five lines of 80 characters of 9-point font (Figure 9.3). If you choose to activate the 132-character-line option, you can horizontally scroll, using a scroll bar at the bottom of the window. Small indicators with labels such as “On Line” give status information about the current state of the emulated terminal. “On Line” means that whatever you type will be transmitted to the connected computer. “Local” means that your typed input will not be sent out from the phone port. “Kbd Locked” means that nothing typed on the keyboard will be sensed by the emulated terminal; typed input will not appear
on the screen, nor will it be transmitted. The "L1"..."L4" indicators serve the same function as four lights on the DEC VT 100 terminal. You will probably not ordinarily need to be concerned with the special functions represented by these lights.

If you have a good connection to another computer, and if that computer has a program handling your input, you can simply type in the MacTerminal window, and all the typed input will be transmitted to the connected computer. But suppose you want to send a complete text document to an Apple II computer in the same room. You would first connect the two
computers' serial ports with an appropriate cable (one that connects the Send Data pin on the Mac to the Receive Data pin on the Apple II, and so on). Then you would start communications programs on the two computers. If you were using the Apple Pascal operating system on the Apple II, you might choose to use the PTerm program from Southwestern Data Systems (10761-E Woodside Avenue, Santee, CA 92071). Just as MacTerminal makes the Mac act like a computer terminal, so PTerm makes the Apple II running the Pascal operating system act like a terminal.

To begin the file transfer process, choose the Send File option from the File menu in MacTerminal. In Figure 9.4 a text document named "Apple2" has been selected from the available choices for transmission to the connected Apple II computer. The same method can be used to send Macintosh text files to other local microcomputers, such as an IBM PC or a Hewlett-Packard microcomputer, or any other machine that has a serial interface and terminal emulation software that supports the use of that interface for machine communications.
To make the file-sending operation work, the terminal emulation program in the other local computer must have an option to receive an incoming file for storage on disk. Most popular microcomputer terminal emulation programs, such as PTerm, do have such an option. This provides you with a straightforward way to transmit text documents from the Macintosh to other computers. These documents are the type you create in MacWrite when you use the Text Only button from the Save As window.

If the other computer is another Macintosh, you can even transmit documents of other types, in addition to text. To do so, you must first establish the appropriate file transfer settings (Figure 9.5). From the Settings menu, select the File Transfer option to bring up a dialog box with the file transfer setting options. Two different transfer methods are possible. The Modem7 method is one that uses a communications protocol developed several years ago for communication among an earlier class of computers, which all used the CP/M operating system. Because the Modem7 method became so widespread, it has since been adopted by the developers of software for other
operating systems as a standard file transfer protocol. This means that communications programs for the IBM PC and the Apple Macintosh were implemented with the Modem7 option.

Modem7 makes it possible to transmit not only text files but also other types of documents, such as MacPaint pictures or Filevision drawing files. You can even transfer applications, although it is likely that software developers' copy protection schemes will prevent the copy of an application from working properly on the new disk. Modem7 is also a useful method for transferring files, especially files that are not simply text files, with other computers that have software that uses the Modem7 transfer method. It performs a number of checks on the transmission data to ensure that it is error-free.

Computers for which Modem7 software is available are typically other microcomputers rather than larger, time-sharing minicomputers or main-
frames. When you communicate with such large systems, you should stick to the Text transfer method.

The option to specify a delay following every character or every line is most likely to be of use when you are communicating with the big time-sharing systems. These computers usually have to take in data from a number of separate sources all at about the same time. When the data is being sent to these machines at the speed of a normal human typist, they can often—but not always—keep up fairly well. Such a system has no way of knowing that your input is being sent by an application (MacTerminal) that can pull characters out of a file and send them out through the serial port a great deal faster than a human being could type them.

When you send a file to a system that can't keep up with the Mac's input rate, some of the characters may be lost. If you have trouble with lost characters when you send files to a "host" time-sharing system, first see whether specifying a small delay after each line solves the problem. The host may have a small buffer available for storing up to a line of text without losing any of the characters for that line. If a line delay doesn't solve the problem, add a delay between characters as well. If necessary, you can degrade MacTerminal's file-sending performance to that of an ordinary typist.

CUSTOMIZING MACTERMINAL FOR THE COMPUTERS YOU TALK TO

Other computer systems you want to use your Macintosh to communicate with may differ in the ways they interact. For example, one may have the Modem7 file transfer method available, and another may not. You can change MacTerminal's settings to match the requirements of any other computer you routinely use. Figure 9.6 shows four documents that are to be sent to different computers. These computers each require different settings of the communication options. The files labeled "Osbornel" and "Apple II" are for local communications with other microcomputers. As it happens, these two computers require that the Mac use different baud rates—a measure of the speed with which the computers communicate. To send the Osbornel document, I must set the speed to 1200 baud, since that is the maximum baud rate of that machine. To send the Apple II document, the baud rate can be as
high as 9600. To use a computer with different communications characteristics, one must customize certain of MacTerminal’s attributes so that it is compatible with the other computer.

To set communications options, such as the baud rate that MacTerminal will use, you use the Compatibility . . . command in the Settings menu. The large dialog box in Figure 9.7 offers a set of baud-rate options together with a number of other communications attributes, such as the number of bits used to form a character. It is unfortunate that all computers don’t do things the same way when it comes to sending and receiving serial information. Fortunately, MacTerminal gives you options for making the Mac conform to whatever methods the communicating computer uses.

In addition to customizing MacTerminal to describe the communication options of each microcomputer you communicate with, you can also modify MacTerminal’s compatibility for any remote time-sharing computer you use. You can use the Communications dialog box to choose an appropriate baud rate, number of bits per character, and so on. With the Terminal Settings option in the Settings menu, you can determine the characteristics of the
### Figure 9.7 The compatibility settings dialog box

**Compatibility Settings**

- **Baud Rate**
  - 50
  - 75
  - 110
  - 134.5
  - 150
  - 200
  - 300
  - 600
  - 1200
  - 1800
  - 2000
  - 2400
  - 3600
  - 4800
  - 9600
  - 19200

- **Bits per Character**
  - 7 Bits
  - 8 Bits

- **Parity**
  - Even
  - Odd
  - None

- **Handshake**
  - XOn/XOff
  - None

- **Connection**
  - Modem
  - Another Computer

- **Connection Port**
  - Modem
  - Another Computer

### Figure 9.8 The terminal settings dialog box

**Terminal Settings**

- **Terminal**
  - UT100
  - TTY
  - IBM 3278

- **Compatibility**
  - ANSI
  - VT52

- **Cursor Shape**
  - Underline
  - Block

- **Character Set**
  - United States
  - United Kingdom

- **Line Width**
  - 80 Columns
  - 132 Columns

- **Protocol Conv.**
  - ASCII Line
  - Husier CDB

- **On Line**
  - Local Echo
  - Status Lights

- **Auto Repeat**
  - Auto Wraparound
  - New Line

- **Repeat Ctrls.**
  - Transparent

**Budgets**

Ralph, you were any problem?
emulated terminal (Figure 9.8). The settable features include some that determine what kind of a terminal the Mac will look like to the other computer.

One of the options that the Phone menu offers is one labeled **Phone Settings**. This command opens a dialog box that lets the user enter a telephone number for the computer for which all the settings customize the MacTerminal document. You can enter not only a telephone number but also an area code and even a telephone access code, such as one from Sprint or MCI, if you are a subscriber to such a service.

If a telephone number has been entered into this box, you can then use the **Phone** menu to actually place the call. Simply pull the menu down to the **Dial** option. The Mac will immediately instruct the modem to place a call to that number. If you have an Apple modem, or one that uses the same signals to control special functions like automatic dialing, your fingers need never touch a telephone to place a computer-to-computer telephone call.

![Figure 9.9 The phone settings dialog box](image)
TWO LOCAL AREA NETWORKS

A number of different local area network implementations are open to Macintosh users. Each has special features that are likely to be an advantage in particular situations. Two simple and inexpensive local area network (LAN) systems are the Corvus Omninet and Apple Computer's AppleTalk system.

Corvus is a manufacturer of microcomputers and computer peripherals that got its start as a hard disk drive manufacturer. One of Corvus's earliest products was a network system that allowed several microcomputers to share the same large disk drive. That product evolved to the modern Omninet system, which has the largest installed base of any microcomputer network manufacturer as of this writing.

The heart of an Omninet system is a hard disk drive. Each hard disk drive made by Corvus incorporates file server hardware. These circuits, together with the appropriate software (also supplied by Corvus), make it possible for a number of different computers to use the drive as though it were connected directly to that computer alone. File server systems are so named because they serve files to the computer systems to which they are connected. As you can imagine, in file server systems there is a danger that one user's documents on the hard disk could be altered by another user. Corvus solves this problem by letting you divide the disk into a number of separate volumes.

A volume is an area or set of areas on the disk that can be treated by the Mac's software as though it were an independent disk drive, one with lesser capacity than the real hard disk. The Corvus software lets one user in the network set up the basic structure of the network system, deciding how many volumes the hard disk will be divided into and how large each one will be. (They need not all be the same size.) This special user, who is called the network manager, also tells the system who the authorized users are and to what volumes each user should have access.

One very simple means of ensuring that one user cannot damage another's documents is for the network manager to ensure that no volume can be accessed by more than one user. This is probably too restrictive a solution for most LANs, since one of the justifications for LAN technology is that it provides very convenient methods for colleagues to share documents that are being collaboratively developed. A better approach is to give each user at least one private volume, to which the other users do not have access, and also to give the user access to a number of shared volumes on the disk. If necessary,
certain users can be given only limited access to certain volumes. It is possible in the Corvus system to give a user the right to read material from a particular volume, but not to write to it. If the user wants to alter a document, he or she must send a copy to a volume for which the user has write access. Collaborators can actually do most of their work on private volumes. Then, when they want to turn over their most recent revision of a working paper, a spreadsheet, or a project chart, it can simply be transferred to a volume shared by a colleague.

A second means of communication provided by Omninet is electronic mail. Users can send documents directly to other users in the network, whether they are currently using the system or not. The recipients can then decide whether to keep the received documents in their private volume space or to put them in the trash after reading them. Other options include forwarding copies of the messages to still other users on the network.

Probably the greatest advantage of the Omninet system is that it permits computers developed by different manufacturers to share the same hard disk drive. When the computers share the same operating system (such as the widespread p-System from Softech Microsystems), they can share almost every kind of file on the disk. When the computers use different operating systems, they can still share text documents. With Corvus networking products, users of the IBM PC, the Macintosh, the Apple IIe, and other computers can all make collaborative use of the same Omnidrive hard disk drive.

In addition to file sharing and electronic mail, Omninet makes it possible for all the users of a network to share printers. Without the use of a LAN, the only way that different computer workstations can share a printer is by physically disconnecting and reconnecting cables whenever a new user wants to use the printer. With Omninet, a node in the network can be the network’s PRINT SERVER, a computer or a specialized hardware device that dumps text files to a printer connected to the print server. If several network users request that new files be printed while the print server is printing another user’s text, then the server will queue those requests and perform each printing job in turn. Check with your dealer to determine the current features of Omninet print serving for Macintoshes if this aspect of network support is especially important for your work.

Apple’s local area network, AppleTalk, lets members of the Apple32 family of computers share devices such as Apple’s file server and print server
systems. The file server features permit several Macintosh or Lisa-2 users to share the Apple hard disk drive. The print server software, used in conjunction with a file server, lets more than one computer use the Apple laser printer without going to the trouble of checking with other users and changing cables whenever you want to print from your system. A laser printer makes an especially good printing device for a LAN system. It is more expensive than many other printer technologies—an expense that pays off in high-quality output and enough speed to handle the printing requirements of many users without imposing an undue wait on any of them.

If you are in the position of having to select a LAN for your work environment, consider your needs in making the choice. If it is important that you connect computers from a number of manufacturers, consider the Corvus Omninet system. Omninet is a mature, well-debugged technology. On the other hand, add-in AppleTalk interfaces for computers such as the AppleIIe and the IBM PC can bring these systems into Apple's network.

If you want to get the most out of a network that does not include a variety of microcomputers but instead fully utilizes the Macintosh and Lisa technologies, then AppleTalk is probably the right system for you. Per-station costs are very low. To add a Macintosh to an existing AppleBus network, all you need to purchase is some cable and the inexpensive connection module, a box about the size of a mouse. This box has an 18-inch drop cable that connects it to the Mac's serial communications port—the one with the telephone icon. Installation is as easy as “installing” the cable that connects a Mac to an Imagewriter printer.

AppleTalk permits up to thirty-two devices to communicate directly, including computers, file servers, printers, modems, and other peripherals. The two most obvious services provided by AppleTalk are file sharing and electronic mail. Other networking applications, such as shared data base systems, are likely to come from third-party software developers.

AppleTalk was designed to permit access to other networks, including more expensive, high-speed networking systems. AppleTalk is a work area network—a Very Local Area Network. It will soon be possible to communicate with Backbone Networks, those that can communicate over greater distances and at higher speeds, including those supported by IBM, from AppleTalk. It's a good bet that Apple Computer will put its technological resources to work to make continuing contributions to the AppleTalk LAN system.
Local area networks and terminal emulation represent two quite different styles of machine communication. LANs presuppose very rapid transfers of information, while terminal emulation commonly uses speeds as slow as 30 characters per second (300 baud) and sometimes even less. As their name implies, LANs can be used only for computers that are quite close together, usually in the same building, while terminal emulation plays a role in linking computers that can be on opposite sides of the earth. LANs also require a significantly larger financial commitment, since hard disk drives are an essential part of most LAN implementations. Terminal emulation requires only a program such as MacTerminal and, if the communicating machines are not directly linked by a cable, a modem—some of which are quite inexpensive.

On the Macintosh, both types of communication system typically make use of the Mac's high-speed serial interface, the one that has a 9-pin connector with a telephone icon. Does this mean that you can't use both communication modes at the same time? Is it impossible to use a hard disk system and a modem together? Some hard disk systems, such as the Tecmar MacDrive, provide a 9-pin connector that functions as your remote serial port while the hard drive is hooked up. If you are using a hard disk that doesn't bring out the telephone connector on its cabinet, you may be able to deal with the problem in software. If your terminal emulation program supports the use of the printer port (as MacTerminal does), you can use that connector for your telephone communications while preserving the high-speed port for LAN usage. Of course, you won't be able to use a printer from that port while the modem is connected there, but with luck you will have access to a print server through the LAN.

Despite the differences in communicating with other computers through the two different methods, they have important features in common. Both can be used to send simple messages to users of other computers. More significant, both can be used to improve your productivity by letting you share work with colleagues at the Macintosh document (file) level, rather than just through the medium of printed output.
CHAPTER 10

MORE PICTURES, MORE WORDS

HELP WITH MACPAINT 214
CREATING GRAPHICS WITH A WORD PROCESSOR 217
INTRODUCING MACDRAW 218
HELP WITH MACWRITE 226
WORD PROCESSING WITH MULTIPLAN 229
ADVANCED WORD PROCESSING WITH MICROSOFT WORD 229
HELP WITH MACPAINT

A number of products have emerged to help MacPaint users in the development of graphic displays. Some of these are separate applications that can use MacPaint documents or that produce documents you can elaborate further with MacPaint.

There is one way that any other Macintosh application can be used to produce a MacPaint document. You can always create a MacPaint “snapshot” of the current screen simply by typing Command-Shift-3 with the Caps Lock key down. This puts a copy of the current screen display on the disk and labels it “Screen 0” (or “Screen 1” if there is already a “Screen 0,” and so on). You can later give these graphic images more meaningful names and, if you choose, edit them using MacPaint.

Products that can be used to produce MacPaint-type documents include digitizing tablets and video digitizers. A DIGITIZING TABLET is a device that can be used to trace drawings or to map points on a printed surface into a graphic representation of that surface. The mouse is a wonderfully easy-to-use input device for graphics work that is essentially freehand. If a tracing must be done, a true digitizing tablet provides a much more effective method for entering the elements to be traced than does the mouse. It is possible to treat the mouse as a pretend digitizer by taping a pointing element, such as a bent paperclip, to it. Then use the point of the paperclip as the tracing pen. The results will require a lot more editing than would be the case if a digitizing tablet were used.

