IBM®

SuperValue

OS/2 Warp uncensored

Productivity Pack on CD

Straight from the IBM Development Team: the Reusable REXX Scripts You've Been Asking for to Rebuild Your Workplace Shell® Objects and Configure Your System Like the Developers Do!

OS/2® Warp uncensored

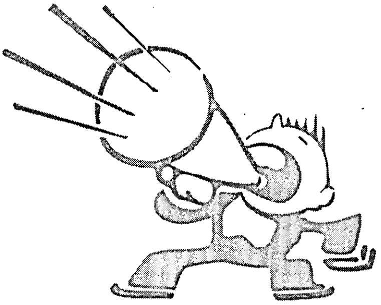
New!

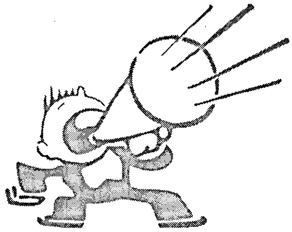
- Increase Your **Productivity** with Expert OS/2 Warp Techniques from the Programmers Who Wrote the OS/2 Workplace Shell
- **Customize** Your Desktop with New Bit Maps and Icons
- **Streamline** Your Work with CONFIG.SYS Tips from IBM Insiders

Peter G. Magid & Ira H. Schneider
Lead Developers of the OS/2 Workplace Shell

IDG BOOKS WORLDWIDE

OS/2® Warp Uncensored





OS/2® Warp Uncensored

by Peter G. Magid & Ira H. Schneider

An IBM Press Book

Published by IDG Books Worldwide, Inc.
An International Data Group Company

Foster City, CA • Chicago, IL • Indianapolis, IN • Braintree, MA • Dallas, TX

OS/2® Warp Uncensored

An IBM Press Book

Published by
IDG Books Worldwide, Inc.
An International Data Group Company
919 E. Hillsdale Blvd.
Suite 400
Foster City, CA 94404

Library of Congress Catalog Card No.: 95-77672

ISBN: 1-56884-474-3

Printed in the United States of America

10 9 8 7 6 5 4 3 2 1

1L/RY/QX/ZV

Distributed in the United States by IDG Books Worldwide, Inc.

Distributed by Macmillan Canada for Canada; by Computer and Technical Books for the Caribbean Basin; by Contemporanea de Ediciones for Venezuela; by Distribuidora Cuspide for Argentina; by CITEC for Brazil; by Ediciones ZETA S.C.R. Ltda. for Peru; by Editorial Limusa SA for Mexico; by Transworld Publishers Limited in the United Kingdom and Europe; by Al-Maiman Publishers & Distributors for Saudi Arabia; by Simron Pty. Ltd. for South Africa; by IDG Communications (HK) Ltd. for Hong Kong; by Toppan Company Ltd. for Japan; by Addison Wesley Publishing Company for Korea; by Longman Singapore Publishers Ltd. for Singapore, Malaysia, Thailand, and Indonesia; by Unalis Corporation for Taiwan; by WS Computer Publishing Company, Inc. for the Philippines; by WoodsLane Pty. Ltd. for Australia; by WoodsLane Enterprises Ltd. for New Zealand.

For general information on IDG Books Worldwide's books in the U.S., please call our Consumer Customer Service department at 800-762-2974. For reseller information, including discounts and premium sales, please call our Reseller Customer Service department at 800-434-3422.

For information on where to purchase IDG Books Worldwide's books outside the U.S., contact IDG Books Worldwide at 415-655-3021 or fax 415-655-3295.

For information on translations, contact Marc Jeffrey Mikulich, Director, Foreign & Subsidiary Rights, at IDG Books Worldwide, 415-655-3018 or fax 415-655-3295.

For sales inquiries and special prices for bulk quantities, write to the address above or call IDG Books Worldwide at 415-655-3200.

For information on using IDG Books Worldwide's books in the classroom, or ordering examination copies, contact Jim Kelly at 800-434-2086.

For authorization to photocopy items for corporate, personal, or educational use, please contact Copyright Clearance Center, 222 Rosewood Drive, Danvers, MA 01923, or fax 508-750-4470.

Limit of Liability/Disclaimer of Warranty: This book is sold "AS-IS", without warranty of any kind, either express or implied, regarding the contents of this book, including but not limited to any implied warranties for the book's quality, performance, merchantability, or fitness for any particular purpose. Neither IDG Books Worldwide, Inc., nor IBM Corporation, nor their dealers or distributors shall be liable to the purchasers or any other person or entity with respect to any liability, loss, or damage caused or alleged to have been caused directly or indirectly by reliance upon the contents of this book.

System Requirements: IBM PC or compatible running OS/2 Warp.

Trademarks: All brand names and product names used in this book are trademarks, registered trademarks, or trade names of their respective holders. IBM and OS/2 are registered trademarks of International Business Machines Corporation. IBM Press is a trademark of IBM Corporation and is used by IDG under license.

IDG BOOKS WORLDWIDE

This logo and *Leading the Knowledge Revolution* are trademarks under exclusive license to IDG Books Worldwide, Inc., from International Data Group, Inc.

About the Authors

Peter G. Magid is an Advisory Programmer on the Workplace Shell Development Team. As the User Interface Design Lead for the OS/2 Workplace Shell, Peter has had chief design responsibilities for what users see and how they interact with the Workplace Shell. Peter also plays an active role in reviewing and consulting on the content and direction of the Workplace Shell programming model, and he developed the code for the LaunchPad.

Peter holds two patents, one for the use of Shadow objects and the other for the new OS/2 Warp Pickup and Drop feature. He has also received an Outstanding Innovation Award from IBM in appreciation of his work on the OS/2 Warp Workplace Shell. Peter has been with IBM since 1984.

Peter's publication credits include the proceedings of a conference on "Using and Customizing the Workplace Shell" and an article in *The IBM Developer's Connection newsletter* on the object life cycle of a Workplace Shell object.

Ira H. Schneider is an Advisory Programmer on the Workplace Shell Development Team. He has been instrumental in developing OS/2 Warp, including performance improvements and functional enhancements to the Workplace Shell. He has also worked on the OS/2 Swapper and on OS/2 System Initialization, and he was Lead Architect for OS/2 1.2. Ira has been with IBM since 1969.

Ira holds two patents, one for the Personal Computer Attachment For Host System Display Stations and the other for an Analog Video Server. He has also received an Outstanding Contribution Award and a Division Award from IBM in appreciation of his past work.

Ira's publication credits include four *IBM Technical Disclosure Bulletin* articles and two *IBM Technical Reports.*

Cathy Bloomfield is a Senior Associate Programmer at IBM Boca Raton. She has been with the OS/2 Development Team since 1989, working on the Printer Manager, Spooler, Print Object, Network Object, and Workplace Shell.

Doris Corel is a Systems Engineer with IBM Austria, on assignment in Boca Raton. She provides widespread technical information for OS/2 in the International Support Organization. She has worked with OS/2 since 1987, originally providing support to insurance, banking, and public sector customers, and later leading projects for implementing OS/2 in large LAN environments.

John M. Hare is a Senior Associate Programmer at IBM Boca Raton. He has been with the OS/2 Development Team since 1985, working on DOS emulation and WIN-OS/2 support. He has written code for many OS/2 Warp features, including DOS settings categories, printing and loading of the DOS settings, displaying of the OEM DOS settings, and TIMER.DRV divisor value settings in SYSTEM.INI.

Salil Kulkarni is a Staff Programmer and Technical Lead for the OS/2 Installation Department. He has been with IBM since 1989, and has worked on all versions of OS/2 System Installation from OS/2 Version 1.2 through OS/2 Warp. Salil also worked on the improved diskette compression techniques used since OS/2 2.1.

Paul Rogers is an Advisory Programmer with IBM Boca Raton. Paul has been involved in OS/2 Multimedia development since 1991. He has been with IBM since 1979, and has worked on OS/2 development projects since 1986. Paul worked on the keyboard device driver for OS/2 Version 1.0.

ABOUT IBM PRESS

IBM Press is committed to bringing you high-quality books about IBM products and technologies. You can count on IBM Press to provide books that are of interest to all computer users — home and office users, network and systems administrators, and application developers and programmers.

It is my pleasure to present the third in a series of IBM Press books — *OS/2 Warp Uncensored*. I am especially pleased to introduce this book as it is the first power user's guide for OS/2 Warp that is authorized by IBM.

Inside these pages, readers will learn about the benefits that OS/2 Warp brings to the home, office, or mobile computer user. Written in plain English, this power user's guide provides complete descriptions of how to exploit OS/2 Warp. I hope you will find this book to be the perfect accompaniment to OS/2 Warp.

Enjoy this IBM Press book.

Beth Tripi
Imprint Manager
International Business Machines Corporation

ABOUT IDG BOOKS WORLDWIDE

Welcome to the world of IDG Books Worldwide.

IDG Books Worldwide, Inc., is a subsidiary of International Data Group, the world's largest publisher of computer-related information and the leading global provider of information services on information technology. IDG was founded more than 25 years ago and now employs more than 7,200 people worldwide. IDG publishes more than 233 computer publications in 65 countries (see listing below). More than 60 million people read one or more IDG publications each month.

Launched in 1990, IDG Books Worldwide is today the #1 publisher of best-selling computer books in the United States. We are proud to have received 3 awards from the Computer Press Association in recognition of editorial excellence, and our best-selling *...For Dummies*™ series has more than 12 million copies in print with translations in 25 languages. IDG Books, through a recent joint venture with IDG's Hi-Tech Beijing, became the first U.S. publisher to publish a computer book in the People's Republic of China. In record time, IDG Books has become the first choice for millions of readers around the world who want to learn how to better manage their businesses.

Our mission is simple: Every IDG and IBM Press book is designed to bring extra value and skill-building instructions to the reader. Our books are written by experts who understand and care about our readers. The knowledge base of our editorial staff comes from years of experience in publishing, education, and journalism — experience which we use to produce books for the '90s. In short, we care about books, so we attract the best people. We devote special attention to details such as audience, interior design, use of icons, and illustrations. And because we use an efficient process of authoring, editing, and desktop publishing our books electronically, we can spend more time ensuring superior content and spend less time on the technicalities of making books.

You can count on our commitment to deliver high-quality books at competitive prices on topics consumers want to read about. At IDG, we value quality, and we have been delivering quality for more than 25 years. You'll find no better book on a subject than an IDG book.

John J. Kilcullen

John Kilcullen
President and CEO
IDG Books Worldwide, Inc.

VIII
WINNER
Eighth Annual
Computer Press
Awards 1992

IX
WINNER
Ninth Annual
Computer Press
Awards 1993

IDG BOOKS WORLDWIDE

IDG Books Worldwide, Inc., is a subsidiary of International Data Group, the world's largest publisher of computer-related information and the leading global provider of information services on information technology. International Data Group publishes over 220 computer publications in 65 countries. More than fifty million people read one or more International Data Group publications each month. The officers are Patrick J. McGovern, Founder and Board Chairman; Kelly Conlin, President; Jim Casella, Chief Operating Officer. International Data Group's publications include: **ARGENTINA'S** Computerworld Argentina, Infoworld Argentina; **AUSTRALIA'S** Computerworld Australia, Computer Living, Australian PC World, Australian Macworld, Network World, Mobile Business Australia, Publish!, Reseller, IDG Sources; **AUSTRIA'S** Computerwelt Oesterreich, PC Test; **BELGIUM'S** Data News (CW); **BOLIVIA'S** Computerworld; **BRAZIL'S** Computerworld, Connections, Game Power, Mundo Unix, PC World, Publish, Super Game; **BULGARIA'S** Computerworld Bulgaria, PC & Mac World Bulgaria, Network World Bulgaria; **CANADA'S** CIO Canada, Computerworld Canada, InfoCanada, Network World Canada, Reseller; **CHILE'S** Computerworld Chile, Informatica; **COLOMBIA'S** Computerworld Colombia, PC World; **COSTA RICA'S** PC World; **CZECH REPUBLIC'S** Computerworld, Elektronika, PC World; **DENMARK'S** Communications World, Computerworld Danmark, Computerworld Focus, Macintosh Produktkatalog, Macworld Danmark, PC World Danmark, PC Produktguide, Tech World, Windows World; **ECUADOR'S** PC World Ecuador; **EGYPT'S** Computerworld (CW) Middle East, PC World Middle East; **FINLAND'S** MikroPC, Tietoviikko, Tietoverkko; **FRANCE'S** Distributique, GOLDEN MAC, InfoPC, Le Guide du Monde Informatique, Le Monde Informatique, Telecoms & Reseaux; **GERMANY'S** Computerwoche, Computerwoche Focus, Computerwoche Extra, Electronic Entertainment, Gamepro, Information Management, Macwelt, Netzwelt, PC Welt, Publish, Publish; **GREECE'S** Publish & Macworld; **HONG KONG'S** Computerworld Hong Kong, PC World Hong Kong; **HUNGARY'S** Computerworld SZT, PC World; **INDIA'S** Computers & Communications; **INDONESIA'S** Info Komputer; **IRELAND'S** ComputerScope; **ISRAEL'S** Beyond Windows, Computerworld Israel, Multimedia, PC World Israel; **ITALY'S** Computerworld Italia, Lotus Magazine, Macworld Italia, Networking Italia, PC Shopping Italy, PC World Italia; **JAPAN'S** Computerworld Today, Information Systems World, Macworld Japan, Nikkei Personal Computing, SunWorld Japan, Windows World; **KENYA'S** East African Computer News; **KOREA'S** Computerworld Korea, Macworld Korea, PC World Korea; **LATIN AMERICA'S** GamePro; **MALAYSIA'S** Computerworld Malaysia, PC World Malaysia; **MEXICO'S** Compu Edicion, Compu Manufactura, Computacion/Punto de Venta, Computerworld Mexico, MacWorld, Mundo Unix, PC World, Windows; **THE NETHERLANDS'** Computer! Totaal, Computable (CW), LAN Magazine, Lotus Magazine, MacWorld; **NEW ZEALAND'S** Computer Buyer, Computerworld New Zealand, Network World, New Zealand PC World; **NIGERIA'S** PC World Africa; **NORWAY'S** Computerworld Norge, Lotusworld Norge, Macworld Norge, Maxi Data, Networld, PC World Ekspress, PC World Nettverk, PC World Norge, PC World's Produktguide, Publish& Multimedia World, Student Data, Unix World, Windowsworld; **PAKISTAN'S** PC World Pakistan; **PANAMA'S** PC World Panama; **PERU'S** Computerworld Peru, PC World; **PEOPLE'S REPUBLIC OF CHINA'S** China Computerworld, China Infoworld, China PC Info Magazine, Computer Fan, PC World China, Electronics International, Electronics Today/Multimedia World, Electronic Product World, China Network World, Software World Magazine, Telecom Product World; **PHILIPPINES'** Computerworld Philippines, PC Digest (PCW); **POLAND'S** Computerworld Poland, Computerworld Special Report, Networld, PC World/Komputer, Sunworld; **PORTUGAL'S** Cerebro/PC World, Correio Informatico/Computerworld, MacIn; **ROMANIA'S** Computerworld, PC World, Telecom Romania; **RUSSIA'S** Computerworld-Moscow, Mir - PK (PCW), Sety (Networks); **SINGAPORE'S** Computerworld Southeast Asia, PC World Singapore; **SLOVENIA'S** Monitor Magazine; **SOUTH AFRICA'S** Computer Mail (CIO),Computing S.A.,Network World S.A., Software World; **SPAIN'S** Advanced Systems, Amiga World, Computerworld Espana, Communicaciones World, Macworld Espana, NeXTWORLD, Super Juegos Magazine (GamePro), PC World Espana, Publish; **SWEDEN'S** Attack, ComputerSweden, Corporate Computing, Macworld, Mikrodatorn, Natverk & Kommunikation, PC World, CAP & Design, Datalngenjoren, Maxi Data,Windows World; **SWITZERLAND'S** Computerworld Schweiz, Macworld Schweiz, PC Tip; **TAIWAN'S** Computerworld Taiwan, PC World Taiwan; **THAILAND'S** Thai Computerworld; **TURKEY'S** Computerworld Monitor, Macworld Turkiye, PC World Turkiye; **UKRAINE'S** Computerworld, Computers+Software Magazine; **UNITED KINGDOM'S** Computing /Computerworld, Connexion/Network World, Lotus Magazine, Macworld, Open Computing/Sunworld; **UNITED STATES'** Advanced Systems, AmigaWorld, Cable in the Classroom, CD Review, CIO, Computerworld, Computerworld Client/Server Journal, Digital Video, DOS World, Electronic Entertainment Magazine (E2), Federal Computer Week, Game Hits, GamePro, IDG Books, Infoworld, Laser Event, Macworld, Maximize, Multimedia World, Network World, PC Letter, PC World, Publish, SWATPro, Video Event; **URUGUAY'S** PC World Uruguay; **VENEZUELA'S** Computerworld Venezuela, PC World; **VIETNAM'S** PC World Vietnam. 05/11/95