A VIDEO DIGITIZER is a video camera with supporting interface electronics that lets a user make a video image of anything he or she points the lens at. Such systems usually take a number of seconds to store an image, so they are best used to make graphic images of still objects rather than moving ones. These graphic images can later be edited using MacPaint.

Koala MacVision
Koala Technologies Corporation
3100 Patrick Henry Drive
Santa Clara, CA 95050

Micron Eye Video Digitizer
Micron Technology, Inc.
2805 East Columbia Road
Boise, ID 83706
These digitizers cost only a few hundred dollars, and they let you capture anything you can photograph in a MacPaint document.

Other products simply provide ways to use MacPaint more easily or more effectively. One such class of ancillary products is clip art portfolios. A CLIP ART PORTFOLIO is a disk with a number of MacPaint documents with potentially useful drawings or patterns. Such portfolios include:

ClickArt
T/Maker Graphics
2115 Landings Drive
Mountain View, CA 94043

Mac the Knife
Miles Computing, Inc.
7136 Haskell Avenue, Suite 300
Van Nuys, CA 91406

McPic
Magnum Software
21115 Devonshire Street, Suite 337
Chatsworth, CA 91311
and

daVinci Graphics Packages
Hayden Software
600 Suffolk Street
Lowell, MA 01853

The images in such portfolios are created by professional artists. Many users will decide that the expense for such samplers is well invested, given the time it would take them to replicate the included artwork on their own. Figure 10.1 was created with MacPaint by combining a number of separate clip art images from ClickArt.
Figure 10.1 A MacPaint document composed with ClickArt, a graphics sampler
To use clip art, you simply copy symbols, patterns, pictures, or whatever into your own MacPaint documents from the clip art documents, using the usual Macintosh cut-and-paste techniques. A number of specialized clip art disks are available, each appropriate to the needs of different groups of users. The daVinci series from Hayden, for example, includes separate packages for landscape design, building design, and interior design. If you keep abreast of clip art developments, you may find that a portfolio that meets many of your own graphics needs is available. Check with your dealer and scan the advertisements in magazines like *MacWorld*.

A special case of clip art is PATTERN ART, files that consist of sets of patterns that you can copy into the pattern palette at the bottom of the MacPaint screen. One source for pattern art is *500 Menu Patterns*, from

**FingerTip Software**

3327 East 14th Street  
Long Beach, CA 90804

You can put any pattern you like into a pattern slot on the MacPaint pattern palette. The only requirement is that the pattern must exactly fill a square sixteen dots on a side. To paste a pattern from the graphics editing window into one of the pattern slots, first click the pattern you want to replace in the pattern palette. Then choose the **Edit Pattern** option in MacPaint’s Edit menu. (As an alternative, you can just double-click on the pattern box you want to replace with the new pattern.) Click the pattern area you like in the editing window. After you click the OK button, the new pattern will be installed in the pattern box you selected. If you save the MacPaint document, it will have that pattern choice available until you choose to replace it.

**CREATING GRAPHICS WITH A WORD PROCESSOR**

A kind of graphics construction is possible from within a word processor, such as MacWrite or Word. Fonts are not limited to containing letters of the alphabet as symbols. For example, Apple’s Cairo font can be used to create a REBUS, a combination of words and pictures that can be read as words or syllables.
To use a graphics font effectively, you have to know where the pictures are on the keyboard. One way to keep track of this is to construct a keyboard map. I made the keyboard map shown in Figure 10.2 using MacPaint, labeling each key both with its conventional alphanumeric character—in New York 9-point font—and with its 18-point Cairo character.

INTRODUCING MACDRAW

MacPaint is an application for creating pixel-oriented graphics. Every dot on the screen is under your control in MacPaint, and the stored MacPaint documents consist of patterns of dots. MacDraw is not a pixel-oriented graphics system. Instead of creating documents that record the placement of every dot in a drawing, MacDraw remembers its documents as collections of objects such as rectangles, arcs, lines, and text. Such objects appear on the screen as collections of pixels. MacDraw does not remember them as collections of pixels, however, but as a set of codes that specify the type of the object and such features as its height, width, and so on. When you move an object on the screen, MacDraw does not move the pattern of dots, but rather erases and redraws the object, using this underlying abstract representation.

Another object-oriented graphics system is Filevision from Telos, covered in Chapter 7. Filevision lets the user store data forms that include extensive information about the represented objects. MacDraw does not have this capability, but it does offer a more extensive repertoire of drawing tools than Filevision, and it lets you make larger drawings. If your drawing requirements do not include storing alphanumeric data about graphic objects, then you may prefer MacDraw to Filevision.

What advantages does MacDraw have over MacPaint? One advantage is that MacDraw documents are in some ways much easier to edit than MacPaint documents. In MacDraw, you can edit a stored document with overlapping objects and decide to spread them apart from each other in the drawing. Because MacDraw stores the scene as a collection of objects rather than as a set of pixels, if one object overlaps another, the obscured portions are not lost. You can ordinarily rearrange the elements of a drawing much more easily if it is done in MacDraw than if it is done in MacPaint.
Figure 10.2 The Cairo keyboard layout
An advantage of MacDraw over both MacPaint and Filevision is that it lets you draw much larger documents than either of those applications does. With the current version of Filevision, you are limited to drawings that fit in the Filevision drawing window. With MacPaint, you can create and edit images about three times the size of the screen—images as large as 8-by-10½ inches. MacDraw lets you work with much larger drawings. It provides both horizontal and vertical scroll bars that let you move over the entire surface of your drawing, from the top left to the bottom right portion of your drawing. Figures 10.3 and 10.4 show two parts of an organization chart. The drawing was completed simply by clicking on tools such as the Rounded Rectangle and dragging out the shapes on the display.

The set of tools available for creating images in MacDraw is more comprehensive than in Filevision. A glance at the menu options reveals that many more fill patterns are available than in Filevision. Other special features include control over line thickness and a special line type with arrowheads.

MacDraw documents can get very large indeed. The printed output from a single MacDraw drawing can cover up to sixty printed pages (Figure 10.5). When all the pages have been pasted together, you can have a composite
Figure 10.4 The bottom right portion of the Organization Chart drawing

Figure 10.5 The drawing size dialog box
MacDraw document to tack to your wall that is 8 feet wide and 4 feet high. That’s enough room for a detailed organization chart for a sizable company. If you prefer, you can have a large-scale landscaping or architectural plan, or any other structured drawing for which it makes sense to consider the pictured elements as separate objects (for example, Figure 10.6).

You might think that the object status of everything in a drawing would be a disadvantage when you want to move a group of drawn elements together. Suppose you wanted to reposition the box labeled “Clerk Level” in the organization chart in Figure 10.4. It would be awkward if you had to select each object and move it separately—two rounded rectangles (one filled with black, the other with white) and two text elements (in different fonts and styles). You don’t have to make four separate moves. You can bind these four objects together into a single object, using the Group option in the Arrange menu. First drag through the whole set of objects, beginning well above and to the left of all of them. All the objects will be simultaneously selected (Figure 10.7). In MacDraw, as in FileVision, you can tell that an object is selected when its small rectangular “handles” are present. At this point, if you invoke the Group option, the four sets of handles will be replaced by one set of handles that includes the entire group. You can now reposition the entire group as though it were a single object (Figure 10.8). Later, if you want to edit one of the components of the group, you can use Ungroup to separate the collection back into separately selectable components.

MacDraw provides many features to give you fine control over the placement of objects in a drawing. The Show Rulers option in the Layout menu will bring up both a horizontal ruler, along the top edge of the drawing window, and a vertical ruler, on the left side of the window (Figure 10.9). As you move the mouse pointer in the drawing window, tiny tic marks on the rulers follow your movements. By pointing the hot spot at a particular point on your drawing, you can read both its horizontal and vertical values on the rulers.

You can even use the Custom Rulers option to bring up a dialog box that lets you choose units of measure for your rulers (Figure 10.10). Your control over the placement of objects is further enhanced by MacDraw’s grid, similar to the FileVision grid. In the screen prints in Figures 10.3–10.5 and 10.7–10.9, the grid was visible. In addition to letting you turn the grid on and off, MacDraw gives you the capability to have the grid on but invisible (Figure 10.11).
One Bedroom

THE NOOK

Figure 10.6 Floorplan for "The Nook"
Figure 10.7 Several objects selected at the same time

Figure 10.8 A moved group
Figure 10.9 MacDraw's rulers

Figure 10.10 The custom rulers dialog box
Frequent users of MacPaint have often wished for some way to manipulate areas of the picture that are too large to fit in the MacPaint window. MacDraw gives you a way to edit drawings that are too large for the MacDraw window. More accurately, MacDraw lets you shrink your drawing to fit more in the window as in Figure 10.12. This makes it possible to select large chunks of drawing by dragging over them with the mouse and then copying, cutting, or moving the entire assemblage. Naturally, you can always use the Enlarge option to restore the drawing to its normal size once you have completed the operations that require working with the reduced versions. MacDraw's many capabilities are reflected in a large number of menu options (Figure 10.13).

HELP WITH MACWRITE

Just as software and hardware firms have produced aids to productive use of MacPaint, so, too, secondary products in support of MacWrite have emerged. One example of MacWrite add-on products is software that lets Macintosh
owners print with fully-formed-character printers, commonly called letter-quality printers. (Now that laser printers provide another way to get letter-quality printing from the Mac, the daisy wheel printers are not the only letter-quality game in town.) Fully-formed-character printers, although slower and less flexible than laser printers, are somewhat less expensive, and they produce the typewriter-like text that many business people feel is essential for their correspondence.

The Mac Daisywheel Connection is a product from Assimilation Process (151 Eastridge Drive, Los Gatos, CA 95030) that lets the Mac talk to a fully-formed-character printer. Assimilation provides both the necessary interface cable and a program that lets the Mac use daisy wheel printers. Other manufacturers also produce cables and software to support Macintosh letter-quality printing.

TTX, Inc., produces a 64K-byte printer buffer called the MACpac for use with the Macintosh. It can be used either with the Imagewriter printer or with TTX’s own letter-quality printer, the TTX 1014. Using a printer buffer is one way to get to use your Macintosh more of the time. Instead of waiting for the printer to finish before you can use the Mac, you can let the Mac send all the printing instructions quickly to the printer buffer, which then passes them on.
Figure 10.13 MacDraw's menus
to the printer at its own, much slower speed. Once the Mac has finished sending to the buffer, you can use it again, even though the printer may continue for another hour.

Other products that work with MacWrite include separate applications that use MacWrite files. MegaMerge (from MegaHaus, 5703 Oberlin Drive, San Diego, CA 92121) lets you use MacWrite to create a form letter, which MegaMerge can then merge with a mailing list you create to print hundreds of individualized letters in sequence. MegaMerge can also be used to print out a series of MacWrite documents as though they form a single long document. This is a useful feature for MacWrite users who must produce long documents.

WORD PROCESSING WITH MULTIPLAN

You can use Multiplan to create tables of purely textual materials. In many cases, you will find that it is easier to format a table of text elements using Multiplan than using MacWrite. In MacWrite it is all too easy to backspace past the tab position when you delete an element at a tab, making your table reformat itself in an unexpected fashion. In Multiplan, you can treat the columns as tab stops, making it easy to place tabular data wherever you want. If it turns out that your table has some numeric elements that are related to each other, as in Figure 10.14, you can set up the formulas to compute whatever portions are computable. Whether or not your table includes formulas, you will find that it is easy to produce simple text tables using Multiplan (Figure 10.15).

ADVANCED WORD PROCESSING WITH MICROSOFT WORD

Microsoft Word is a powerful word processing program that offers many capabilities that MacWrite does not have. Before considering some of these features, it makes sense to consider the advantages MacWrite has over Word. First, MacWrite is easier to learn. In part because it has fewer capabilities, you will need significantly less time to fully master MacWrite than to master Word.
Second, MacWrite is faster than Word at doing a number of word processing tasks.

There are a number of reasons for the speed difference. Some of them have to do with the way the two applications were implemented. MacWrite’s application code is all in the form of native machine instructions for the Mac’s 68000 processor, but Word’s code has two levels—a set of “p-code” instructions that are more compact than the corresponding 68000 code would be, and an interpreter to convert those instructions into real 68000 code as required. This approach was dictated by Microsoft’s desire to incorporate significantly more functionality into Word than could fit in an application program designed for a 128K Mac.

Another reason for the speed difference is that Word’s program code must be divided up into a number of semi-independent segments, chunks of program that won’t all fit in the Mac’s memory at the same time. When you ask Word to perform a function that uses a segment not currently in RAM, you have to wait briefly while that segment is loaded from the disk with the Word application on it. Owners of 512K Macs will find that Word runs a bit quicker
on their machines, since each code segment will have to be loaded only once from the disk. Thereafter, when you use a Word option that requires a new segment, Word will find that the segment is already in memory.

Another delay that Word sometimes imposes on the user is that of waiting for new portions of the text to be loaded from RAM. Like the newer, disk-based version of MacWrite, Word keeps only a portion of its documents in memory. When you scroll to a different portion of the document, you experience a delay while the new section of the document is read from the disk.

Any delays due to disk-accessing time can be greatly reduced if you are working with a hard disk drive (see Chapter 12). Some users who find Word's delays annoying when working from micro-floppies say that Word is a delight when used with a hard disk system, such as Tecmar's MacDrive or the Corvus Omnidrive.

A simple glance at Word's menus (Figure 10.16) does not reveal this application's extra power and flexibility. Many of Word's features are accessible only through dialog boxes that are brought up when certain of these options—the ones with the trailing "..."—are chosen.

Word's many enhancements and extra features fall into two general categories. Some features are ones that almost everyone will agree are very nice to have but would not in themselves justify a switch to a new word processor. An example of such a feature includes the Glossary option in the Edit menu. The Word glossary is a list of correspondences between very simple abbreviations and longer phrases. Rather than retype a frequently used long phrase every time it must appear in the text, the author can simply type its abbreviation, followed by Command-Backspace. When Word sees the Command-Backspace keypress, it finds the translation for the abbreviation in

<table>
<thead>
<tr>
<th>Font Size (Points)</th>
<th>Characters/Line</th>
<th>Lines/Window</th>
<th>Characters/Window</th>
</tr>
</thead>
<tbody>
<tr>
<td>9</td>
<td>82</td>
<td>25</td>
<td>2050</td>
</tr>
<tr>
<td>12</td>
<td>70</td>
<td>17</td>
<td>1190</td>
</tr>
<tr>
<td>18</td>
<td>41</td>
<td>13</td>
<td>533</td>
</tr>
<tr>
<td>24</td>
<td>35</td>
<td>9</td>
<td>315</td>
</tr>
</tbody>
</table>

*Figure 10.15 The printed Multiplan text table*
Figure 10.16 The Word menus
the glossary and inserts that phrase in place of the abbreviation. In the glossary associated with the Word document being edited in Fig. 10.17, the phrase “Object-Oriented Graphics Construction Systems” will appear when the abbreviation “OCS” is followed by the Command-Backspace keypress.

Adding new elements to the Word glossary is an easy matter. You first copy a phrase to the Clipboard from your document. Then choose Show Glossary from the Edit menu, type in the name of the new item, and type Command-V to paste the Clipboard contents into the glossary.

Another Word feature is the set of enhanced selection methods it provides. If you move the pointer just to the left of the left margin, it suddenly changes orientation, pointing to the upper right instead of to the upper left. When you now click, the entire line of text at that point is selected (Figure 10.18). If you double-click, the whole paragraph is selected. A third new type of selection available is the selection of whole sentences without dragging through the entire sentence. To select a sentence, position the text pointer anywhere in the sentence and then click while holding the Command key down.

![Figure 10.17 The glossary window](image-url)
The computer industry faces a crisis in software productivity. Advances in hardware and in operating system software increase computer usability, accessibility, and the performance/cost ratio, enabling more powerful systems to be made available to larger numbers of users. The problem is, what will those users do with these powerful computer systems? Recent experience has shown us that users will not program their computers.

In the early stages of the personal computer phenomenon, most computer owners were perforce programmers. Since there were virtually no application programs available for PCs, users had to write their own programs or do nothing with the machines.

The programs were small and primitive, and real computer programmers scornfully referred to them as toys when they thought about them at all.

Then the hardware capabilities of personal computers grew.

Figure 10.18 Selecting a line

Every word processing user can appreciate these kinds of improvements, but Word offers many more significant enhancements—including features that many will agree fully justify the additional effort of learning a new and more complex application.

Long Documents

With Word, the maximum size of your documents is determined not by the capacity of memory but by the capacity of disk. This advantage is most easily realized by users with external disk drives, including hard disks, but even single-drive Mac owners can create long documents with Word.

Wide Documents

MacWrite limits you to documents that have up to 7½ inches of text per line. With Word, you can create wider text documents. A horizontal scroll bar at the bottom of the text window lets you look at either end of a long line on your screen.
Paragraph Formatting

Word makes it easy to format separate paragraphs differently, without inserting new rulers wherever you want to make a format change. You simply select the paragraph to be reformatted and choose the option you want from the Paragraph menu (Figure 10.19).

Automatic Footnotes

Word lets you enter footnotes as you write or edit a text. You simply select the point at which you want to enter a new footnote and then choose Footnote from the document menu. A dialog box gives you the option of numbering the footnote yourself or letting Word do so automatically. The footnote entry area is opened as a separate pane at the bottom of the text window—much like the Multiplan panes discussed in Chapter 4 (Figure 10.20). You can choose whether to have the footnotes appear on the same page in which they are referenced or at the end of the document.
Working with Two Parts of a Document at Once

Word's ability to display two portions of the same document is not limited to footnotes. Just as in the Multiplan, you can in Word use the split bar to create separate panes in an editing window.

Editing More Than One Document at Once

Word will let you have several open text documents on the screen at one time (Figure 10.21). At any moment only one window can be active—ready to receive text or scroll to a new location. To activate another, just click on it. It is an easy matter to compare part of one text document with another in the midst of an editing session. As you can imagine, it is also easy to cut or copy from one document and paste into another.
Print Merging

Finally, Word gives you mail-merging capabilities that MacWrite does not provide directly (although MegaMerge can be used to provide similar features with MacWrite). Print merging can be used to automatically generate individualized letters by merging list data with a form letter. It can also be used to chain several documents together with sequential page numbering.

Whether Word's extra features are worth the extra learning time and extra expense to acquire is a matter for your own judgment. If your text efforts are limited to short documents such as letters, notes, and memos, you will probably decide to stick with MacWrite. If you spend a lot of time writing or if you prepare long documents, however, Word merits serious consideration.
CHAPTER 11

PROGRAMMING WITH MACPASCAL

INTRODUCING MACINTOSH PASCAL  239
STRUCTURED PROGRAMMING AND THE REALITIES OF PERSONAL PROGRAMMING  240
HOW MACPASCAL FILLS APPARENTLY COMPETING REQUIREMENTS  243
USING MACPASCAL TO PROGRAM KINETIC LINES  245
Many different programming languages are available for the Apple Macintosh computer. You can program in Microsoft BASIC, MacBASIC, FORTRAN, MacFORTH, Modula-2, LISP, Logo, C, Assembly Language, or UCSD Pascal. One of the programming languages best suited to the Macintosh's capabilities is Macintosh Pascal.

INTRODUCING MACINTOSH PASCAL

Pascal is an excellent language to learn because its design reflects many modern concepts about programming. Until Macintosh Pascal, however, it was very hard for people who were not already experienced programmers to teach themselves how to program in Pascal. For beginners, Pascal was harder to deal with than interactive languages like BASIC. Unlike most other Pascal implementations, Macintosh Pascal is interactive. This makes it a very good implementation for Mac owners who want to teach themselves how to program in Pascal.

Macintosh Pascal—often called MacPascal by its friends—is a full implementation of American National Standard Pascal (ANS Pascal). A number of extensions that take advantage of special Macintosh features are included. MacPascal is a programming environment. Like Multiplan or Filevision, Macintosh Pascal is a Macintosh application—quite an elaborate application for the development of documents that are not text files or spreadsheets or drawing files, but Pascal programs. As a full Macintosh application, it offers a special set of programmer's tools that make Pascal programming a lot easier and more productive than with conventional microcomputer Pascals.

Macintosh Pascal is not, at present, a full Macintosh development language. That means you can't write programs that make the MacPascal environment disappear, letting you create your own menu bar, your own windows, and so on. It would probably be a mistake to plan to write a major application in Macintosh Pascal. Complex applications are better developed on a Lisa-2/10 computer, using the Lisa Pascal Workshop.

MacPascal is very useful as a tool for the self-directed study of Pascal programming, but this is not its only use. Many significant programming tasks
can be accomplished with Macintosh Pascal. It just isn’t the right language in which to develop large-scale applications that must take over the whole computer.

STRUCTURED PROGRAMMING AND THE REALITIES OF PERSONAL PROGRAMMING

Different programming languages have different design goals. BASIC was designed to make it easy for people with little or no mathematical training to begin programming quickly. LISP was built to develop programs that manipulate complex data structures. The design of the Pascal programming language is quite simple, and its structures help ensure the reliability of its programs. These attributes make it both a good language for learning modern programming concepts and a good language for very large projects.

Pascal is a good language to learn because of its simplicity. In ANS Pascal, there is a limited vocabulary of control structures, standard data types, and standard procedures to learn. It is a good language for writing large programs because it promotes structured programming methods. Structured programming makes programs much easier to understand and therefore much easier to fix or modify, even long after they were first written.

In structured programming, parts of a program responsible for different actions are separated into partially independent modules (Figure 11.1). Modules can contain other modules, but strict rules of module accessibility prevent one part of a program from interfering with data that is supposed to belong to a different module. Structured programming helps prevent unexpected interactions between unrelated parts of a program. This makes programs more reliable; there are fewer unexpected behaviors in highly structured programs than in less structured ones. Structured programs are also usually much more readable, and that helps make them a lot easier to correct or add new features to.

The basic laws of modern programming are three:

1. Plan, plan, plan.
2. Design completely, in advance of programming.
3. Code in a top-down, structured fashion.
The term **TOP DOWN** means that you should first think about how a program will work globally (at the top level), then how each of the components of the global design will work, then how each of those components will work, and so on. In terms of Figure 11.1, you first think about the outermost module, the main program, then you think about the next highest level, the initialization section and the main body. You begin by planning the outermost modules and work down to the innermost ones.

When you design a program in this fashion, you eventually get to a level where all the components are terms already defined in Pascal. You have then completely designed a program, starting at the top, a very general statement.
of the program's function, and working down to the bottom, the sequence of Pascal statements that make up a program.

A form of program development much less favored by programming experts is called BOTTOM-UP programming. When you program in a bottom-up fashion, you first define all the smallest pieces of a program, then string them together to make a complete program. Experience has convinced most experts that this is an unreliable way to program. With top-down design methods, a programmer is much more likely to write simple, logical coding that does the job in a straightforward and logical way.

Top-down coding doesn't always appeal very much to inexperienced programmers. Part of the problem is that novice programmers tend to be a bit less certain of the effects of the programming language statements they use. They want to test the effects of each programming language statement or small group of statements on the machine. For a novice or intermediate programmer, it is frustrating to have to write an entire program before being able to see how any part of it works.

Even professional programmers find it useful to build major program modules separately from each other and to test them independently. This can be done by writing simple test programs that use these modules in a way that checks how they will perform in the larger program. This approach is called INCREMENTAL PROGRAMMING, and when used in conjunction with top-down design, it can help to assure rapid development of correct code. The method is usually not as practical for the less-experienced programmer, for whom a new module may be only a single statement. A beginner would like to simply type in a statement and see what it does.

Programming languages are a medium of communication between human programmers and machines. The native language of a central processing unit (such as the Motorola 68000 processor chip that is the brains of the Macintosh) doesn't look anything like Pascal or any other high-level programming language. It is closest to Assembly language, but Assembly is a difficult way for most people to program. It forces thinking about problems at a very detailed level from the start, distracting you from the structure of the problem itself. It is also much further removed from natural languages (like English) than a high-level language like Pascal.

Because a computer cannot understand programming language statements directly, they must somehow be translated into the machine's native language—its primitive instructions—before such statements can be carried
out by the computer. This translation process is performed by the computer itself under the control of a translation program.

There are two major types of translation programs that convert high-level language statements into machine instructions. They are called interpreters and compilers. INTERPRETERS translate language statements into computer actions one line at a time as the program runs. A COMPILER translates all the statements of a program after it is typed in, in advance of the program's being run.

Compilers and interpreters have different advantages and deficiencies. Compiled programs usually run more quickly than interpreted languages on the same machine. But interpreted languages are usually easier to develop in, particularly for less-experienced programmers. With an interpreted language, it is possible for a programmer to type in a single line of code and ask to have it translated and executed on the spot. Interpreters typically provide much more highly interactive programming environments than do compilers.

Interactive languages are better designed to serve programmers with a bottom-up programming style. This is probably one of the reasons that the BASIC and Logo languages have been very popular on personal computers. Both are typically implemented as interpreters and can therefore support the bottom-up methods that appeal to novice programmers.

Most Pascal implementations are compilers. Pascal meets the needs of experienced programmers, who want to use top-down design, but most implementations provide unfriendly environments for novice programmers. Macintosh Pascal provides a highly interactive environment for program development. It can therefore satisfy the needs of users with different programming styles.

HOW MACPASCAL FILLS APPARENTLY COMPETING REQUIREMENTS

The conflict between the advantages of top-down methods, which have been shown to promote program correctness and reliability, and bottom-up methods, which seem so well suited to non-professional programming, is largely resolved in Macintosh Pascal. Here is a language in which expert
programmers can feel comfortable but which novices can also use to learn the programming art. Macintosh Pascal is an interpreted Pascal, and one that provides a number of special features for interactive programming.

These features—including instant execution of one-line statements, stepping through a program a line at a time, examining stored data values while a program is running, and others—are provided through special windows and menu options. Macintosh Pascal provides up to six windows to give the programmer many different views of the program. When you first open the Macintosh Pascal icon, you are presented with three windows (Figure 11.2). The program text window is on the left and is labeled with the name of the program under development—in Figure 11.2 it is called “Untitled.” The other two windows are output windows for the program, one for graphics, labeled “Drawing,” and one for text, labeled “Text.” There are ways to make text appear in the Drawing window, but that is another matter.

```
program Untitled;
var
   (Your variable declarations)
begin
   (Your program statements)
end
```

Figure 11.2 The three standard windows
I wanted to write a program like those I'd seen on other computers that produce a continuously changing pattern of line segments. The basic method for the program is to paint a number of lines on the screen, each a bit different from the one next to it. Before drawing a new line, the oldest line of the set should be erased. If the computer can produce new lines fast enough, the whole set gives the appearance of a moving, organic system. Figure 11.3 shows three "snapshots" of a kinetic lines drawing, taken about two seconds apart.

To begin with, I used the standard text-editing techniques to change the Pascal program code in the "Untitled" window. I replaced the program name "Untitled" in the first line with "KineticLines." In Pascal, identifiers—names for things—cannot have spaces in them. For the body of the program, I set up a repeat . . . until structure (Figure 11.4). In Pascal, all the statements between the words repeat and until are done over and over again, in sequence, until the condition described after the word until becomes true.

The reserved words of Pascal, key words such as program, begin, end, repeat, if, and then, appear in Chicago font, and the other words appear in Geneva. Macintosh Pascal does this automatically as it analyzes each line when it is typed. Reserved words are so called because their use is reserved for special purposes in Pascal. You can't create a new variable or procedure and name it "program" or "if." The automatic font changes seem to help make the meaning of a program clearer to the reader, and even to its author. If you accidentally type in "repeat" when you mean "repeat," you become aware of your mistake very soon because the system doesn't put the word into the font that indicates it is a reserved word.

In Figure 11.4, the begin and end are followed by the comment "{KineticLines}.") A comment is anything enclosed between a set of curly brackets, { }. Comments have no effect on the way a program runs, but they help the programmer remember what the program is supposed to do.

The main body of the KineticLines program is between the begin-end pair. When the program is run, this sequence of commands is what will happen. To make it work, the system must be told the meanings of each of the undefined procedures such as EraseOldLine and DrawNextLine.

In my first, top-level version of KineticLines, I assumed that the program would display some fixed number of straight lines. At any moment the program should erase the oldest line. How it does this would be fully specified.
Figure 11.3 A sketch of a sample kinetic output

in the procedure EraseOldLine. Then it should draw the next line (in the manner prescribed by DrawNextLine) and then go on to the next line number, in procedure SetNextLineNum. The process should repeat indefinitely or until the user wants to quit—when UserWantsOut becomes true.

For now, I decided that I probably didn’t want to worry about when UserWantsOut should become true. Since a MacPascal user can always stop a
running program by pulling down the Pause menu to Halt, it isn't really necessary to put a method for user escape into a program like KineticLines. So I made UserWantsOut a constant whose value is always False.

A constant is a name that always represents the same value. Nothing in a program can change the value of a constant. Some constants are numbers, others have a character value like "A," and others have the value True or False. Some programmers use strange names like HellFreezesOver for the constant False. This makes the meaning of the program structure:

\[
\text{repeat} \\
\quad \ldots \\
\text{until HellFreezesOver}
\]

intuitively clear.

![Figure 11.4 Global structure for KineticLines](image)
The few lines of code, mostly empty procedure names, that KineticLines consists of at this point constitute a top-level design for the program. By the generally accepted rules of structured program development, the next step would be to write the declarations of the empty procedures. Instead, I wanted to make sure I knew how to draw a line in Macintosh Pascal. In effect, I wanted to suspend top-down design and do a little bottom-up coding.

From the Windows menu, I chose Instant to open a window that lets users enter single statements for immediate execution (Figure 11.5). I guessed that there might be a MacPascal procedure named DrawLine. I imagined that DrawLine would draw a line from the point specified by its first two parameters (5 and 5 in Figure 11.6) to the point determined by the second two parameters (20 and 100 in Figure 11.6). After I clicked the Do It button in the Instant window, a line appeared in the Drawing window. The top left corner of the Drawing window is 0, 0. I had thought that the order of parameters in DrawLine would be “top, left, bottom, right” \( (y_1, x_1, y_2, x_2) \), just as it is in

![Figure 11.5 Pulling down the Instant menu](image-url)
Macintosh Pascal graphic procedures, such as FrameRect and FillOval. Instead, the order is "left, top, right, bottom," or "\(x_1, y_1, x_2, y_2\)."

Other Pascal implementations do not have the Instant window feature. This option makes it possible to do a little bottom-up coding just to make sure you are on the right track. In this case it helped me to understand better how the lower-level code I planned to use (the DrawLine procedure) really worked. Without the Instant option, I would have either had to stop my work on the KineticLines program and write a small complete program to test the workings of DrawLine, or not find out my misconception until I tested a larger, finished program, one in which it would probably be harder to figure out the exact cause of an unexpected effect.

Knowing how DrawLine works, I felt I was ready to flesh out KineticLines just a bit and run it. The first running version would be a far cry from the final version I wanted to create, but it would be reassuring to have something working at an early stage of program development. The process of producing a series of programs, each of which is closer to your final goal, is called
STEPWISE REFINEMENT. In a step-by-step fashion, you modify and elaborate a program that is initially simple to get to the final result you want.

In the stepwise refinement technique, you are creating a sequence of programs where each is not very different from the one before. This means that when something doesn’t work as you expected it to, the possible sources of the problem are limited. You have a similar version of the program that did work as you expected, so you can focus on the differences between the two versions as the probable source of the error.

MacPascal encourages stepwise refinement more than most Pascal systems do. In many Pascals, the process of compiling a version of the program is so time-consuming that users are tempted to make great leaps from one version to the next, just so they won’t have to spend so much time waiting for compilations to end. MacPascal is virtually an “instant Pascal.” Once you have entered a few new lines, you can quickly see the behavior of the new version of the program by choosing one of the options from the Run menu.