Dedication

Peter G. Magid

To my wife Heidi, who is the love of my life and the wind beneath my wings. Also to my daughter Hannah and my son Daniel, who fill my life with joy.

Ira H. Schneider

To my wife Lorraine, who inspires me to fly like an eagle and encourages me when I'm deep in despair; and to my cats Natasha and Cleopatra who think I make a purr-fect pillow.

Acknowledgments

The authors would like to thank the people at IBM Press and IDG Books for all their hard work and encouragement in making this book possible, especially: Liz Jean, Amy Marks, Eric Stone, Beth Tripi, Diane Walewski, and Carolyn Welch.

There are many hundreds of people responsible for developing, delivering, marketing, servicing and supporting OS/2 Warp. If you want to see a list of the people who worked directly on developing OS/2 Warp, click on empty space on the OS/2 Warp Desktop and hold the Ctrl, Alt, and Shift keys while pressing the letter **O**. When you are finished looking at the list, press any key to return to the Desktop.

OS/2 Warp could not have achieved the level of quality and success that it has without all of their hard work. But the authors would like to express their particular gratitude to the past and current members of the Workplace Shell Development team for giving our users the best user interface there is for a major operating system:

> Chris Andrew, Bernabe J. Arruza, Alan B. Auerbach, Thomas A. Bellwood, Paul W. Bennett, Cathy A. Bloomfield, Marc Bloomfield, Lewis W. Boehm, Jeff Brazier, Lori T. Brown, Claudette Caldwell, Roberta P. Carey, Deepa S. Desai, Rick M. Efruss, Mark Ericson, Brad Fraley, Jeffrey A. Jones, Sheila E. Harnett, Eric E. Osmann, Mindy Pollack, Joseph C. Polimeni, Aaron Reed, Merle D. Sterling, James L. Taylor, Andrew Violette, Michael J. Walker

We would also like to thank the people who directly contributed to the contents of this book. They are Cathy Bloomfield, Doris Corel, John Hare, Salil Kulkarni, and Paul Rogers, who wrote chapters; Alan Auerbach, who wrote the Object Tool and gathered the rest of the contents of the CD-ROM; Brian Black, who provided some of the pointer sets on the CD-ROM; Michael Harris, who provided many of the icons on the CD-ROM; Bob Klemetsmo, who converted photographs to bit maps used in some of the screen images; and Kevin McMahon, who provided many of the bit maps on the CD-ROM.

We wish to especially thank Lori Brown and James Taylor for their guidance, leadership, and inspiration during the development of the Workplace Shell for OS/2 Warp. They forever have our admiration and gratitude.

(The Publisher would like to give special thanks to Patrick J. McGovern, without whom this book would not have been possible.)

Credits

IBM Press Staff

Imprint Manager
Beth Tripi

Acquisitions Manager
Diane Walewski

Managing Editor
Liz Jean

Cover Designer
Jeffrey Lewis

IDG Books Worldwide Staff

Vice President and Group Publisher
Brenda McLaughlin

Associate Publisher
Eric Stone

Brand Manager
Pradeepa Siva

Associate Acquisitions Editor
Anne Hamilton

Associate Developmental Editor
Amy Marks

Copy Editors
Carolyn Welch
Bill Cassell

Assistant Editor
Heather Albright

Production Director
Beth Jenkins

Supervisor of Project Coordination
Cindy L. Phipps

Pre-Press Coordinator
Steve Peake

Associate Pre-Press Coordinator
Tony Augsburger

Media/Archive Coordinator
Paul Belcastro

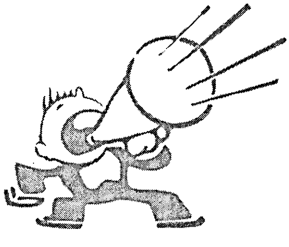
Associate Project Coordinator
Sherry Gomoll

Production Staff
Gina Scott
Carla C. Radzikinas
Patricia R. Reynolds
Melissa Buddendeck
Dwight Ramsey
Rob Springer
Theresa Sánchez-Baker
Kathie Schnorr
Linda M. Boyer
Chris Collins
Maridee Ennis
Angela Hunckler
Mark Owens
Anna Rohrer

Proofreader
Henry Lazarek

Indexer
Liz Cunningham

Book Design
Amy Peppler Adams
designLab, Seattle



Contents at a Glance

Foreword xxxiii

Introduction 1

Chapter 1 Installing OS/2 Warp 5

Chapter 2 Workplace Shell Metaphor 41

Chapter 3 Object Management 89

Chapter 4 The LaunchPad 101

Chapter 5 Workplace Shell Object Types 119

Chapter 6 Customizing the Appearance of Windows 203

Chapter 7 Customizing the Workplace Shell 255

Chapter 8 Using REXX Utilities 321

Chapter 9 OS/2 System Objects 395

Chapter 10 CONFIG.SYS Options Affecting the Workplace Shell 441

Chapter 11 Boot Time Services 465

Chapter 12 The BonusPak Applications 479

Chapter 13 Multimedia 525

Chapter 14 Printing 567

Chapter 15 Command Line Interface 599

Chapter 16 DOS and WIN-OS/2 Settings 691

Appendix A CD-ROM Contents 731

Appendix B Developer's Desktop 743

Appendix C System Keyboard Definitions 763

Index 769

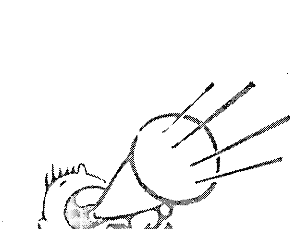


Table of Contents

Foreword xxxiii
Introduction 1
Conventions Used 1
What's on The CD-ROM 2
A Brief History of OS/2 2
Chapter 1: Installing OS/2 Warp 5
Minimal Hardware Requirements 5
Recommended Hardware Support 6
Which Version of OS/2 Warp Should I Buy? 7
Choosing the Right Installation Media 7
Preparing for Installation 8
Installing OS/2 Warp on large partitions 8
Installing OS/2 Warp on compressed drives 9
Installing OS/2 Warp using a third-party device 10
Installing a fix for a file on diskette 1 10
Phase 1 of OS/2 Warp Installation 11
Using the Easy Installation option 11
Using the Advanced Installation option 13
Installing Boot Manager 13
Creating other partitions 15
Formatting the installation partition 16
Informational messages during phase 1 17
Dual boot support 18
Completing phase 1 18
Phase 2 of OS/2 Warp Installation 18
Country information and hardware configuration 18
Installing and configuring printers 20
Installing and configuring multimedia hardware .. 21
Installing and configuring other hardware 22
Software selections 23
Installing documentation 24
Installing fonts 24