To put the stepwise refinement technique into practice, you have to begin

![Figure 11.7 The undefined procedure bug](image-url)
by writing very simple versions of programs—ones that don't really attempt to produce the desired final result. Instead of first creating a program that really draws a kinetic line pattern, I wrote one that repeatedly draws and erases the same line on the screen. Even though it was very simple—only a few lines long—it turned out to have an error. When I tried to Run it, I got a display that showed the program had a bug. My program referred to something I hadn't yet defined, the procedure SetNextLineNum (Figure 11.7).

The little dialog box with the bug icon in it told me that the procedure didn't exist, and a line pointer in the form of a thumbs-down hand showed which line the dialog box was referring to. Clicking in the dialog box made it go away. I found a temporary solution—I changed the procedure call into a comment by enclosing it in curly brackets, since I didn't want to bother defining the procedure yet. When I ran this slightly altered version of the program, it worked. A flashing line appeared in the Drawing window (Figure 11.8). It flashed because it was continuously being drawn and erased by the repeated calls to DrawNextLine and EraseOldLine.
program KineticLines;
const
UserWantsOut = False;
MaxNumLines = 205;
كا रollbar = 16;
type
Coord = (x1, y1, x2, y2);
var
LineNum, NumSize, NumLines : integer;
Kln : array[0..MaxNumLines, Coord] of integer;
Max, StepSize : array[coord] of integer;
DrawW : Rect;

procedure InitKinetic;
var
  c : Coord;
  LL : integer;
begin
  GetDrawingRect(DrawW);
  Max[x1] := (DrawW.right - DrawW.left - Scrollbar);
  Max[y1] := (DrawW.bottom - DrawW.top - Scrollbar);
  Max[x2] := Max[x1];
  Max[y2] := Max[y1];
  NumLines := Max[y1] div 2;
  for c := x1 to y2 do
    Kln[0, c] := Random mod Max[c];
  for LL := 1 to NumLines do
    for c := x1 to y2 do
      Kln[LL, c] := 0;
  LineNum := 1;
  NumSize := 0;
end;

procedure SetNextCoords;
var
  c : Coord;
begin
  if NumSize <= 0 then
    begin
      for c := x1 to y2 do
        StepSize[c] := Random mod 13 - 6;
      NumSize := 5 * 1 + Random mod 12
    end;

Figure 11.9 A complete version of KineticLines
for c := x1 to y2 do
begin
  KLn[LineNum, c] := KLn[LineNum - 1, c] + StepSize[c];
  if (KLn[LineNum, c] < 0) or (KLn[LineNum, c] > Max[c]) then
    StepSize[c] := -1 * StepSize[c]
end;
end; {SetNextCoords}

procedure EraseOldLine (L : integer);
begin
  PenMode(notPatCopy);
  DrawLine(KLn[L, x1], KLn[L, y1], KLn[L, x2], KLn[L, y2])
end; {EraseOldLine}

procedure DrawNextLine (L : integer);
begin
  PenMode(patCopy);
  DrawLine(KLn[L, x1], KLn[L, y1], KLn[L, x2], KLn[L, y2])
end; {DrawNextLine}

procedure SetNextLineNum;
var
  c : Coord;
begin
  LineNum := succ(LineNum);
  NumSize := pred(LineNum);
  if LineNum > NumLines then
    begin
      for c := x1 to y2 do
        KLn[0, c] := KLn[NumLines, c];
      LineNum := 1;
    end
  end
end; {SetNextLineNum}

begin {KineticLines}
  InitKinetic;
repeat
  EraseOldLine(LineNum);
  SetNextCoords;
  DrawNextLine(LineNum);
  SetNextLineNum
until UserWantsOut
end; {KineticLines}

Figure 11.9 (continued)
The fact that I could get a very simplified version of the program to work helped convince me that my top-level design for the program was a reasonable one. Now I had to flesh out that design in a series of stepwise refinements, adding only a few lines to a procedure and introducing only one or two new variables to each revision of the program. At each stage, I could test the results of my revisions without waiting for a lengthy compile process to complete.

If you have Macintosh Pascal, you might like to type in the version of the KineticLines program given in Figure 11.9. It came out of a series of revisions of the original design, all based on the successive refinement principle. An interesting characteristic of the final version of the program is that it works correctly with Drawing windows of different sizes.

A Macintosh Pascal user can use the title bar and size box of the displayed windows to reposition and resize them. The version of KineticLines in Figure 11.9 checks to see how large the Drawing window is and uses that information to decide how many lines to draw and to avoid drawing lines outside the visible area.

As you explore programming with Macintosh Pascal, stay alert to opportunities to modify portions of your old programs for new uses. Building on your previous work is an important part of productive programming in MacPascal or any other programming language.
CHAPTER 12

THE MAC GOES ON

THE STANDARD MACINTOSH USER INTERFACE 256
MORE PROGRAMMING LANGUAGES 259
UPGRADING YOUR MAC 263
MACINTOSH ACCESSORIES AND PARAPHERNALIA 269
FUTURE MACINTOSHES 274
All the major applications covered in this book have made consistent use of the standard Macintosh user interface. The standard interface makes user-entered data available in windows. It provides pull-down menus for the selection of options in an essentially modeless environment—one in which most of the options are available most of the time. When an application must limit the user to a small set of choices, a dialog box provides those choices. In the standard interface, the most frequently used menu options have Command key equivalents. The data that the user edits can be cut or copied to the Clipboard and later pasted into the same or a different document. Finally, the Apple menu gives the user access to the Desktop Accessories. This interapplication consistency makes it very easy for an experienced Macintosh user to learn how to use new applications.

The major applications exhibit even greater standardization than this. For example, they provide a number of similar menus with similar choices. A File menu provides options for opening, closing, and saving data, for printing, and for quitting the application. An Edit menu provides the Cut, Copy, Paste, and Show Clipboard functions.

The reasons for the standardization are twofold. First, Apple used a carrot rather than a stick to encourage applications developers to employ a consistent user interface. Instead of trying to artificially impose standards, they gave developers the sophisticated user interface methods in a way that cost the developers virtually no machine resources. Apple developed all the program code required to present menus and detect menu choices, to use dialog boxes, to make Command keys work, and to make it possible to use the Clipboard. The Mac’s designers put almost all that code into ROM, memory that could not be used for anything else and that was always present. It was all there for free, and it didn’t require that the developers devote any of their precious RAM space to provide such functions. Not having to write their own user interface routines not only saved time but also meant that more of the Mac’s RAM could be devoted to program code and to storing the users’ data.

The second reason for the consistent user interface is that developers recognized that users appreciate consistency. The more nearly standard an application’s ways of interacting with the user, the less the user will have to learn anew after purchasing a new application. An experienced Mac user who wants to quit or to print or to close a document always pulls down the File
menu first to look for those options, because that’s where he or she has found them in the past. Developers have, on the whole, been quite consistent about the use of menus in order to help users cash in on their expectations.

Mac user interface standards will sometimes be violated. It is possible for developers to depart radically from the standard Mac modes of interaction, and some may find compelling reasons to do so. Rather than finding applications that completely abandon the Mac philosophy—although it certainly is possible to do so if a developer tries hard enough—you are more likely to find some applications that interpret it in unexpected ways.

An example of a somewhat unorthodox approach to Mac software can be observed in DB Master, a personal data base system from Stoneware. Like most other data base systems, DB Master lets the user define a format for storing data of a particular type of data. The user can then store information in records of that format, sort and retrieve the information, print it, and so on.

One unusual aspect of the DB Master implementation is that it makes use of two applications, one for defining the form of the records in a data base, the other for entering and manipulating information in that format. The first application, called Create File, is somewhat unorthodox in its menus. It provides four of them, none labeled "Edit." These are the menus shown above the line in Figure 12.1.

The second application, called Use File, is the one that lets you enter, edit, recall, and print information, using the forms you defined in Create File. The use of two separate applications makes more memory available by freeing RAM that one application requires to use another. The structure of the Use application is unusual in its use of special screens and dialog boxes.

When Use File is first opened, it presents a dialog box that asks you to choose the data base you want to work with. You are given this choice even if you open a particular DB Master document (data file) by double-clicking on it. In Use File there are again four menus, which are different from those of Create File. They are presented at the bottom of Figure 12.1. Unlike many other Macintosh applications, DB Master often takes away the background screen display when it brings up a new dialog box. In Figure 12.2 the display presented when you select Design New ... is shown. The data display that was on the screen just before has been removed.

Applications that use the standard Mac menu style have several benefits. They can help the user in the first few minutes of learning a new application, and, if you are an infrequent user of certain applications, you will probably find it easier to return to them after an absence if they are closer to the
<table>
<thead>
<tr>
<th>File</th>
<th>Field Edit</th>
<th>Print Style</th>
<th>Field Options</th>
</tr>
</thead>
<tbody>
<tr>
<td>Save...</td>
<td>Cut Field</td>
<td>Set Label Style...</td>
<td>Set Field Type...</td>
</tr>
<tr>
<td>Redesign...</td>
<td>Copy Field</td>
<td>Set Data Style...</td>
<td>Set Field Length...</td>
</tr>
<tr>
<td>Quit...</td>
<td>Paste Field</td>
<td></td>
<td>Enter Formula...</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Change Key...</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>File</th>
<th>Edit</th>
<th>Report</th>
</tr>
</thead>
<tbody>
<tr>
<td>Close...</td>
<td>Undo %Z</td>
<td>Design New...</td>
</tr>
<tr>
<td>Copy...</td>
<td>Cut %H</td>
<td>Change Old...</td>
</tr>
<tr>
<td>Merge...</td>
<td>Copy %C</td>
<td>Delete...</td>
</tr>
<tr>
<td>View File Design</td>
<td>Paste %U</td>
<td>Print...</td>
</tr>
<tr>
<td>Quit</td>
<td>Erase %E</td>
<td>Save This Format</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Record</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Add New Records</td>
<td>%A</td>
</tr>
<tr>
<td>Set Field Defaults</td>
<td></td>
</tr>
<tr>
<td>Select All Records</td>
<td>%F</td>
</tr>
<tr>
<td>Set Selection Criteria</td>
<td>%F</td>
</tr>
<tr>
<td>Find Next Record</td>
<td>%N</td>
</tr>
<tr>
<td>Delete Record...</td>
<td>%D</td>
</tr>
<tr>
<td>View Computed Results</td>
<td>%E</td>
</tr>
</tbody>
</table>

Figure 12.1 The major menus of DB Master
standard. This doesn’t mean you should ignore any application whose user interface diverges in any way from the standard. You should examine the application for yourself; you may find you like its style even better. I have to admit that DB Master was surprisingly easy to learn to use, despite its unusual structure.

MORE PROGRAMMING LANGUAGES

When this book was written, MacPascal was by far the best-structured programming language available for the Macintosh. It is, however, far from the only Macintosh language. Two BASICs, a version of Forth, two implementations of FORTRAN, at least one Logo, a LISP, an assembler, two implementations of the Modula-2 language, and at least three versions of C should be available for this time. The focus of this book is using applications productively, not programming, but many useful programming language options are also available for the Macintosh.
Microsoft BASIC

The first release of this BASIC implementation is not very impressive. It provides three windows, one with your program listing—the text of your program—one for the output of the program, and one for entering new lines into the program (Figure 12.3). The program listing window cannot be used for editing, an awkwardness that seems to have no inherent justification. The speed with which Microsoft BASIC programs are executed on this system is not very satisfactory. Many 8-bit microcomputer BASICs do as well or better. Perhaps Microsoft will improve its BASIC in the future.

Despite its faults, Microsoft BASIC is being productively used by many Mac owners. A movie producer uses Microsoft BASIC for the Mac to compute moments of force for constructing camera scaffolds and cranes. Complete communications systems, such as MacTEP, have been created in this programming language. Because it was the first language available for programming on the Mac, most users' groups have more extensive libraries of programs in Microsoft BASIC than in any other language.

![Figure 12.3 Microsoft BASIC](image-url)
MacForth

MacForth is an implementation of the programming language Forth for the Macintosh. Many uncomplimentary things have been said about Forth by people who are not Forth programmers. Forth is supposed to be very difficult to read (it has been called a “write-only” language). This makes it difficult to maintain large programs written in Forth. On the other hand, Forth code typically runs much more quickly than the equivalent programs written in BASIC.

Forth achieved great popularity a number of years ago when memory limits for small systems were much lower than now and programmers were looking for methods that required very little RAM. Serious Forth programmers seem to profess the language like a religion and to produce numerous puns about setting forth, starting forth, and so on.

Macintosh BASIC from Apple

Macintosh BASIC from Apple looks like a winner. Whether it is really a conventional microcomputer BASIC is another question. Like Pascal, it encourages structured programming practices. If this is so, it may be a terrific marketing coup on Apple’s part. Although Pascal is a much better programming language than BASIC (especially for large projects), its pre-Macintosh implementations haveterrified novice programmers, in part due to a number of good reasons discussed in Chapter 11. If Apple has created a friendly Pascal disguised as a BASIC implementation in order to reassure novices, they may have performed a fundamental service to the future of personal computer programming.

Fortran

FORTRAN was the first higher level programming language, and it shows its primitive origins in its design. Still, many engineers are much more familiar and comfortable with FORTRAN than with other programming languages, and it is to be hoped that they will be pleased with one or more of the implementations of this language on the Mac.
LISP

LISP is a very odd language, one in which a great many parentheses are used, as in this statement:

\[(\text{CDR (CDR (CDR (CDR X)))})\]

When this statement is executed, LISP responds with the list “X”, minus its first four elements. It is an old language; older than any of the others listed here except FORTRAN, but it is still in high favor in academic computing circles, which is much more than can be said for the next most ancient language on this list, BASIC. LISP is the most widely used computer language in the field of Artificial Intelligence, the study of methods for producing apparent intelligence in machines.

Logo

Logo is sometimes thought of as a graphics programming language for children, but it is really much more than that. Logo is more closely related to LISP than to any other programming language, and it is possible to create quite sophisticated programs in the language. Many educators believe that Logo is a much better first programming language than BASIC.

Pascal

The MacAdvantage from SofTech Microsystems is a Pascal-based development system for the Macintosh.

Modula-2

The inventor of the Pascal programming language introduced Modula-2 as the successor to Pascal. It has many of the virtues of Pascal and corrects some of Pascal’s faults. Whether it is a good language on the Macintosh will depend on the quality of the implementations, which are not yet available at this writing.

C

C is a less structured language than Pascal or Modula-2, but it makes it easier for the programmer to make use of machine-level details than do those
programming languages. C is attaining increasing popularity at many universities that have minicomputers from DEC (Digital Equipment Corporation) equipped with the UNIX operating system. Most of UNIX is written in the C language, and C is usually the best-supported language on a machine that uses UNIX. As a result, many computer science graduates have become fervent proponents of C.

**UPGRADING YOUR MAC**

There are many kinds of peripheral devices that will help to make your Macintosh much more useful, but the most obvious Macintosh upgrade is the conversion of a 128K-byte Mac to a 512K-byte machine. While there are a number of substantial advantages to making the upgrade, many users will decide that they can get along very well on the original 128K-byte system.

**A Fat Mac Upgrade**

The conversion is accomplished by replacing the entire main logic board of the Mac. The most obvious difference between the 128K and 512K systems is that they use different RAM chips to supply the Mac's memory. The 128K Mac uses 64K-bit RAM devices. These integrated circuits each provide a little more than 65,000 bits, enough storage for 8,192 characters. Since there are sixteen RAM chips in a Mac, the 64K-bit devices provide a total of 128K bytes. In a 512K Mac—affectionately dubbed the "Fat Mac," although it weighs no more than the original version—the RAM chips are all 256K-bit chips, each providing enough memory for more than 32,000 characters. Sixteen of these integrated circuits provide 512K bytes, storage for 524,288 characters.

**Why a 512K Mac?**

Moving up to a 512K Mac is essential only for those Macintosh users who want to use larger programs that require more memory. One such program is ThinkTank512. This program lives up to the "idea processor" label to a much greater extent than does ThinkTank128 (described in Chapter 6). ThinkTank512 permits the development of much larger outlines. More importantly,
however, the lowest nodes in an outline can be small bodies of text, or even pictures. ThinkTank512 is not only a tool for developing ideas but also for presenting them, through the use of a “slide show” option built in to the application. Just as you might use a slide projector to present a set of slides during an oral presentation, you can use the Mac to present a series of text or graphic displays on the Macintosh screen, advancing from one to another at the click of a mouse button. It is even possible to present animated sequences of graphics in ThinkTank512. This application is a good example of how the Mac’s power can be even more fully exploited in a Fat Mac application.