Installing optional system utilities 25

Installing tools and games 26

Installing OS/2 DOS support 26

Installing OS/2 WIN-OS/2 support 26

Installing multimedia software support 28

Installing high performance file system 28

Installing serviceability and diagnostic aids 29

Installing optional bitmaps 29

Formatting drives other than the installation partition ... 29

Bringing up a command prompt 29

Configuring software 30

Using advanced options 30

Using the OS/2 Maintenance Partition 32

Response Files and Installing over the Network 32

Installing Features Selectively 33

Uninstalling Features 33

Using the Device Driver Installation Program 34

Adding Programs to Your Desktop after Installation 35

Creating Utility Diskettes 35

Troubleshooting 36

Chapter 2: Workplace Shell Metaphor 41

Why the Workplace Shell? 42

What is the Workplace Shell model? 44

Pop-up menus 44

Templates 45

Graphical User Interface 46

Hidden Programs 46

Associations between Data Objects and Programs 47

Default associations 48

Establishing and changing associations 49

Settings Notebooks 51

System component Settings Notebooks 53

Standard Objects and User Interface Controls 53

Folders 53

The Shredder 56

Printer objects 56

The LaunchPad 56
Color, font, and scheme palettes 56
Help 57
The Getting Started tutorial 57
Templates 58
Pop-up menus 58
Characteristics of Standard Objects 59
Direct manipulation of objects 59
Views of objects 59
Shadow objects 62
Standard visual and interaction paradigms 63
Putting the Workplace Shell Model to Work 67
Objects in the Workplace Shell 67
Folders in the Workplace Shell 69
Templates in the Workplace Shell 76
Associations in the Workplace Shell 77
Program object parameters in the Workplace Shell 82
Summary 87
Chapter 3: Object Management 89
Selecting Objects 89
Using point selection 90
Using area (marquee) selection 90
Using range swipe (ordered) selection 91
Using range swipe (unordered) selection 92
Using range click selection 92
Using add mode selection 92
Understanding Drag Concepts 93
Introducing Pickup and Drop 94
Using pickup 95
Canceling pickup 95
Dropping from a pickup 96
Object Management Functions 96
Copying and moving objects 96
Shadowing objects 97
Printing objects 97
Deleting objects 97

Creating objects 97
Naming objects 98
Managing disks 98
Starting programs with an object 99
Confirmation options 99
Summary 99
Chapter 4: The LaunchPad 101
Working with the LaunchPad 101
Fast and easy access 102
Action Button area 102
Object Button area 103
Keyboard access 108
Customizing the LaunchPad 108
Setting LaunchPad options 108
Adding and removing objects 113
Dropping colors and fonts on the LaunchPad 115
Special Considerations 116
Disabling the LaunchPad 116
Copying the LaunchPad 116
Green question marks 116
Summary 117
Chapter 5: Workplace Shell Object Types 119
Workplace Shell Objects 119
WPObject—The Parent of All Workplace Shell Object Classes .. 121
WPObject — properties 121
WPObject — behaviors 125
WPObject — pop-up menu items 127
WPObject — Settings Notebook pages 134
The Base Classes — WPAbstract, WPFileSystem, and WPTransient 139
WPAbstract—the base class that lives in the OS2.INI file .. 140
WPFileSystem — the base class for file system objects .. 141
WPTransient — the base class for temporary objects ... 149

Abstract Objects 151
WPDisk — the Disk object 153
WPPalette — the Palette object class 163
WPShadow — the Shadow object 165
WPShredder — the Shredder object 167
WPSpecial Needs — the Special Needs object 167
Data File Objects 168
WPDataFile — the Data File object 169
Other data file classes 170
Folder Objects 172
WPFolder — the Folder object 172
WPDesktop — the Desktop Folder object 187
WPDrives — the Drives Folder object 194
WPMinWinViewer — the Minimized Window Viewer Folder object 194
WPNetGrp — the Network Group Folder object 195
WPNetwork — the Network Folder object 195
WPProgramGroup — the Program Group Folder object 195
WPRootFolder — the Root Folder object 195
WPServer — the File Server Folder object 196
WPSharedDir — the Shared Directory Folder object . . . 196
WPStartup — the Startup Folder object 196
WPTemplates — the Templates Folder object 197
Transient Objects 198
WPJob — the Job object 198
WPMinWindow — the Minimized Window object . . . 198
WPPdr — the Printer Driver object 200
WPPort — the Printer Port object 200
WPQdr — the Print Queue Driver object 200
Summary 200
Chapter 6: Customizing the Appearance of Windows 203
What You Can Customize 203
Colors 206
Fonts 220
Folder backgrounds 223

How to Customize Colors, Fonts, and Folder Backgrounds 223
The Color Palette 223
The Font Palette 230
The Folder Settings Notebook 235
The Scheme Palette 246
Summary 253
Chapter 7: Customizing the Workplace Shell 255
The Mouse Object 256
Timing page 256
Setup page 258
Mappings page 259
Pointers page 261
Comet Cursor 266
The Keyboard Object 269
Timing page 269
Mappings page 271
Special Needs page 272
The System Object 274
Screen pages 274
Confirmations page 276
Title page 279
Window pages 280
Input page 284
Print Screen page 285
Logo page 285
The Sound Object 287
Sound page 287
Warning Beep page 290
The Spooler Object 291
Spool Path page 291
Print Priority page 292
Pop-up menu items 292
The WIN-OS/2 Setup Object 293
3.1 Session page 293
Data Exchange page 297

The Country Object 299
Country page 300
Time page 300
Date page 301
Numbers page 302
The System Clock Object 304
Date/Time page 304
View pages 305
Alarm page 309
The Power Object 310
Power page 311
View page 312
Suspend and resume 314
The Plug and Play for PCMCIA Object 315
Summary 319
Chapter 8: Using REXX Utilities 321
Loading the REXX Utilities 322
REXX Utility Functions 322
User Interface Functions 325
RxMessageBox 325
SysCls 327
SysCurPos 328
SysCurState 329
SysGetKey 329
SysGetMessage 330
SysSleep 332
SysTextScreenRead 332
SysTextScreenSize 333
Workplace Shell Functions 334
Referencing Workplace Shell Objects 334
Setup Strings 339
SysCopyObject 357
SysCreateObject 358
SysCreateShadow 360
SysDeregisterObjectClass 361
SysDestroyObject 361

SysMoveObject 362
SysOpenObject 363
SysQueryClassList 364
SysRegisterObjectClass 365
SysSaveObject 366
SysSetObjectData 367
File Access Functions 368
SysDriveInfo 368
SysDriveMap 369
SysFileDelete 370
SysFileSearch 371
SysFileTree 373
SysGetEA 377
SysIni 380
SysMkDir 383
SysPutEA 384
SysRmDir 386
SysSearchPath 387
SysSetIcon 387
SysTempFileName 389
SysWaitNamedPipe 390
Miscellaneous Functions 391
SysDropFuncs 391
SysOS2Ver 391
Summary 392
Chapter 9: OS/2 System Objects 395
Objects in the OS/2 System Folder 395
System Setup folder 396
Command Prompts folder 397
Drives folder 407
Productivity folder 407
Games folder 425
Startup folder 427
Shredder 428
Minimized Window Viewer 428

Objects in the Information Folder 428

Master Help Index 428

Glossary 430

Tutorial 430

README 430

Books 431

Objects in the Network Folder 431

The Network Group folder object 432

The Server folder object 434

The Shared Directory folder object 435

The Remote Printer object 438

The Network Shadow object 438

The Nowhere folder 438

Summary 440

Chapter 10: CONFIG.SYS Options Affecting the Workplace Shell 441

Order of Processing of CONFIG.SYS Statements 442

Workplace Shell Related Statements 444

Starting the user interface 444

PROTSHELL= statement 445

SET RUNWORKPLACE= statement 446

Workplace Shell Configuration Files 447

SET SYSTEM_INI= statement 448

SET USER_INI= statement 448

Starting the Workplace Shell 448

SET DESKTOP= statement 448

SET AUTOSTART= statement 449

SET RESTARTOBJECTS= statement 450

SET SHELLEXCEPTIONHANDLER= statement 452

SET AUTOREFRESHFOLDERS= statement .. 452

SET OBJECTSNOOZETIME= statement 453

SET DEFERICONLOAD= statement 454

SET MENUSTYLE= statement 454

Setting the global environment 455

SET OS2_SHELL= statement 455

LIBPATH= statement 456

SET PATH= and SET DPATH= statements...... 457
SET TMP= statement.. 458
SET HELP= statement 458
SET GLOSSARY= statement............................ 459
SET BOOKSHELF= statement........................ 459
Starting the DSOM environment 459
Starting the DSOM daemon process 460
Starting the Workplace Shell DSOM server 460
Running Programs 461
RUN= statement 461
CALL= statement 462
Summary 463
Chapter 11: Boot Time Services 465
Creating a Boot Manager 465
Boot Manager overview 466
Boot Manager example 466
Creating the Boot Manager partition 467
Creating the DOS partition.............................. 468
Configuring the Boot Manager menu 470
Setting the C: Partition.................................... 472
Using the Archive and Recover Feature 473
Accessing an OS/2 Command Line before the system is booted 474
Resetting your display back to VGA mode 474
Restarting the system at the Maintenance Desktop 475
Recovering your archived Desktops 475
Archiving your Desktop directory 476
Creating unique CONFIG.SYS files for different boot environments 477
Summary 478
Chapter 12: The BonusPak Applications 479
How to Install the BonusPak 481
IBM Works 482
IBM Works word processor 483
IBM Works spreadsheet 485
IBM Works database and report writer 487

Personal Information Manager (PIM) 488
Phone/Address Book 489
Appointments 490
Notepad 493
Planner 493
To-Do List 494
Year Calendar 494
Event Monitor 495
FaxWorks for OS/2 495
The Information Superhighway 499
What is this Information Superhighway? 500
IBM Internet Connection 500
HyperACCESS Lite for OS/2 518
IBM Person to Person 519
VideoIN 521
Multimedia Viewer 521
System Information Tool 522
Summary 524
Chapter 13: Multimedia 525
The Challenge of Delivering Multimedia Data 525
A brief history of OS/2 Multimedia 527
OS/2 Warp: A solid platform for robust Multimedia . . . 529
A Tour of OS/2 Warp Multimedia 530
Multimedia folder 533
But, wait, there's more: OS/2 Multimedia in the BonusPak . . 548
On-line information 556
At Your CMD: Multimedia in REXX 557
PLAY.CMD and RECORD.CMD 557
Programming REXX Multimedia 560
Fun Stuff Out There 564
OS/2 games 564
Additional multimedia support and applications 565
Introduction to the Printing Subsystem 567

Chapter 14: Printing 567
Creating a Printer Object 568
Dragging a Printer template from the Templates folder . 568
Other ways to create a Printer object 571
Creating a Network Printer object 572
Using the Printer Object 572
Selecting print job options 574
Printing a job 577
Understanding print job types 578
Using the Printer Object's Settings Notebook 579
Selecting local printer options 579
Selecting network printer options 592
Advanced Topics 594
Printer data flow 594
Printer sharing 595
Printer pooling 595
Redirection 596
New polling support 597
Summary 598
Chapter 15: Command Line Interface 599
File access 600
Partitions and drives 600
File systems 600
Path and filenames. 600
Extended attributes. 603
Command symbols 603
Grouping symbols 604
& 604
&& 604
|| 604
( ) 604
Piping and redirection 605
| 605
>, >>, <, 2>, 2>&1 605

Taking characters literally 606
^ 606
“ ” 607
Executing commands silently 607
Environment variable substitution 607
Commands 608
Internal commands 608
CALL 608
CD, CHDIR 609
CHCP 610
CLS 611
COPY 611
DATE 615
DEL, ERASE 615
DETACH 617
DIR 617
DPATH 621
ECHO 621
ENDLOCAL 621
EXIT 622
EXTPROC 622
FOR 622
GOTO 623
IF 624
KEYS 625
MD, MKDIR 627
MOVE 627
PATH 628
PAUSE 629
PROMPT 629
RD, RMDIR 638
REM 638
REN, RENAME 638
SET 639
SETLOCAL 639
SHIFT 640

START 640
TIME 643
TYPE 643
VER 643
VERIFY 643
VOL 644
External commands 644
ANSI 644
ATTRIB 644
BACKUP 646
BLDLEVEL 647
BOOT 648
CACHE 648
CHKDSK 649
CLIPOS2 650
CMD 651
COMP 652
DISKCOMP 653
DISKCOPY 653
E 654
EAUTIL 655
FDISK 657
FDISKPM 659
FIND 660
FORMAT 661
HELP 663
ICONEDIT 664
KEYB 666
LABEL 668
MAKEINI 668
MODE 669
MORE 672
PATCH 672
PMCHKDSK 672
PMFORMAT 673
PMREXX 674

PRINT 674
PSTAT 675
RECOVER 676
RESTORE 678
REXXTRY 680
SETBOOT 680
SORT 681
SPOOL 681
SYSLEVEL 681
SYSLOG 682
TEDIT 682
TREE 682
TUTORIAL 682
UNDELETE 683
UNPACK 683
VIEW 684
WPDSINIT 685
XCOPY 685
XDFCOPY 688
Summary 688
Chapter 16: DOS and WIN-OS/2 Settings 691
Overview Of DOS and Windows Support 692
DOS or WIN-OS/2 Session Settings 694
DOS or WIN-OS/2 Settings Categories 695
DOS keyboard settings 696
KBD_ALTHOME_BYPASS 696
KBD_BUFFER_EXTEND 697
KBD_CTRL_BYPASS 697
KBD_RATE_LOCK 697
DOS memory settings 698
DOS_HIGH 698
DOS_RMSIZE 698
DOS_UMB 699
DPMI_DOS_API 699
DPMI_MEMORY_LIMIT 699
DPMI_NETWORK_BUFF_SIZE 699

EMS_FRAME_LOCATION 699
EMS_HIGH_OS_MAP_REGION 700
EMS_LOW OS_MAP_REGION 700
EMS_MEMORY_LIMIT 700
MEM_EXCLUDE_REGIONS......................... 700
MEM_INCLUDE_REGIONS 701
XMS_HANDLES 701
XMS_MEMORY_LIMIT 701
XMS_MINIMUM_HMA................................ 701
DOS mouse and touch screen settings 702
MOUSE_EXCLUSIVE_ACCESS 702
TOUCH_EXCLUSIVE_ACCESS 702
DOS printer settings 703
PRINT_SEPARATE_OUTPUT 703
PRINT_TIMEOUT 703
VIDEO_FASTPASTE.................................... 704
DOS video settings 704
VIDEO_MODE_RESTRICTION 705
VIDEO_ONDEMAND_MEMORY................ 705
VIDEO_RETRACE_EMULATION................ 705
VIDEO_ROM_EMULATION 705
VIDEO_SWITCH_NOTIFICATION............. 705
VIDEO_WINDOW_REFRESH 706
VIDEO_8514A_XGA_IOTRAP 706
Other DOS settings 706
AUDIO_ADAPTER_SHARING 707
COM_DIRECT_ACCESS 707
COM_HOLD.. 707
COM_RECEIVE_BUFFER_FLUSH............... 708
COM_SELECT 708
DOS_AUTOEXEC.. 709
DOS_BACKGROUND_EXECUTION 709
DOS_BREAK 710
DOS_DEVICE 710
DOS_FCBS 711
DOS_FCBS_KEEP.. 711