Another reason for purchasing a Macintosh-512K or an upgrade to 512K bytes is that it enables you to run major applications like Lotus Jazz. Jazz is one of the highly integrated applications available for the Mac. Lotus and Ashton-Tate are two companies that have a good deal of expertise in producing comprehensive applications that include many of the capabilities provided by separate applications of the type described in this book. A single such integrated application may include word processing capabilities, spreadsheet features, data filing, charting, and communications functions.

Integrated application packages, such as Lotus Symphony and Ashton-Tate’s Framework, have been very popular for computers with conventional user interfaces, such as the IBM PC. These applications provide even more application functions in a single package than does Lotus 1-2-3, the very successful product for the IBM PC that played an important role in establishing the integrated software concept. The Mac achieves a high degree of integration across different applications, thanks to the consistency of the user interface and to the transferability of data among applications through the Clipboard. Will the additional advantages provided by a single integrated application lure Mac owners into using such systems in preference to separate applications?

Jazz, previewed as this book was nearing completion, offers a number of worthwhile features. For users who must frequently move back and forth among spreadsheets, word processing documents, structured data files, charts, and communications with other computers, Lotus Jazz offers great increases in speed and convenience, compared to the use of separate applications. The speed enhancements are due to two factors. One is that more than one of the Jazz functions can be resident at once on the Mac. For example, you can have both a spreadsheet and a report that refers to the spreadsheet open at the same time. Going back and forth between them can be as quick as a mouse click. The second speed factor is due to the execution speed of the
Upgrading Your Mac 265

separate function modules themselves. Each was hand-crafted in assembly code for speed and economy.

Speed is not the only reason and perhaps not even the major one for using Jazz. Perhaps its greatest strength is a capability called Hot Views. Jazz supports conventional cutting, copying, and pasting from the Clipboard. Hot Views is a new feature that allows you to link a portion of one document to another. For example, you can first create a bar graph that represents a portion of a spreadsheet, and then put a Hot View copy of the graph in a word processing report. Once this is done, you can make changes to the spreadsheet, and they will automatically be reflected, not only in the chart, but also in the text report. In fact, Hot Views are updated instantly even in closed documents.

One of the nice aspects of using Hot Views is that it results in a much more natural-looking display on the screen. For example, you can get mail merge functions (such as those described for Microsoft Word in Chapter 10) by having a Hot View in a letter from a data base. In the conventional approach to mail merging one uses unnatural-looking commands, such as "<<DATA branchmanagers>> <<name>> <<address>>" in a word processing document to describe what data fields are to be used. In Jazz, the Hot View automatically pastes actual text from the data base source into the letter.

Macintosh users who need all or several of the five Jazz functions—word processing, spreadsheet, database, graphics, and communications—will appreciate having one consistent user interface for all five functions. In addition, the spreadsheet is particularly large (8192 rows by 256 columns), fast, and well-designed. Many business users of the Macintosh will find that they can often keep Jazz loaded all day on their machines, going back and forth quickly among tasks that require its different functions, without ever having to take the time to quit an application.

If the components of the integrated application are not truly competitive with their stand-alone counterparts from other software companies, the integrated applications packages may make a much smaller impact in the Macintosh world than they have in the IBM PC world. Mac users are in a position to demand very high quality in each applications area, since they already have a high degree of integration thanks to the Macintosh user interface.

While a 512K Mac is not essential if you don’t need the applications that demand its capabilities, it does have a number of advantages in working with conventional applications. Most substantial applications, such as those discussed in this book, make extensive use of memory overlays or code segments.
This means that when an application is programmed it is divided up into different pieces that can function somewhat independently. Not all the pieces of a program are loaded from disk into the Mac’s RAM at one time, in order to leave memory free for user data. When a new code segment is brought into the memory of a 128K Mac, there usually isn’t enough room to keep the earlier code segment, so it is purged. When its functions are called for again, it must be read from the disk again, a process that results in a delay before the application can go on. On a 512K Mac, code segments are much less likely to be purged, so large applications will run significantly faster, particularly after each code segment has been used at least once, and therefore loaded from disk into RAM.

You must be the judge of the value of the 512K Mac for your own work. Try the applications you actually plan to use on a Fat Mac at your dealer’s to determine whether the price differential is worthwhile for you. In the long run, when RAM prices have declined, probably most active Mac users will upgrade their machines. Doing so is likely to be less expensive later on, when Apple can purchase 256K-bit chips less expensively.

Whatever you do, don’t try to upgrade your 128K Mac to 512K by yourself. It is more than a matter of desoldering the 64K-bit chips and soldering in 256K-bit chips in their place. For one thing, that is likely to result in the destruction of one or more traces on the board itself, thereby ruining the board. Worse, several custom-integrated circuits manufactured by Apple must also be replaced, in addition to the RAM chips. The Mac upgrade is one place where do-it-yourself just doesn’t pay off.

A Hard Disk Upgrade

Most other upgrades to your Mac have to take place outside the box and can therefore be thought of as peripherals rather than as upgrades. One very special kind of peripheral addition, however, is a hard disk drive. Working with a hard disk can result in such substantial improvements in performance that adding one must count as the most significant upgrade possible after the Fat Mac upgrade.

The most obvious consequence of choosing a hard disk is the ability to store more. Figure 12.4 shows the disk window for a MacDrive disk from Tecmar. Tecmar makes a number of hard disk drive models for the Mac.
Some have fixed hard disks of 5 or 10 megabytes (5 or 10 million characters). Others have removable 5-megabyte cartridge disks. Such a cartridge is just over 4 inches square and less than 1/8 inch thick. Keep a few tens of these around and you can have easy access to hundreds of megabytes of storage.

The directory window in Figure 12.4 is for a 5-megabyte removable disk system. Almost 3 megabytes of storage is in use, most of it documents tucked away in labeled folders. This disk has hundreds of documents from many applications. A hard disk helps you fully realize the integration capabilities of the Mac, because you can easily move from application to application, preserving the contents of your Clipboard without having to keep reinserting the previous system disk. Copying and pasting between documents, especially between documents from different applications, is much easier using a hard disk than with floppy drives alone.

At least as important an advantage as capacity is disk access speed. Despite the many advantages of the Apple floppy system, it has one glaringly obvious defect. It is slow. The problem is that the floppy has to keep changing its speed of rotation before it accesses information from different portions of the disk, and that takes time. The hard disk doesn’t have that problem, and it
rotates much more quickly and has much faster head movement than does a floppy drive. It can access information much more quickly.

Macintosh hard disks can't transmit the information to the Mac quite as quickly as the floppy drives can, but accessing the information is usually the more important determinant of disk effectiveness. Working with disk-based applications such as Microsoft Word or Telos Filevision becomes even more of a pleasure when new data is brought in so quickly from a hard disk. Unfortunately, there are some problems in using a hard disk that stores many documents. Opening new files and leaving applications take a very long time when there are many files on the disk. Apple may release a new version of the Finder that reduces the severity of this problem.

Corvus developed software that seems to solve, at least partially, the speed problems that pop up when disks have many files. A Corvus Omnidrive disk is separated into a number of volumes, each of which is treated as a separate disk by the Mac's applications. With an Omnidrive, you can keep many documents, distributed among several volumes on the hard disk.

When you work with a Tecmar hard disk, you must be careful to use the Finder's Eject option when you have finished working with the disk before you turn it off. You have to choose the Eject option even if the hard disk is a fixed, non-removable type. It may seem strange to try to eject a disk that is fixed, but the Directory—that part of the disk that tells where all the files are located, how long they are, and so on—gets updated by the Finder when you eject the disk. If you forget to do this, your new documents will effectively disappear from the disk when you turn it off. If your hard disk has a lot on it and you are using a 128K-byte Mac, you may get a message telling you that there isn't enough memory to eject your hard disk. Don't worry about this. You can just click the OK button and turn off the drive and the Mac at this point. The directory has already been updated.

The Tecmar and Corvus hard disk drives provide additional speed for using the Mac in several ways. They load applications and documents more quickly than do floppy drives, of course. They also make it faster to use complex applications. Portions of a complex application in separate code segments are swapped into RAM from disk much more quickly with a hard disk. In this respect, a hard drive gives you some of the advantages of a Fat Mac.

However, having a fully utilized hard disk drive can impose serious speed penalties as well as provide advantages. Once your drive has more than 150 or so files on it, you will notice that it takes longer to quit an application on the
hard drive than on a floppy drive. Choosing the Open option when there are a large number of files to choose from is also a time-consuming process. If you are a real speed freak, you may find that you prefer to restrict the number of files on your hard disk.

The Corvus Omnidrive system partially overcomes this problem by splitting a hard disk into a number of separate volumes that act like separate disks, each with a modest number of documents and applications. A Macintosh application provided by Corvus called Volume Manager lets you partition the disk into volumes. Another application, Mount Manager, is used to mount a volume, making it accessible to your Mac. A mounted volume is visible as a disk icon on the desktop. By not placing too many files on any single volume, you can enjoy the capacity advantages of a hard disk without giving up its speed advantages.

MACINTOSH ACCESSORIES AND PARAPHERNALIA

The most important accessories for using your Macintosh productively are the right applications. This book has described some of the most important Macintosh applications programs. As developers and users become more familiar with the Mac’s capabilities, even more powerful and original applications will appear. Be on the lookout for new ways to use your Mac to get things done.

Magazines

Early product announcements and reviews can be found in Infoworld. Reviews in Byte magazine are likely to appear much later but are usually thorough. In-depth coverage of Macintosh products and techniques is provided by MacWorld. Additional Mac coverage can be found in a special Mac section of the magazine APlus. Since the Mac was undoubtedly worth what you paid for it, you might as well spend a few dollars more and subscribe to major Macintosh magazines.
Newsletters

There are a number of Macintosh clubs and newsletters that can serve as sources of information about the Macintosh, its applications, and its peripherals. Here are several:

<table>
<thead>
<tr>
<th>Club Name</th>
<th>Location</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Club Mac</td>
<td>Boulder, CO 80302</td>
<td>$35/yr</td>
</tr>
<tr>
<td>ICONCEPTS</td>
<td>Athens, TX 75751</td>
<td>$18/yr</td>
</tr>
<tr>
<td>Semaphore Signal</td>
<td>Aptos, CA 95003</td>
<td>Free</td>
</tr>
<tr>
<td>ICON</td>
<td>Santa Clara, CA 95052</td>
<td>$20/yr, plus $20</td>
</tr>
<tr>
<td>ICON Concept</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

You are also likely to find a local computer club for Macintosh owners or a Mac special-interest group in an Apple computer club. For an opinionated and unusual perspective on many matters related to the 68000 processor, including occasional gibes at the Macintosh, you might enjoy

<table>
<thead>
<tr>
<th>Club Name</th>
<th>Location</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>DTACK GROUNDED</td>
<td>Santa Ana, CA 92705</td>
<td>$15/yr</td>
</tr>
</tbody>
</table>

Hard disks

A hard disk drive, I have suggested, is more than an item of peripheral equipment in an ordinary sense. It is a significant upgrade to your Macintosh system, one that offers addictive speed and capacity. A hard disk makes it
much easier to use your Mac as an integrated environment in which you pass data among applications.

A second floppy drive

A more modest mass storage add-on for the Macintosh is a second floppy disk drive. Having a second floppy drive offers a number of advantages that expand your effective use of the Mac. Obviously, more storage space is available to your applications programs. If you are using an application that can create documents the size of which depend on the amount of disk space available, you will be able to create much larger Mac documents if you have a second drive, since the disk in that drive need not have system or application files. Examples of such disk-size dependent applications include Microsoft Word and some data base applications. (Other data base applications, such as DB Master, can make use of documents that are distributed across several disks.)

There are other advantages to having a second floppy drive. One is that you can use a system disk that has a very large system folder. You can have a large Scrapbook, a full Note Pad, and a large number of fonts installed. Even though most of your system disks will be occupied by such support files, you can still make use of application functions that require a good deal of disk space by using disks with only applications and documents—but no system files—in the second drive. This makes it possible to have a full system disk but still print sizable documents from MacWrite, an application that temporarily needs lots of disk space for a print document.

Probably one of the most important reasons for investing in a second floppy drive, however, is that it will encourage you to back up your documents more often. If you make regular backups, some day something will happen that will make you glad you do (see chapter 3).

Printers

Printers are another major kind of peripheral equipment. The three major types of printers for the Mac are high-resolution dot matrix printers (like the Imagewriter), laser printers, and fully-formed-character printers. Laser
printers can be used both for graphics and to produce letter-quality text, so they can take the place of both the dot matrix and the fully-formed-character printers. Because of the expense of laser printers, some Macintosh-equipped offices may choose to get by with a combination of the two less-expensive types of printers.

**Graphics Peripherals**

Graphics peripherals offer special features for users of graphics composition applications. Graphics input devices include digitizing tablets and video digitizers, discussed in Chapter 10. Peripheral graphics output devices include pen plotters and screen projection systems. A **Pen Plotter** draws lines on paper with colored pens under the control of computer software. A **Screen Projection System**, such as the ones from Professional Data Systems (220 Redwood Highway, Suite 120, Mill Valley, CA 94941), can be used to display a giant Mac screen for presentations to large groups.

**Voice**

Voice input and output devices are two more types of peripherals that can be added to the Mac. Voice output devices usually sound a little odd, but with half an hour’s experience, most people find that they are quite easy to understand. Of course, the Mac has built-in sound production capabilities that can match most inexpensive add-on speech synthesizers for comprehensibility. SmoothTalker from First Byte, Inc. provides this capability.

Voice input systems do not yet recognize human speech accurately enough for many practical purposes. Claims of 95 percent and 98 percent accuracy should be examined very skeptically. Find out whether the conditions under which these accuracy rates were obtained are at all similar to the conditions under which you would like to use the system. Be aware that even limited accuracy usually requires using very small sets of words to be recognized. If possible, try the system using the vocabulary you plan to use. Finally, ask yourself whether you are willing to put up with an input device that misunderstands your input at least 2 percent of the time. Would you use a keyboard that got one out of every fifty keystrokes wrong?
Modems

Modems are another major peripheral category. If you can afford it, a modem with higher speed capabilities is preferable. When your time is of value, it doesn’t make sense to idle at the keyboard, waiting patiently for the screen to fill with input from a remote computer. On the other hand, having a 1,200/300 baud modem doesn’t necessarily mean that you should always use the higher speed. If your usage of a remote time-sharing computer involves long periods of thought while you examine information on the screen and consider what to do next, you should check out the structure of the time-sharing system’s rates. Some such computer systems charge four times as much for connect time on the higher speed lines, or more. If you are not transmitting in either direction for a good part of the time that you are hooked up to a system, there is no point in paying a premium for that “dead time.”

Media and Supplies

“Media and supplies” is a fancy name for disks, ribbons, and paper. Floppy disk manufacturers, observing the success of the Mac, have jumped on the 3.5-inch disk bandwagon. At the time I wrote this, Mac-size micro-floppies were available from Apple itself, BASF, Hewlett-Packard, Maxell, Sony, and Verbatim. Many more sources have undoubtedly appeared on the scene by now.

Continuous-form printer paper is available from many sources. Since your Mac is an ideal system for constructing presentation-quality printed documents, you may want to buy boxes of 20-lb. paper with micro-perforated edges (sometimes called “laser cut”). At first and even second glance, sheets of such paper, once separated, are not distinguishable from ordinary 20-lb. bond. It gives your printed documents a more professional look than the somewhat less expensive papers.

Disks and paper are one area in which prices greatly vary, to all appearances without regard for quality. If you use the Mac a lot, you will probably wind up buying a great number of blank disks and a good deal of paper in the long run. When you encounter a good buy, consider purchasing in large quantities.
Mac Support Stuff

Skim through one of the major Macintosh magazines and you will find advertisements for carrying cases, printer stands, mouse pedestals, furniture, and all manner of other objects customized to support your use of the Mac. One of the most useful such items is one or more high-capacity disk cases. They provide dividers that let you separate your Mac disks into labeled logical groups. Vendors include Evco (100 West North Street, Box 70, Deforest, WI 53532).