DOS_FILES 712
DOS_LASTDRIVE 712
DOS_SHELL 712
DOS_STARTUP_DRIVE 713
DOS_VERSION 713
HW_NOSOUND 714
HW_ROM_TO_RAM 714
HW_TIMER 714
IDLE_SECONDS 714
IDLE_SENSITIVITY 714
INT_DURING_IO 715
SESSION_PRIORITY 715
WIN-OS/2 settings 716
WIN_RUN_MODE 716
WIN_DDE 717
WIN_CLIPBOARD 717
WIN_ATM 717
All DOS settings 717
Glossary of Terms 718
All DOS And WIN-OS/2 settings 718
Printing and Loading DOS or WIN-OS/2 Settings 720
Reliability of user changes via a text editor 720
Why making changes via a text editor is important 720
Printing DOS or WIN-OS/2 settings 721
Loading DOS or WIN-OS/2 settings 721
REXX Setup of a DOS or WIN-OS/2 Object 725
WIN-OS/2 Window — Separate Session 727
WIN-OS/2 TIMER.DRV Changes for MIDI 728
Summary 728
Appendix A: CD-ROM Contents 731
Bitmaps 732
Icons 733
Object Tool 736
Pointers and Pointer Sets 736
REXX Scripts 737

Appendix B: Developer's Desktop **743**
Desktop Basics 743
System Startup 749
Work Area Folders 751
Word for Windows Setup 753
WIN-OS/2 Setup 759
Appendix C: System Keyboard Definitions **763**
Index **769**

Foreword

I am a perfect example of how OS/2 Warp's power and promise can conquer. I lived in a single-tasking DOS world until 1992. When I accidentally bumped into OS/2 (thanks to my colleague Brad Kliewer), I found that I could print long reports, do difficult queries, download files and continue to do my word processing all simultaneously. No longer would my days be devoted to waiting for the printer to finish a 200-page report before I could return to writing an article. Hours were returned to me. My life was changed. It was as though I had been wandering for years in a desert and suddenly discovered a motorcycle and a road map. That was in the days of 2.0. I became so excited about the operating system that I began *OS/2 Professional* to talk to other computer users about the joys and power of OS/2.

Today, OS/2 has evolved through several updates, solved its technical problems, and now stands as OS/2 Warp, the world's most productive and easy-to-use operating system. Even as I am writing this foreword, I have been checking my e-mail, dragging documents to the printer, and running a database query — all based on a few clicks to my LaunchPad. That's productivity, and it's all based on OS/2's unique interoperability among DOS, Windows, and native OS/2 applications. Now OS/2 Warp stands ready to unlock the powers of the Software Solutions division of IBM, known as SWS. SWS has just released the latest version of DB2 and a palette of powerful functions called WorkGroup. WorkGroup is dynamic groupware that allows a shared address book, shared fax and e-mail, document imaging and processing, and workflow automation. And it all happens from those nifty little drawers in the LaunchPad. If OS/2 Warp is the promise of power, WorkGroup is the delivery.

A global grassroots movement of OS/2 Warp supporters and special interest groups has sprouted up to help users learn to more fully exploit the extraordinary promise and power of OS/2 Warp. And technical help books such as this one are regularly published to fill in the gaps. As editor-in-chief of *OS/2 Professional,* I see OS/2 books every day. None has been more useful than *OS/2 Warp Uncensored.* In just five minutes, I learned enough OS/2 Warp secrets to improve my skills dramatically.

Authors Peter Magid and Ira Schneider are themselves among the most important members of the OS/2 Workplace Shell development team. Therefore, their writing is no ordinary analysis or tip sheet. These chapters contain the real how and why

OS/2 Warp works and how you can use its features to work better — from the perspective of the men who helped build it. Their work is easy enough for the first-time user to understand and benefit from within minutes, and it's significant enough for long-time power users to find it indispensable.

Although OS/2 Warp is not the most widely used operating system in the U.S., it is the most superior operating system in the U.S. Its adoption by millions of devoted users is a clear success for an operating system that has succeeded not on hype, but on genuine quality. OS/2 Warp is generally concentrated in corporate America, but every month, more OS/2 Warp stations migrate down the corporate pyramid to the average user. Moreover, those users are increasingly exporting OS/2 Warp to their home desktops, where people want the same high-octane productivity they enjoy on the job. OS/2 Warp accelerates every user's desktop, whether at home or at work. With *OS/2 Warp Uncensored* in your knapsack, you can travel along with the best of them.

Edwin Black
Editor-in-Chief
OS/2 Professional
Rockville, MD
July 1995

Introduction

This book is an intermediate to advanced power users' guide for OS/2 Warp. It was written by the people who know the most about the product — people who actually *wrote* the code. Who else can give you the inside information about how they intended the product to be used? You will find tips by the developers on how to get the most out of OS/2 Warp and some insight into how we use our own systems.

Conventions Used

Throughout this book, you will see several margin icons, designed to draw your attention to important material in the text:

Uncensored

Some of the features in OS/2 Warp don't work the way you expect them to, but they do work the way we intended them to. We tried to give you some insight into why we designed particular features the way we did. If we were repeatedly asked questions about a feature of the system, we included special information about that feature so you won't have to ask the same questions.

Developer's Desktop

You can follow our lead when customizing parts of your own system. We also included examples of interesting ways to make your system look and behave the way you want it to.

CD-ROM

Check the CD-ROM at the back of this book for items discussed in the text. The CD-ROM includes bitmaps, icons, pointers, an object tool, and several REXX scripts you can use.

Tip

Many of our tips are based on personal experience, but some come from feedback we received from other users.

Warning There are some areas where you may experience trouble with your system. Be careful!

Note Notes and comments give you a little background on certain OS/2 Warp features.

So, go ahead. Read through the sections of the book that interest you. Try out the tips we gave you. And, make OS/2 Warp work *for* you.

What's On The CD-ROM

The CD-ROM at the back of this book includes the following:

- Bitmaps you may use for your Desktop background, the background for other Workplace Shell folders, or your Lockup picture.
- Icons you may use for your folders and other Workplace Shell objects.
- Pointer sets you may use to replace the standard mouse pointers.
- An Object Tool that lets you examine and change the properties of any Workplace Shell object.
- Sample REXX scripts.
- REXX scripts that allow you to recreate Workplace Shell objects that you accidentally deleted or changed.

A Brief History Of OS/2

Here is a brief history of the evolution of OS/2 Warp:

- OS/2 Version 2.0 (released in April 1992) was the first 32-bit version of OS/2. This release introduced our brand new, object-oriented graphical user interface, the Workplace Shell.
- OS/2 Version 2.1 (released in May 1993) contained built-in support for Windows 3.1 applications.
- OS/2 Version 2.1 For Windows (released in November 1993) allowed Windows 3.1 users to maintain their Windows configuration while running OS/2.

- ✘ OS/2 Warp Version 3 (released in October 1994) is faster and requires less memory (RAM) than OS/2 2.1. It supports a much wider range of hardware, has an easier to use installation program, an improved look to the Workplace Shell, and contains a BonusPak of applications (including the IBM Internet Connection and World Wide Web Browser, a Multimedia Viewer, and IBM Works, a suite of application programs).

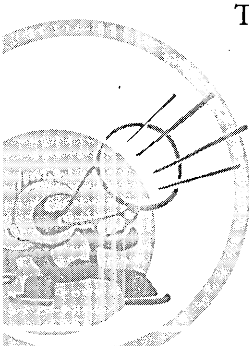
Chapter 1

Installing OS/2 Warp

The Secret Just Beyond That Mountain of Diskettes!

This chapter will discuss in detail some considerations you should keep in mind as you begin to install OS/2 Warp onto your system. A little planning can make the task of installation much easier and more pleasant. You will get some insight into the new features of the installation process, as well as an idea of why we made these changes.

Minimal Hardware Requirements



To run OS/2 Warp, you need a computer with the following hardware:

- ✘ Intel 80386 processor or higher.
- ✘ A minimum of 4MB of memory (RAM). The more memory you have, the better, but 4MB will suffice.
- ✘ A hard disk with 35MB to 60MB of available disk space. Multimedia Support can take an additional 10MB of disk space.
- ✘ A 1.44MB diskette drive (one that accepts 3.5-inch diskettes).

- At least VGA video support (i.e. 640 x 480 x 16).
- A pointing device such as a mouse.
- If you are installing multimedia support or HPFS you need at least 6MB of memory.

Recommended Hardware Support

You will be able to run OS/2 Warp with the miminum requirements listed above. However, you should consider upgrading your system to get better performance. The following hardware is recommended to run OS/2 Warp:

- An Intel 80486 processor or higher.
- 8MB of memory (RAM). Generally speaking, buying memory is a better deal than upgrading your processor unless you are running CPU-intensive applications like a CAD-CAM program. Most applications in today's market are not CPU bound, but rather I/O bound, so the more memory you have, the better off you will be. I suggest putting in at least 12MB to 16MB of memory.
- A hard disk with 100MB of disk space. Today's multimedia applications demand a great deal of disk space, so you will want to have at least 300MB to 500MB available for these types of applications. This space does not have to reside on your installation partition. I recommend that you set up a partition solely for applications. Also, the price of hard drives has dropped significantly in the past few months, and in fact you can buy a 1GB drive for around $350 to $400.
- A CD-ROM drive, preferably a quad-speed (4X), although 2X is fine. A lot of useful applications ship on CD-ROM, and you will want to take advantage of these. Furthermore, the OS/2 Warp installation is much easier if you have a CD-ROM drive because you don't have to shuffle diskettes.
- An SVGA adapter and display. OS/2 Warp has improved its video support tremendously, so you will want to upgrade your hardware. Also, there are video accelerators on the market that improve the performance of multimedia applications and games that use full-motion video.

Which Version of OS/2 Warp Should I Buy?

Two different versions of OS/2 Warp exist on the market today. The first one released was the version called OS/2 Warp Version 3. This release is intended for those users who have an existing copy of Windows 3.1, Windows 3.11, or Windows for Workgroups 3.11 installed on their system. This version of OS/2 Warp uses the existing version of Windows that is installed on your hard disk to give you Windows support in OS/2. If you do not have Windows installed, you can still install OS/2 Warp Version 3; however, you will not get any Windows support.

If you do not already have one of the versions of Windows listed above, then you should plan to use OS/2 Warp Version 3 with WIN-OS/2. This version of OS/2 Warp ships with Windows 3.1 support built into the product.

Regardless of which version you choose, there is no difference in the final installed version of OS/2 Warp. You get exactly the same features in either product.

From this point on, I will refer to both products as OS/2 Warp. If I need to distinguish between the two versions, I will do so; otherwise all references to OS/2 Warp apply equally to both versions.

Choosing the Right Installation Media

OS/2 Warp ships on 3.5-inch diskettes as well as on a CD-ROM. If you have a CD-ROM that is supported by OS/2 Warp, you would be better off choosing the CD-ROM package (which also includes two OS/2 Warp diskettes). Installing OS/2 Warp from CD-ROM saves you some time shuffling the diskettes during the installation process. You can also create a set of OS/2 Warp diskettes from the CD-ROM package.

OS/2 Warp does not ship on 5.25-inch diskettes. If you have two diskette drives on your computer, and drive A: is a 5.25-inch diskette drive and drive B: is a 3.5-inch diskette drive, then you can reconfigure your diskette drives to install OS/2 Warp from the 3.5-inch diskette drive.

Preparing for Installation

Tip

First things first. OS/2 Warp ships with an installation README file. You definitely should read this prior to installing OS/2 Warp. There is some valuable information in this file, and you should not overlook it, as it provides late-breaking information to help make installation painless. In OS/2 Warp we have made reading the README file a step easier by including an editor on the diskettes. The following steps will guide you through this simple process:

1. Insert the OS/2 Warp Installation diskette into drive A: and restart your computer.
2. Remove the OS/2 Warp Installation diskette and insert diskette 1 into drive A: when prompted to do so, and then press **Enter.**
3. At the Welcome screen, press **F3** to display a command prompt.
4. Remove OS/2 Warp diskette 1 and reinsert the OS/2 Warp Installation diskette into drive A:.
5. Type **TEDIT README.INS** and then press **Enter.**
6. When you have finished reading the README.INS file, press F4 to quit the editor.

If you are planning to install OS/2 Warp (that is, the version without WIN-OS/2), you will need the original Windows diskettes. If you bought a system preloaded with Windows, you may have been given a CD-ROM that has Windows on it by the manufacturer. They should also have given you a utility program that can create Windows diskettes from the CD-ROM. You should do this before you install OS/2 Warp. If you do not have a CD-ROM drive, you must have the original Windows diskettes.

Installing OS/2 Warp on large partitions

Tip

If you plan to install OS/2 Warp on a partition that is greater than 528MB or 1024 cylinders, make sure you have an IDE hard disk and a BIOS translation utility program that can access disk space beyond the 1024 cylinder boundary. If you do not know if your partition size is greater than the 1024 cylinder boundary, do the following:

1 Insert the OS/2 Warp Installation diskette into drive A: and restart your computer.

2 Remove the OS/2 Warp Installation diskette and insert diskette 1 into drive A: when prompted to do so, then press **Enter.**

3 At the Welcome screen, press **F3** to display a command prompt.

4 Type **FDISK /query** and press **Enter.**

An entry at the end of the table of **BIOS=NNNN MB means that the partition is greater than 1024 cylinders. The BIOS limit for that disk is NNNN MB where NNNN is the maximum size you can make the partition if you want to install OS/2 Warp onto it.

Ensure that you have properly shut your system down prior to starting the OS/2 Warp installation process. If you have problems shutting your system down, run the CHKDSK utility first. The following steps guide you through this process:

1 Insert the OS/2 Warp Installation diskette into drive A: and restart your computer.

2 Remove the OS/2 Warp Installation diskette and insert diskette 1 into drive A: when prompted to do so, and then press **Enter.**

3 At the Welcome screen, press **F3** to display a command prompt.

4 Remove OS/2 Warp diskette 1, and insert OS/2 Warp diskette 2 into drive A:.

5 Determine the drive you want to install OS/2 Warp on.

6 At the command prompt type **CHKDSK [d:] /f,** and then press **Enter.** The parameter [d:] signifies the drive you want OS/2 Warp to be installed on.

So, if you want to install OS/2 Warp to drive C:, you would type **CHKDSK C: /f** and then press **Enter.**

Installing OS/2 Warp on compressed drives

If you are planning to install OS/2 Warp onto a compressed drive, you may have some problems. For example, if you are using DoubleSpace on the drive you want to install OS/2 Warp onto, then you must decompress this drive prior to beginning installation.