A catalog of Mac support materials and just plain fun stuff—clothes, coffee cups, jewelry, and office accessories—for fans of Apple Computer is available from The Apple Collection (P.O. Box 306, Half Moon Bay, CA 94019).

FUTURE MACINTOSHES

Apple does a good job of keeping its Macintosh secrets, but a survey of recent technology trends suggests some interesting possible futures for the Mac. Some of these advances are unlikely to occur before 1986, and some will come much later, if at all. It is nice to know, however, that the Mac’s design is one that should adapt well to a number of expected advances in technology. With a bit of luck, your next computer will be one that fits snugly into the Apple 32 family, giving you an upgrade path that lets you take your hard-earned Mac documents along with you.

The first type of advances to expect in Macintosh technology should be more of the same. That is, the next generation of Macs may have even more memory and access to higher-capacity, faster disk drives. A Mac with a megabyte of RAM and a 50-megabyte disk drive is a very achievable design goal.

Watch for progress in the size of future Macs, as well. The biggest consumer of space in your Mac’s case is the display tube. If it could be replaced by a flat panel display, similar to the optional liquid crystal display available for the Apple //c, then the Mac could be much smaller. In addition to eliminating the relatively large and heavy monitor, many of the functions of
the analog circuits card would be replaced by the liquid crystal component of such a "Flat Mac." When I'm hurrying through airports with my Mac case slung over my shoulder and my attaché case in the other hand, I fantasize about the day when I can leave the old Mac on my desk and just slip a Flat Mac into the briefcase when I leave on a trip.

Many users of color computers have asked when a color Mac will come to be. Color is likely to be available through an external monitor sometime soon. More memory is required to support color, so the color Macs will probably also be Fat Macs. It may be a long time before color display technology catches up to Macintosh standards for the in-the-box monitor, however. The Mac was made to be a machine that you can sit and stare at all day long. Its crisp, rock-steady display doesn't tire your eyes the way any current technology color display does. When color monitors can deliver the kind of performance the Mac demands, Apple will be ready to produce Macs with that capability. The low-level Quickdraw graphics primitives are already prepared to handle color. An external color monitor would make it possible to use color when necessary, without relying on it as your primary display device. A color-only Mac is still going to take some time.

Another trend to look for some years from now is "supercharged Macs." Motorola has just begun producing the 68020 chip, a processor with even more power than the 68000 but one that is software-compatible with it. Furthermore, it consumes even less electrical power, making it a strong contender for the brains of a battery-powered Flat Mac.

Enjoy your Mac with confidence. You are riding the wave of a powerful new technology that hasn't even begun to crest.
MOUSE SKILLS

INTRODUCING THE FIVE MOUSE SKILLS  277
SELECTING AND UNSELECTING  280
CHOOSING FROM MENUS  283
DRAGGING  286
WINDOWS  286
DIALOG BOXES  294
KEYBOARD COMMANDS  295
MULTIPLE SELECTS  299
MANIPULATING WINDOWS  300
PROBLEMS AND TECHNIQUES  308
INTRODUCING THE FIVE MOUSE SKILLS

To make effective use of your Macintosh, you must learn how to use the mouse, since mouse actions are used to do almost everything but enter text. The physical skills involved are very simple. There are only five: pointing, clicking, double-clicking, shift-clicking, and dragging. The effects of all five of these actions depend on the location of the mouse pointer.

Usually the mouse pointer looks like a slanted arrow on the screen, like this:

but it often takes on other shapes as well. The shape of the mouse pointer tells you what kinds of things you can do with objects you are pointing to. When the pointer is the slanted arrow, the mouse is usually pointing in an area in which there are icons you can select.

The mouse pointer takes on different shapes in other environments. For example, when the mouse is pointing at text that can be edited, the shape of the pointer is an I-beam.

When the pointer is an I-beam, you can click the mouse button to select a new point for text insertion or backspacing.

Other pointer shapes are used by particular applications. The MacPaint application, for example, has a large set of different mouse pointers that are used when the mouse takes on different functions. Thus, when you are using the mouse to select an object of interest, the mouse pointer is a small lasso—

When you use the mouse to click dots on or off in MacPaint, the mouse pointer is a picture of a pencil—

and so on.

Pointing

What could be simpler? To point with the mouse, simply move the mouse on your desktop until the pointer rests on the object you want to point to. The symbols used for mouse pointers—the slanted arrow, the I-beam, the lasso,
and so on—each have a single dot that specifies where the symbol is really pointing. This dot is the pointer’s hot spot. Figure A.1 shows the hot spots of some common pointers.

Clicking

To click the mouse, you simply depress and release the mouse button. The only aspect of clicking that may require some practice is learning not to move the mouse as you click. Beginning Mac users sometimes accidentally move the mouse while releasing the button. This can result in a wrong selection—a choice just above or below the one you actually wanted. If this happens to you, just make the selection over by repositioning the mouse pointer and clicking again.

**Figure A.1 Hot spots for five pointer shapes**
Double-Clicking

A double-click is just what it sounds like. To double-click, you simply click twice in rapid succession, without moving the mouse between clicks. The Macintosh can be set to require any one of three different minimum intervals between clicks in order for the two clicks to be counted as a double-click. This option is available from the Control Panel accessory in the Apple pull-down menu—the leftmost object at the top of your screen. As you gain some expertise with your Mac, you may want to use the control panel to set the double-click interval to the shortest value (Figure A.2). This will make it easier for you to click something and then click it again without having the system erroneously interpret the two clicks as a double-click.

Shift-Clicking

Shift-clicking is used to select more than one object at a time. For example, you can point to an icon and click the mouse button to select it, then point to a different icon and shift-click to select that one too. You don’t have to have the Shift key down when you select the first item, only when you want to add to your selection.

To shift-click, point the mouse and click while holding either shift key down. You have to push the shift key down before you press the mouse button in order for the Mac to see the action as a shift-click. Aside from the fact that it is a two-handed operation, shift-clicking is not more difficult than ordinary clicking.

Dragging

Dragging is used to move objects on the screen. You point to the object, depress the mouse button, and then slide the mouse to a new location, without releasing the mouse button. When the mouse pointer (or the object) gets to where you want it, release the button.

These five simple mouse skills are used in many different ways. Turn on your Macintosh now, and let’s explore some of them.
SELECTING AND UNSELECTING

If you’ve used the Macintosh at all, you’ve probably often selected items by pointing and clicking the mouse. Turn on your Macintosh now and insert a system disk, such as the Write/Paint disk. Clear the desktop by clicking in the Close Box of any open window. This is the small empty square at the left end of a window’s title bar. Note the position of the pointer in Figure A.3. After you’ve closed the disk window, only the disk icon—usually in the upper right corner of your screen—and the trash icon remain, along with the menu bar at the top of the screen. See Figure A.4.

Selecting an object tells your Mac that you want to do some work with that object. You can select disks, applications, and documents, as well as many special objects available through various applications. On this bare desktop, the Write/Paint disk is the currently selected item. Its dark appearance means that it is the selected icon.

Because it is currently selected, you can perform certain actions on Write/Paint. To see what those actions are, select the File menu from the menu bar. Move the mouse until its pointer rests on the word “File” in the menu bar. This is the title of the File menu. To select the File menu, press down on the
Figure A.3  Clicking the Close Box of the Write/Paint window

Figure A.4  The bare desktop
mouse button. As long as you hold down the button, the File menu is displayed (Figure A.5). Note that the File menu choices appear in two different intensities. Currently available choices appear in solid black letters, while unavailable options are words in gray. These dimmed commands are ones that cannot apply to the currently selected item. In this picture, only the Open, Get Info, and Eject commands can apply to the currently selected disk.

Release the mouse button while it isn’t pointing to any command in the menu, and the File menu disappears. Now unselect the currently selected disk by pointing anywhere on the desktop where there is no object—no icon or menu item—and click the mouse. You’ve just selected nothing, and thereby unselected the disk. A glance at the screen shows that the disk is not currently selected (Figure A.6). It now appears as a white icon against the gray background of the desktop, rather than as a black disk icon. (On your display the slanted arrow pointer will be in a different position that reflects the position of the mouse on your desktop.)

Figure A.5 The File menu choices available when a disk is currently selected
CHOOSING FROM MENUS

The concept of choosing from a menu is something like the concept of choosing an object. In both cases, you use the mouse to point to a selection, but in the case of a menu item your choice is a command. The physical action of choosing from a menu with the mouse is very similar to dragging.
A PULL-DOWN MENU is a list of choices that is ordinarily hidden except for the title of the menu. The top line of your Macintosh display has a number of pull-down menu titles. The leftmost one is a picture of a small apple with a bite out of it. Pull down the Apple menu by pointing to the apple icon and holding down the mouse button. Again, don’t just click it and let it go. Hold it down. The menu that appears lists the set of standard applications called the Desk Accessories (Figure A.8).

Still holding down the button, drag the mouse down slowly. Notice that each menu item “lights up”—appears in inverse video—as you point to it. Move the mouse cursor to the Alarm Clock option and finally release the button. Two things happen now. First the Apple menu disappears. Like a window shade that won’t stay down if you stop holding it, it pulls back up under its title, the little apple icon. Second, a small window appears on your screen with the current time (Figure A.9).

You have used the mouse to choose the Alarm Clock from the Apple menu. The apple icon is always the leftmost menu title in the menu bar. Each
Figure A.8 The Apple pull-down menu

Figure A.9 The Macintosh clock window
word or symbol in the menu bar is the title of a pull-down menu. All the pull-down menus are used the same way. You have to drag the mouse pointer down to the selection you want to make from the menu.

**DRAGGING**

Place the mouse pointer on the trash can icon, hold down the button, and slide the mouse to the left a bit. A dotted-line rectangle about the size of the icon moves along with the mouse pointer. Now release the button, and the trash can will snap from its old location to the new one marked off by the rectangle. You *dragged* the icon to a new position.

Now point to the clock window you just opened. Put the point of the mouse arrow inside the clock window. Depress the button and, holding it down, drag the clock to a new position on the screen. If you now put it to the right of the trash, your screen looks something like Figure A.10. Now point to the close box for the Alarm Clock window and click it to put it away.

The dragging skill has many uses in addition to choosing commands from pull-down menus and moving windows and icons around on the desktop. Dragging can be used to pull a copy of a document from one disk to another, for example. In some applications, dragging techniques are even used to change the shape and size of objects.

**WINDOWS**

*WINDOWS* are so named because they provide you with a view of your file much like the view that you would get by looking through the window of a house into the interior. If the window is too small to display everything inside, the user can increase its size by dragging the size box in the lower right corner. If there is still too much to display in the larger box, you can use a scroll bar to display the hidden information. A *Scroll Bar* lets you slide the information around under the window. Think how convenient it would be to be able to
slide a window around the surface of the house to see everything inside. Macintosh windows work something like this. Instead of moving the windows around, however, you use scroll bars to move information behind the window.

Scroll bars can be found at the right of a window, or at the bottom, or sometimes in both locations. A scroll bar at the right is used to move the information up and down behind the window. A scroll bar at the bottom has the function of moving the information to the left and right. See Figure A.11.

There are three different ways to use a scroll bar to change what is displayed in a window. Clicking an arrow will scroll the display one line in the direction of the arrow. Clicking in the gray area of the scroll bar will replace the current windowful of information with the next one—or the previous one. The direction of the scroll depends on which side of the scroll box you click. The Scroll Box is the white box in the scroll bar. It marks the relative location of the present windowful in the information source. Finally, you can scroll to a particular point in the information by dragging the scroll box to a new location in the scroll bar.
While a research scientist at the Xerox Palo Alto Research Center, Alan Kay conceived of a visionary approach to computing which he termed Personal Computing. He proposed the development of a handheld computer he called the Dynabook. The Dynabook was designed to weigh only a few pounds, but it would provide the computational power and mass storage capabilities of a small mainframe computer. More important than its hardware capabilities, however, were its user interface and its standard software.

The Dynabook can be viewed as the logical predecessor of the Macintosh in many respects. Like the Dynabook, the Mac is a graphics computer. The concepts of windowing and mouse pointing were invented in the course of the development of Dynabook prototypes.

The Dynabook was designed for use by people without prior training or experience in the use of computers. The goal was to create a personal information appliance capable of being used with as much ease as a pencil.

Figure A.11 Moving a document behind a window
Windows are very flexible. Their size and position can usually be
determined by the user rather than being fixed by the application. This makes
it possible to display more than one window at a time on the Mac screen. To
move a window on the screen, simply drag the window's title bar. That is,
point to the title bar, depress the mouse button, and then roll the mouse to a
new location, releasing the mouse button when you've lined up the window
outline where you want it. When you release the mouse button, the old
window will be erased and then reappear in the new location. Using this
technique, you can position two windows on the screen so that there is room
for both. If one is too large or not large enough, you can change its size with the
size box.

Opening Windows

In order to be able to use an object such as an application or a document, you
must first open a window for that object. If you want to work with the contents
of a disk and there is no window open on the desktop for that disk, then you
must open it. To open a disk, you first select the disk by clicking the disk icon,
and then choose the Open option from the File menu. Drag open the menu,
and release the mouse button when the pointer is on the Open option. See
Figure A.12.

Once a window for the disk is open, it is easy to select an application or a
document within the disk and open it. An application is a computer program,
such as a word processor, a game, or a spreadsheet program. A document is a
file of information that you create using an application. Macintosh documents
are very clever in that each one knows what application made it. When you
open a document, the application you must use to work with the document is
automatically opened at the same time (if it is on one of the disks whose icons
are on the desktop).

You can easily open a document in the same way you just opened the disk.
Select a document from the open disk window, and then choose Open from
the File menu. Figure A.13 shows the MacWrite Sample Memo being opened.
After you release the mouse button with the pointer on Open, MacWrite is
loaded automatically, along with the selected document, the Sample Memo
(Figure A.14). Click the close box to get out of the Sample Memo.

Using Open is only one way to open a window for an object. A shortcut is
also available. Simply point to the object you want to open and double-click.
Figure A.12 Opening a window for a disk

Figure A.13 Opening a document
The double-click method can be used to open disks, applications, and documents. For example, point to the Sample Memo icon in the disk window now, and double-click it. Double-clicking is a much quicker way of opening a document than first selecting and then using **Open** from the File menu. Even if an object has already been selected, you can still double-click it to open it.

In many applications, it is also possible to open documents from within the application. To do this, you first use the File menu to **Close** the current document you are working on. Then select **Open** from the same menu. A list of the available documents will appear in a small scrollable window. You can select which document to open from this window by clicking the name of the document you want and then clicking the **Open** box that appears next to the window of names (Figure A.15). To open a document more quickly, you can simply double-click the name.
Closing Windows

Just as there are two ways of opening windows—using Open or double-clicking—so there are also two ways of closing windows. Again, one is easier than the other. The hard way, which still isn’t very hard, is to pull down the File menu and choose Close (Figure A.16). Choosing this command closes the currently active window. You can always tell which window is currently active by the title bar. The window that has the band of tight horizontal lines on either side of its title at the top of the window is the active one. None of the other windows can overlap the active window, except for pull-down menus.

The easier way to close the currently active window is to click its close box in the upper left corner (Figure A.17). Try both methods for closing your disk window now.
Figure A.16  Closing the disk window from the File menu

Figure A.17  Closing the disk window with its close box
DIALOG BOXES

DIALOG BOXES are special temporary windows that appear on top of other windows to present choices to the user. Double-click the Sample Memo document to get into MacWrite with the Sample Memo text. Add one line at the beginning of the memo by positioning the insertion point there—put the I-beam pointer there and click—and typing the Return key. Now click the close box in the Sample Memo title bar. The window that appears, asking whether to "Save changes before closing" is a dialog box (Figure A.18).

Three choices are presented in this box, and you must pick one of them. None of the normal commands is available until you make one of the choices in the dialog box. In this box, "Yes" causes your revised version of the document (with the extra line at the beginning) to be saved in place of the old version. "No" closes the document, leaving the previously saved version intact. "Cancel" means that you want to cancel your previous command, namely, to close the window. If you choose this option, you will be able to go on editing the Sample Memo.

The user picks one of the three options by pointing and clicking one of them. As an alternative, the default choice—the one that has a double-walled box around it—can be selected simply by typing the Return key. Typing Return, in this case, would select "Yes" and would save the change. For now, though, just select "No"—point to and click the "No" box.

Some dialog boxes ask you to enter or edit some words in addition to choosing among boxed options. To see an example of this, choose the New option from the File menu in MacWrite. You now can enter some text in a MacWrite document window entitled "Untitled." Enter a word or two of blather, and then click the close box. You will again be presented with the MacWrite dialog box that asks whether you want to save your changes, as in Figure A.19. Choose "Yes" this time. Now a new dialog box appears (Figure A.20). This one wants you to type a name for the new document. A vertical bar at the beginning of the little input window—the rectangle just below the "Save current document as"—marks the insertion point. As soon as you type a bit here, the dialog box activates the Save option, which was previously dimmed (Figure A.21). Now that your blather has a name, it can be saved as a new document. Alternatively, you can choose to eject the disk in order to store BLATHER on some other disk. For now, unless you've developed a great
fondness for the blather you typed in, just click the “Cancel” box and proceed to get out of the application without saving the text, as you did before.

You will frequently encounter dialog boxes in your use of Macintosh applications. When they appear, you will usually have to make some choice before you can proceed.

**KEYBOARD COMMANDS**

The standard way of issuing commands is by means of the pull-down menus. As you’ve seen already, however, there are shortcuts to the pull-down menus. Double-clicking an icon is a shortcut to choosing the Open command from the File menu. Clicking the close box is a shortcut to close a window.
Figure A.19 The "Save Changes" dialog box

Figure A.20 The "Save As" dialog box
There are keyboard shortcuts for many other menu commands. Keyboard commands require that you type some letter *while holding down the Command key*. The Command key is the one with the clover-like symbol on it, just to the left of the spacebar. To issue a command from the keyboard, you hold this key down like a Shift key and type another key. It won’t take you long to learn the most common keys, and a little effort now will repay you in time savings later. While it is possible to get by using the pull-down menus, you might as well learn the keyboard commands too.

Most of the keyboard commands are easy to remember. In the File menu (Figure A.22), Command-D will duplicate a selected item; Command-I will bring up an information window about a selected item; and Command-E will eject the disk if it is the currently selected item. In the Edit menu (Figure A.23), Command-C will copy the currently selected item to the Clipboard. The Clip-Board is a part of memory reserved as a special buffer for moving information between applications or documents, or even from one part of a
document to another. The Copy command does not copy information directly to its final destination. It just puts the information to be transported onto the clipboard. The clipboard is only a temporary buffer. When your Mac is turned off, whatever is on the clipboard is wiped out.

Command-X is an instruction to cut the selected material. If you are imaginative, you might think of an X as having a shape like an open pair of scissors. Whatever is cut is deleted from its current position, and a copy is placed on the clipboard so that it can be pasted in elsewhere.

Some of the editing commands, such as Command-V for Paste, are a bit harder to remember. Command-V pastes the contents of the clipboard at the currently selected location. Perhaps the letter V is meant to remind us of a caret (\^), which is used in proofreading to mark an insertion point. Users of the WordStar word processing program are already familiar with the use of V for pasting.
MULTIPLE SELECTIONS

It is often useful to be able to select more than one object at a time. Sometimes you want to print more than one document. At other times you want to copy more than one document or application to another disk.

There are two ways to select multiple objects. If the objects to be selected are contiguous, you can just drag a box over the area that contains the files. In Figure A.24, three applications were all selected by dragging the mouse from a point just above and to the left of the MacWrite icon down to a point just to the right and below the Font Mover icon. All three could now be simultaneously removed, for example, by dragging any one of them down to the trash icon.

Suppose you want to select two or more items that are not contiguous. Dragging a selection rectangle over the items would include some items that are not desired. Here a different method can be used, *shift-clicking*. To select more than one item using *shift-click*, for every item to be selected after the...
first you must hold down one of the shift keys while you point to the item and click the mouse button (Figure A.25). Shift-clicking makes it possible to select two or more items separated by items you don't want to select. In this example, the MacWrite application and the Sample Memo document have been selected by shift-clicking. They can now be dragged as a unit. Give it a try.

If you type Command-I—for information—with these two objects selected, two information windows will open up on your screen, one for each of the selected objects (Figure A.26). Click the close box on the top window, and it will close up, revealing the second information window below (Figure A.27).

**MANIPULATING WINDOWS**

Almost everything you see on the Macintosh screen is presented in a window. Windows provide a very flexible approach to the presentation of information. Most windows can be manipulated in a variety of ways. You can change a
Figure A.25 Selecting separated objects

Figure A.26 Multiple information windows
window's size, its location on the screen, and which part of its document it presents.

Most windows have a Size Box in their lower right corner. By dragging this box, you can move the window's right and bottom edges, making it smaller or larger (Figure A.28). Try dragging the size box of your disk window up and to the left to make the window smaller. Point to the size box, hold the mouse button down, and roll the mouse pointer up and left. A set of dotted-line window borders shows the new window size. When it looks right, release the button. The window now has a new size (Figure A.29).

**Scrolling**

When a disk window is not large enough to show all the information that is available to you, you can use scrolling methods to get to the information you want. Recall the image of the information as a parchment scroll. To see
Figure A.28 Dragging a window's size box

Figure A.29 The smaller disk window
something not currently in the window, one should scroll the document underneath the window. This is made possible on the Mac through Scroll Bars. A Scroll Bar is either a horizontal bar at the bottom of the window or a vertical bar at the right. The scroll bars represent the length of a document. If a scroll bar of an active window is white, the entire document is displayed in the window. If the scroll bar is gray, then the window is currently displaying only part of the information available. You can use the scroll bar to see more.

Move your mouse pointer to the area of the vertical scroll bar just below the white Scroll Box on the right of the small disk window now open. Click the Scroll Bar here, and observe that the window scrolls to the next windowful below (Figure A.30).

You can also use the scroll bar to scroll the window just a little bit at a time. Point at the rightward-pointing arrow at the right of the horizontal scroll bar, at the bottom of the disk window. Click here, and note that the window moves the displayed directory just a bit to the right, rather than by a whole windowful (Figure A.31). You can also move the window part of the way through the

![Figure A.30 Clicking the Scroll Bar](image-url)
information, by dragging the scroll box. The length of a scroll bar represents the length of the information. If you want to display something about halfway through, drag a scroll box to the halfway point of the scroll bar.

Play around with scroll bars until you are comfortable with them. You won't damage your documents by experimenting with scroll bars.

Moving Windows

You can put windows anywhere you want them on the screen by dragging their Title Bars. Simply point to a window's title bar, hold down the mouse button, and roll to a new location (Figure A.32). As you drag the window, its outline moves to the new position. When you release the button, the window appears in the new location on the screen (Figure A.33). If you start to drag a window and change your mind, it isn't necessary to try to carefully reposition the rectangular outline back over the window. Instead, just push the pointer up
into the menu bar at the top of the screen. Windows can't be moved up to obscure the menu bar, so if you release the button here, the window will just stay in its original position.

Overlapping Windows

When several windows are on the screen, they sometimes overlap each other and part of the information is obscured (Figure A.34). To see an example of overlapping windows, choose the Calculator option from the Apple menu. As the currently active window, it will rest on top of the other windows in the same screen area. In order to make a window active, and thus make it overlap the others, all you have to do is click somewhere in the visible part of the partially obscured window (Figure A.35). Do this to the overlapped disk window, and it will partially obscure the calculator.
Figure A.33 The relocated disk window

Figure A.34 Overlapping windows
If you want to move a window that is partially overlapped by another window, you can do so without making that window the active window. Simply hold down the Command key while you drag the partially obscured window by its title bar. The window will move without coming to the fore and obscuring any part of the active window.

You use the same mouse skills—pointing, clicking, shift-clicking, double-clicking, and dragging—in all your work with the Mac.

**PROBLEMS AND TECHNIQUES**

**Ejecting a Disk**

New Mac users sometimes get the impression that the keyboard commands don't always work. For example, Eject seems to work only some of the time for them. Before a disk can be ejected, it must be the currently selected item.
Pay attention to the appearance of the disk icon, which is dark when it is selected. If some object in the disk directory window, such as a document or application, is currently selected, then the disk icon will be white and the disk can't be selected. Change the selection to the disk by clicking in the directory window, not pointing to any of the disk contents. Alternatively, you can click the disk icon itself. The icon will now be dark, and you should be able to eject it.

The Programmer's Switch

Everybody makes mistakes, and sometimes the only way to restart is to make the Mac REBOOT—start up all over again, just as it does when you turn it on. For example, if you turn on your Macintosh and insert a non-system disk, the Mac will spit the disk back out and the screen will show a sick Mac (Figure A.36). You can now insert a good boot disk, but the Mac won't do anything. The only way to get going again, if you don't have the programmer's switch installed, is to turn the computer off with the power switch in the back, then turn it on again. This is fine for every once in a while, but turning computers on and off is hard on them. Leave the Mac running if you plan to come back to it in a few

![Figure A.36 A Mac unhappy with its boot disk](image-url)
hours. If you have the programmer's switch installed, you can use the **Reset** button to make the Mac reboot without turning the machine off and on.

**Invisible Icons**

Some users find that after they try to drag document icons to the trash the documents are not always thrown away. They no longer appear in the disk window, but the disk space available doesn't change, and when they eject the disk they see the documents appear for a second on the desktop. What's going on?

These documents didn't quite make it to the trash. Instead, they were somehow misplaced under your disk window. Try dragging the window to one side to see whether you have some documents stranded on your desktop (Figure A.37). After you find them in this way, you can drag them to the trash or put them back in the disk directory. Remember that it is the position of the…

![Figure A.37 Documents stranded on the desktop](image)
pointer's hot spot—the tip of the slanted arrow—that determines what you are pointing at when you release the mouse button. Make sure it points inside the trash icon when you are throwing something away. You know you're on target when the trash icon turns black, showing that it is currently selected.

**Dragging Accurately**

Not everyone finds using the mouse completely natural. Try to pay attention to where the mouse pointer is located when you release the mouse button when you finish dragging. It is the position of the pointer, not the outline of the dragged object, that determines the destination. If you are still having trouble, try to make larger targets for the destinations of your drags. If you want to put something in the trash, first double-click the trash icon to open a trash window (Figure A.38). Then you can drag items from a disk window into the trash window, which is a lot easier to hit with the pointer than the trash icon itself.
Room for Mouse Moves

Some users find that operations that call for mouse movements are difficult because there isn't much free space on their desks. A common error is starting over when the mouse runs up against an obstacle. Suppose you start to drag something and your hand runs up against some object in your path. Just keep the mouse button depressed and lift the mouse up in the air. Move it back in the direction you came from, set it back on your desk, and continue rolling it in the direction of the drag. It may take a few attempts to get used to picking up the mouse while keeping the button held down, but you'll find you can do it easily with practice.
INDEX
Add another function, 165
Add New Records option, 257
Alarm Clock option, 284, 286
Align Center (Middle) option, 23, 84
Alphabets, foreign, 50-51
APlus, 269
Appearance, control of, 150. See also Fonts; Formatting
Apple, 12
languages, 261. See also specific languages
Apple Bus System, 209-211
Apple Collection, 274
AppleLine, 196, 199, 200
Apple Macintosh, 2-10
Apple menu, 7, 284, 306
Apple Pascal operating system, 202. See also
Macintosh Pascal
Apple 32 system, 197
Apple II, 5, 201, 202
Apple IIe, 210, 211
Applications, 9-10, 257
defined, 66
pixel-oriented, 42
word processing, 39-41. See also Word
processing
see also specific applications
Archive disk. See Disks, archive
Arrange menu, 222
Art
analysis, 160
pattern, 217
see also Drawing; Graphics
Assembly language, 239, 242, 259
Atkinson, Bill, 12
Backup disks
importance of, 69-71
techniques, 72-77
Backspace key, 44, 45, 47-48
Bar, vertical, 134
Bar Graph option, 108
BASIC, 42, 240, 243, 259
Macintosh, 261
Microsoft, 239, 260
Baud rates, 205-206
Brush Mirrors option, 16
Brush Shape option, 30
Budget summary example, 99-103
Byte, 43
C language, 239, 259, 262-263
Calculator, 7, 92
Calendar option, 183, 306
Cancel box, 294, 295
Cash flow, 188
Central processing unit (CPU), 2, 4
Change to Milestone option, 177
Characters, special, 50-51
Chart (Microsoft), 9, 12, 30, 104
application, 111, 120, 122
menus, 128, 188
and Multiplan, 127-129
project managing, 175
references, 129
size changes, 191-192
Charting in Multiplan, 104-106, 127-129
Charts, 110, 120-127
with MacPaint, 31
with Multiplan and Chart, 127-129
pie, 111-119
resizing, 122, 124
resources on, 129
Clear, 163
ClickArt, 215, 216
Clicking, 277, 278. See also Double-clicking
Clip art portfolio, 215, 217
Clipboard, 23, 25, 26, 264, 297
in charting, 27, 129
Macintosh, 8
using, 46-47
Close box, 280, 281, 291, 292, 293, 300
Close option, 47, 57, 291, 292
Club Mac, 270
Color (Mac), 275
Column width option, 86, 87
Command keys, 49-50, 65, 84
equivalents, 256
Multiplan, 107, 108
Command-A, 124
Command-B, 49
Command-Backspace, 48, 231, 233
Command-C, 26, 46, 88, 142, 297
Command-D, 53, 93, 178, 297
Command-E, 297
Command-F, 84, 106
Command-G, 84, 90
Command-H, 83, 84
Command-I, 49, 76, 297, 300
Command-L, 135
Command-M, 177
Command-R, 84, 135
Command-Shift-4, 79
Command-U, 50
Command-V, 26, 46, 88, 136, 180, 298
Command-X, 46, 88, 136, 180, 298
Commands, keyboard, 25, 61, 86, 87, 295-298
Comments, 245
Communication
electronic, starting, 197-199
methods
local area network (LAN), 195, 196-197, 209-211
local serial connect, 195, 196
micro-floppy disk, 195, 196
telephone connection, 195, 196, 197, 199, 208
types, 212
Communications . . . command, 206
Compiler, 243
Computer-to-computer communication, 194-212
Computers, desktop, 4
Concepts—The Magazine, 43
Control Panel accessory, 279, 280
Copy option, 25, 26, 46-48, 53, 142, 298
CoreEdit, 44
Corvus, 73, 268
CP/M operating system, 203
CPU. See Central processing unit
Critical path, 176, 187
Custom Rulers . . . . option, 222
Cut option, 25, 46-48, 136

Data disk, 27, 36
Data menu, 111, 117
Data form, 167, 170
Data series, 111
Dates menu, 183, 184
DaVinci Graphics Packages, 215, 217
DB Master, 257-259
Definition form, 165, 167, 168
Delete
   Backspace key, 47-48
   Cut, 47-48
Desktop Accessories, 6-7, 256, 284
Desktop Accessory Calculator, 92
Dial option, 208
Dialog boxes, 56, 72, 75, 111, 113, 256, 257, 294-295
Cancel, 294-299
communications, 206, 207
customs rulers, 222, 225
drawing size, 221
footnotes, 235
Format Values, 114, 115
phone, 208
print setup options, 140, 143
Save As, 296, 297
Save changes, 296
Set Report format, 259
terminal settings, 206, 207
text options, 124, 125
Digital Equipment Corporation (DEC) VT 100
terminal, 200, 201
Digitizer, Thunderware, 215
Digitizers, video, 214-215, 272
Digitizing tablet, 214, 272
Directories
disk, 64-67, 268
mini, 78
printed, 78-79
Disaster, 59-61
   recovering from, 36-37
Disk copying, 71-72
Disk directory, 64-67, 268
damaged, 65
views of, 68-69
window, 84
Disk drives, 4
   hard, 209-212, 231
Disks
   archive, 78
   backup application, 69-77
   boot, 65, 75, 76
data, 27, 36
document storage, 73, 74
ejecition, 308-309
floppy, 4, 70, 71
hard, 69, 71, 270-271
hard upgrade, 266-269
managing, 62-79
micro-floppy, 71, 73, 195, 196
mini-floppy, 63
working application, 73, 77
Write/Paint, 35, 280
Documents
   archiving, 103
defined, 66
Documents, continued
editing with Word, 236
long, 43, 234
making room for, 35-36
multiple, 37-38
opening, 289-290, 292
short, 43
transferring material between, 23-26
wide, 234
Dollar option, 91, 92
Dot matrix printers, 40, 271, 272
Double-clicking, 65, 277, 279, 291
Dragging, 277, 279, 286, 299
accurate, 311
with formulas, 93-94
outline heading, 139-140
to select, 45, 58-59, 127
size handles, 124-125
Drawing, 7-8, 11-38
menu, 248, 254
with mouse, 15
technical, 31, 32
Drawing files, 147, 149, 161-173
DTACK GROUNDED, 270
Dummy text documents, 53-55
Duplicate option, 53, 178, 179

Edit menu, 16, 19, 25, 30, 231, 233, 256, 297, 299
Edit Pattern option, 217
Editing, 43-45, 111
Element, 163, 165
Enlarge option, 226
Eject, 268, 308
Equal (=) symbol, 91, 92, 93, 99
Errors, 36, 250-251

FatBits, 14, 18, 21
shortcuts, 27-29
Fat Mac, 263, 268
Fields, 154, 168
File menu, 27, 37, 118, 150, 280, 282, 283, 289, 291, 292, 298
File Server systems, 200
File transfer process, 202-205
Files, 63-64
drawing. See Drawing files
sharing, 210
stored pictures, 147
system, 66
Filevision (Telos), 9, 12-14, 71, 218, 220
drawing files, 161-173
features, 147-150
menus, 147, 150, 151
printing lists, 154-157
range of uses, 157-161
Symbol Edition, 163-167
Fill Down option, 99
Fill Right option, 84
Filled Rectangle tool, 30, 33
Filled Rounded Rectangle tool, 23
Fill-in Oval tool, 20
Fill Oval option, 249
Financial statements, 80, 106, 108
FingerTip Software, 217
500 Menu Patterns, 217
Flat Mac, 274
Floppy disks. See Disks, floppy
Floppy drive, second, 271
Folders, 52, 53, 66-69, 73
Footers, 48, 53
Footnotes, automatic, 235, 236
Footprint, 4
Font Mover application, 52, 65-66
Fonts
Cairo, 12, 217-218, 219
changes, 49, 124, 127, 245
menu, 177
Monaco, 131, 134
removing, 52
sizes and styles, 40-41, 49-50, 134, 142, 177
FontSize menu, 131
Foreign alphabets, 50-51
Form letters, 229
Format changes, 48
Format menu, 56-57, 84, 106, 124
Format window, 111
Formatting, 111, 154-156, 235
Forms, creating with MacPaint, 34
Formula bar, 92, 93
Formulas in Multiplan, 91-99, 103-104, 105
FORTRAN, 239, 259, 261
FrameRect, 249
Freehand Line tool, 163
Fully-formed character printers, 271-272
Gallery menu, 114, 116, 122
INDEX 317

Games, 5, 12, 13
George Graphics, 38
Get Info, 76
Glossary option, 231, 233
Goodies menu, 16, 18, 30, 31
Graphics, 11-38
   peripherals, 272
   services, 214-217
   with word processing, 217-218
   invisible, 177
Group option, 222

Halt option, 247
Handles, 124-125, 147
Hardware, 2-4
Headers, 48, 53
Hide These option, 150, 152
Highlight Some option, 152, 153, 171
Highlighting, 150-154, 157
I-beam pointer, 44, 45, 47, 277, 294
IBM mainframe computer, 196, 199
IMB PC, 202, 204, 210
ICONCEPTS, 270
Icons, 16, 68-69, 72, 131, 163, 270, 310-311
Illustrations, intelligent, 158-159
Imagewriter printer, 8, 40, 227, 271
Information, 63, 149, 300-303
   exchange, 195, 197, 212. See also Computer-to-
   computer communication
   manipulation, 150-154
InfoWorld, 43, 81, 269
Insert Ruler, 48
Insertion point, 44-45, 134-136
Instant menu, 248, 249
Inverse video, 45, 46, 58-59, 294

Jazz (Lotus), 264, 265
Justification, 57-58, 84

Key Caps, 51
Koala Video Digitizer, 214

Laser printer, 8, 40, 227, 271-272
Lasso tool, 20

Layout menu, 177, 222
Letterhead text, 53, 54, 55
Link function, 171, 172, 265
LISP, 239, 240, 259, 262
List-formatting options (Filevision), 154, 156
Lists, printing, 154-157
Local area network (LAN), 195, 196-197, 209-211
Local serial connect, 195, 196
Logo, 239, 243, 259, 262

MacBASIC, 239
MacCats . . . 99 Ways to Paint a Cat with
   MacPaint (Flanagan), 38
Mac Daisywheel Connection, 227
MacDraw, 9, 12, 14, 149, 218-226
   menu, 228
   project managing, 175
MacDrive (Tecmar), 73, 212
MacForth, 239, 259, 261
Macintosh
   accessories and paraphernalia, 269-274
   clubs, 270
   future trends, 274-275
   graphics, 11-38
   operating system, 63
   upgrading, 263-269
Macintosh Pascal, 9, 42, 239-240, 244
   programming with, 238-254
   reserved words, 245
Macintosh techniques, 308-312
MACpac, 227
MacPaint, 7-8, 12-14, 16, 25-26, 27, 28, 42, 73, 147,
   149, 175, 218, 220, 268
   aids to use, 38, 214-217
   menus, 17
   problems, 30-38
   shortcuts, 27-30
   techniques, 15-23, 30-38
MacPascal. See Macintosh Pascal
MacProject, 9, 143, 175-188
   menus, 178, 180
   MacProject schedule, 188-192
   Mac the Knife, 215
   MacTEP, 197, 260
   MacTerminal, 9, 197, 200-205
   customizing, 205-208
   menus, 201
MacWork/MacPlay (Poole), 38
MacWorld, 43, 217, 269
MacWrite, 6-8, 12, 42-53, 230, 231
aids to use, 226-229
menus, 60, 61
techniques, 51-61
Magazines, 269. See also individual titles
Mail, electronic, 210
Mail merge function, 265
Mailing labels, 147, 158
Maps, 147-148, 160-161
intelligent, 157-158
Mathematical characters, 50
McPic, 215
Media and supplies, 273
MegaMerge, 229, 237
Memo writing, 8
Memory, 4, 257, 263, 256-266. See also Random access memory (RAM); Read-only memory (ROM)
Menus, 283-286. See also specific menus and applications
Micron Eye Video Digitizer, 214
Microsoft, 12, 43, 111. See also Chart; Word
Milestone, 177
Minus sign, 131, 135
Modem, 196, 199, 273
Modem7 option, 204, 205
Modula-2 language, 239, 259, 262
Mouse, 3, 312
drawing with, 15, 29, 214
in formula writing, 81
moving objects with, 147, 149
selecting objects with, 84, 86
see also Mouse pointer; Mouse skills
Mouse pointer, 3, 277-278, 311
shape, 44, 84, 277
Mouse skills, 276-312
clicking, 277, 278
double-clicking, 277, 279
dragging, 277, 279
pointing, 277-278
shift-clicking, 277, 279
Multiplan, 9, 73, 81-108, 127-129, 175
books on, 108
command keys, 107, 108
Disk Directory, 84
word processing, 229, 230
Multiplan Models for Business (Cobb, Cobb, Henderson), 108
Multiplication sign (*), 92
New option, 294
Newsletters, 270
Note Pad option, 200, 271

"OK" button, 117
Omninet (Corvus), 209-211, 268, 269
Open option, 47, 57, 65, 67, 269, 289, 291, 295
Option key, 51, 65
Option-key combination, 51
Outlines, 131-140
printing, 140-141
in text documents, 142-143
Paintbrush tool, 15, 16, 30
Pascal, 9, 12
American National Standard (ANS), 239, 240
UCSD, 239
see also Macintosh Pascal
Paste option, 25, 26, 47, 48, 53, 57, 89, 137, 142, 298
Pattern art, 217
Pattern menu, 23, 30
Pause menu, 247
Pen plotter, 272
Pencil tool, 27-29
Percent option, 90
Peripheral equipment, 270-274
Phone option, 208
Pictorial data bases. See Filevision
Pie charts, 111-119
Pixels, 5, 14
Plot Series box, 117, 118
Plus sign, 131, 135
Power of Multiplan, The (Williams), 108
Presentation Graphics on the Apple Macintosh (Lambert), 129
Print Catalog option, 27, 79
Print Draft mode, 38
Print Final mode, 37-38
Print Info options, 154, 156
Print Labels option, 158
Print merging, 237
Print option, 37, 55-56, 57, 118, 140, 142
Print server, 210-211
Print buffer, 227
Printers. See Fully-formed character printers; Imagewriter; Laser printer
Printing, 8, 37-38, 55-56, 154-157
Programming, 240-242, 245-254
Programming languages, 238-254, 259-263
Project managing, 174-193
Project Table option, 191
Protocol converter, 196, 199, 200
PTerm program, 202, 203
Pull-down menus, 256, 254, 285, 286

Quickdraw, 12-13

RAM. See Random access memory
Random access memory (RAM), 2, 43, 61, 63, 256, 257, 266
Read-only memory (ROM), 2-4, 65, 256
Rebooting, 309-310
Rectangle, 177, 178, 181
rounded, 177, 220
Reduce option, 226, 227
Repeat . . . until structure, 245
Reserved words (Pascal), 245
Reset key, 75, 310
Resource cost, 188
Resource Timeline option, 188, 189, 191
Restore option, 35
Return key, 86, 131, 294
ROM. See Read-only memory
Rotate option, 30
Rulers, 48
Run menu, 250

Save As option, 35, 54, 117, 203, 205
Save option, 57, 59, 61, 294
Scatter Plot option, 122
Schedule charts, 175-188
Scrapbook option, 6, 200, 271
Screen projection system, 272
Scroll areas, 137, 138
Scroll bar, 88, 179, 220, 286, 287, 304
Scroll box, 287, 304, 305
Scrolling, 29, 231, 302-305
Search option, 139, 141
Select All menu, 124
Selecting, 45, 47, 84-86, 182, 280-283, 299-300
accidental, 58-59
Send File option, 202
Settings menu, 206, 208
Shift-clicking, 47, 277, 279, 299, 300
Shift-dragging, 182

SHIFT-RETURN, 86
SHIFT-TAB, 86
Size box, 302, 303
Size handles, 124-125
SmartCable, 198, 199
Software, 2-4
Sort option, 139
Speed enhancement, 264-265, 268, 269
Spilling Paint Can tool, 20
Spreadsheets, 80-108
Stepwise refinement technique, 250-251, 254
Style menu, 177
SUM function, 93, 94
Symbol tool, 163, 165
Symbols, special, 50-51

Tab key, 48-49, 84, 86, 185, 229
Tasks, 177-180, 184-187, 189
Task Info window, 184-185
Telephone connect, 195, 196, 197, 199, 208
Telos Software Products, 12, 147
Terminal Settings option, 206, 207
Text, 40-42, 53-55, 56-58
accidental selection, 58-59
dummy, 53-55
moving, 57
selected, 45-46
styles and sizes, 8, 35
Text Only button, 203
Text option, 111, 117, 124
Text tool, 23, 163, 165
Text transfer method, 205
ThinkTank, 9, 74, 128, 131-140, 143-145, 175, 263-264, 265, 512
ThinkTank Extra menu, 139
Time-sharing system, 204-205, 206
Tinker menu, 150, 170, 171
Title boxes, 23, 25
Tools menu, 18
Transferring material, 23-26
Typesetting, 40

Undo, 30, 35
Ungroup option, 222
UNIX operating system, 263
Unselecting, 280-283
Untab, 48
User interface, 2, 4
  nonstandard, 10
  standard, 256-259

Video digitizer, 214-215, 272
View menu, 68
Voice peripherals, 272

Windowing, 6, 8
Windows, 6, 29-30, 68, 134, 167, 286-289, 301-308
  closing, 292-293
  directory, 56, 65, 66, 84
  menu, 248
  moving, 302-306
  opening, 289-291
Word (Microsoft), 9, 43, 71, 229-237
  menus, 232
Word processing
  applications, 39-61
  and graphics, 12, 217-218
  with Multiplan, 229
  with Work, 229-237
Write/Paint disk, 35, 280
MORE MACINTOSH BOOKS FROM SCOTT, FORESMAN AND COMPANY

MacCats: 99 Ways to Paint a Cat with MacPaint
This entertaining book helps you explore MacPaint's creative capabilities and master a variety of graphics skills on the Macintosh. Fully illustrated. By Flanagan. $9.95, approx. 192 pages

Multiplan Mastery on the Macintosh
Leads you from the basics to advanced concepts of spreadsheet design with Multiplan. Includes dozens of illustrations and sample applications. By Townsend. $15.95, approx. 224 pages, coming 4/85

Inside the Macintosh
Written for experienced programmers, this book explores the fundamentals of the Macintosh and its operating system in depth. Includes sample programs in BASIC, Pascal, and Assembler. By Twitty. $16.95, approx. 304 pages, coming 6/85

Microsoft BASIC Programming for the Mac: A Beginner's Guide
Learn to program in Microsoft BASIC. This book offers helpful programming techniques, numerous programs and examples, and advice on managing menus, windows, and the mouse. By Aker. $12.95, approx. 208 pages, coming 5/85

Order Card

Send me:

____ MacCats, $9.95, 18143
____ Multiplan Mastery on the Macintosh, $15.95, 18182
____ Inside the Macintosh, $16.95
____ Microsoft BASIC Programming for the Mac, $12.95, 18167

☐ Check here for a free catalog
To order, contact your local bookstore or computer store, or send the order card to

Scott, Foresman and Company
Professional Publishing Group
1900 East Lake Avenue
Glenview, IL 60025

In Canada, contact
Macmillan of Canada
164 Commander Blvd.
Agincourt, Ontario
MIS 3C7

Please check method of payment:

☐ MasterCard  ☐ Visa  ☐ Check/Money Order

Amount Enclosed $____________________
Credit Card No. ____________________________________________
Expiration date ___________________________________________
Signature ________________________________________________

Name (please print) __________________________________________
Address ___________________________________________________
City ___________________________ State ________ Zip ____________

Add applicable sales tax, plus 6% of Total for U.P.S. (Publisher pays regular book rate).

Full payment must accompany your order. Offer good in U.S. only.

A18110
"Mac Power offers a wealth of practical information on Macintosh software, along with a variety of creative examples. I was particularly impressed by Allen Munro's thorough knowledge and accurate coverage of Filevision software." — Bob Foster, Vice President, Marketing, Telos Software Products

This handy guide introduces you to eleven of the most popular Macintosh software packages currently available, and gives you tips on using each package productively.

You'll learn how to
- produce your own graphics with MacPaint and MacDraw
- Polish your word processing skills with MacWrite and Microsoft Word
- create and edit spreadsheets with Multiplan
- produce high-quality business graphics with Microsoft's Chart
- create and manipulate outlines with ThinkTank
- organize your files with Filevision
- communicate with other computers using MacTerminal

Mac Power includes many easy-to-follow examples, some shortcuts, and over 250 illustrations created by the author using the Macintosh. Munro also offers a helpful appendix on mouse skills for new users, and a brief guide to MacPascal and Mac peripherals. Whether you use your Macintosh at home or at work, this book will help you tap its full potential.

Allen Munro is currently Assistant Director of the Behavioral Technology Laboratories at the University of Southern California and a freelance writer specializing in personal computing topics. A resident of Manhattan Beach, California, Dr. Munro has had articles appear in such magazines as Byte, Creative Computing, Apple Orchard, Computer Dealer, InCider, InfoWorld, and Softalk.