Installing OS/2 Warp using a third-party device

Tip

If you are using a third-party device that requires a device driver not shipped in the OS/2 Warp package, you will have to copy the device driver to diskette 1 and modify the CONFIG.SYS file on diskette 1. To do so, follow these steps:

1. Insert the OS/2 Warp Installation diskette into drive A: and restart your computer.
2. Remove the OS/2 Warp Installation diskette and insert diskette 1 into drive A: when prompted to do so, and then press **Enter.**
3. At the Welcome screen, press **F3** to display a command prompt.
4. Type **TEDIT CONFIG.SYS** and press **Enter.** This will allow you to edit CONFIG.SYS. At the bottom of the file, type **BASEDEV=XXXXXX,** where XXXXXX is the name of the third-party device driver and any parameters that it may need. Refer to the manufacturer's instructions.
5. At the bottom of CONFIG.SYS, type the statement **SAVECONNECT=1.** This will ensure that the files and CONFIG.SYS statements you have added to diskette 1 will be properly added to the system at the end of installation.
6. Copy the third-party device driver onto diskette 1.

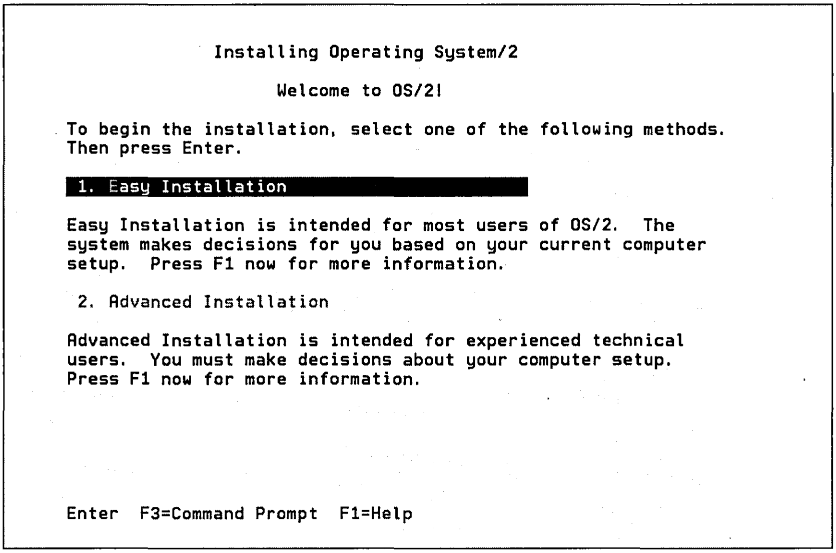
Installing a fix for a file on diskette 1

If you have a fix for a file that you were told to copy onto diskette 1 prior to installation and you are installing using the OS/2 Warp CD-ROM package, do the following:

1. Insert the OS/2 Warp Installation diskette into drive A: and restart your computer.
2. Remove the OS/2 Warp Installation diskette and insert diskette 1 into drive A: when prompted to do so, and then press **Enter.**
3. At the Welcome screen, press **F3** key to display a command prompt.
4. Type **TEDIT CONFIG.SYS** and press **Enter.** This will allow you to edit CONFIG.SYS. At the bottom of the file, type **SET COPYFROMFLOPPY=1.** This will copy the file you have replaced on diskette 1 from the diskette instead of the CD-ROM.

Phase 1 of OS/2 Warp Installation

The OS/2 Warp Installation department went to great lengths to give you a quick and easy installation, but at the same time allow enough flexibility for advanced users to create custom configurations. The result was a two-selection option that is shown in Figure 1-1. These two selections are referred to as Easy Installation and Advanced Installation. If you choose **Easy Installation,** this is probably the only decision you'll have to make during the entire installation process. At all other panels you can simply press **Enter** to proceed.



```
                    Installing Operating System/2

                          Welcome to OS/2!

     To begin the installation, select one of the following methods.
     Then press Enter.

      1. Easy Installation

     Easy Installation is intended for most users of OS/2.  The
     system makes decisions for you based on your current computer
     setup.  Press F1 now for more information.

      2. Advanced Installation

     Advanced Installation is intended for experienced technical
     users.  You must make decisions about your computer setup.
     Press F1 now for more information.




     Enter  F3=Command Prompt  F1=Help
```

Figure 1-1.
Welcome Dialog

Using the Easy Installation option

Easy Installation, as its name implies, is intended for a novice user or one who does not want to deal with the particulars of configuring his or her system during the installation process. If you choose **Easy Installation**, you will get the following results:

1 You will install OS/2 Warp on the C: partition. If the C: partition is not acceptable to install on (because, for instance, the partition size is too small, less than 35MB), then you will see a fatal error panel displayed, and the installation will abort.

2. If you have DOS version 3.3 or greater installed or if the C: drive is formatted as FAT prior to installation, you will get the Dual Boot icon. See Dual boot Support section for more details.
3. If the partition is not formatted or partitioned, the entire drive will be partitioned in one single partition, and the partition will be formatted as FAT.
4. You will not be presented with an option to change your installation partition, nor will you be presented with the ability to repartition your system during installation. The installation program will also detect whether it needs to format the partition or not.
5. Your OS/2, Windows, and DOS applications will be automatically migrated to your Desktop for you.
6. Your old CONFIG.SYS and AUTOEXEC.BAT files get migrated over for you as well.
7. You will get the following features:
    - All OS/2 Warp documentation
    - All system fonts
    - All system utilities
    - All tools and games
    - OS/2 DOS Support
    - WIN-OS/2 Support (if Windows exists, in the case of OS/2 Warp Version 3)
    - REXX
    - Multimedia Software Support (if hardware is detected and you have at least 6MB of memory [RAM])
    - Serviceability and diagnostic aids
    - Optional bitmaps

- All hardware support that is determined to be necessary, including:
  - APM
  - PCMCIA
  - CD-ROM
  - SCSI
  - Multimedia
  - Printer (you may have to select the printer that is attached to your system)
  - Video (VGA support)

Using the Advanced Installation option

The Advanced Installation option is intended for a more experienced computer user, or in particular one who is familiar with a previous version of the OS/2 installation program. The panel flow will look fairly similar to previous versions of the OS/2 installation program with a few new features and some bells and whistles.

You should use the **Advanced Installation** option if you want to do any of the following:

✘ Install OS/2 Warp to a drive other than C:

✘ Format your installation partition HPFS

✘ Install the Boot Manager

✘ Customize your CONFIG.SYS and AUTOEXEC.BAT

✘ Selectively install hardware and software support

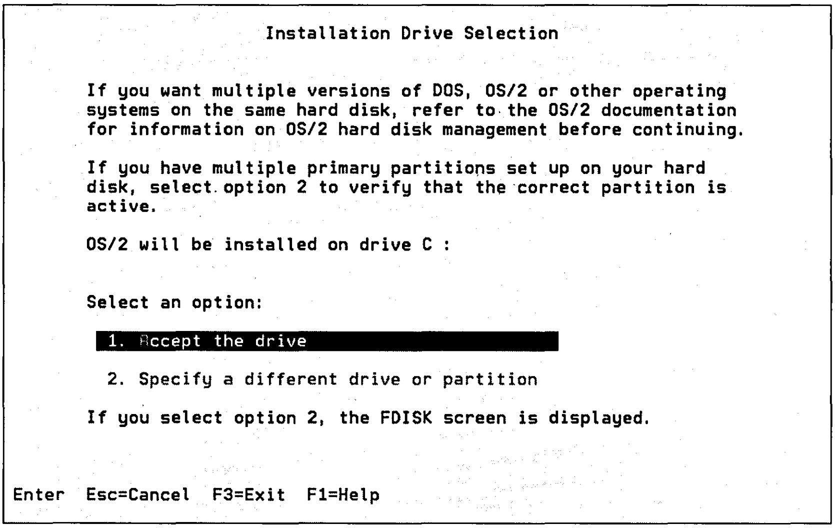
✘ Repartition your hard disk during the installation process

✘ Selectively migrate applications

Installing Boot Manager

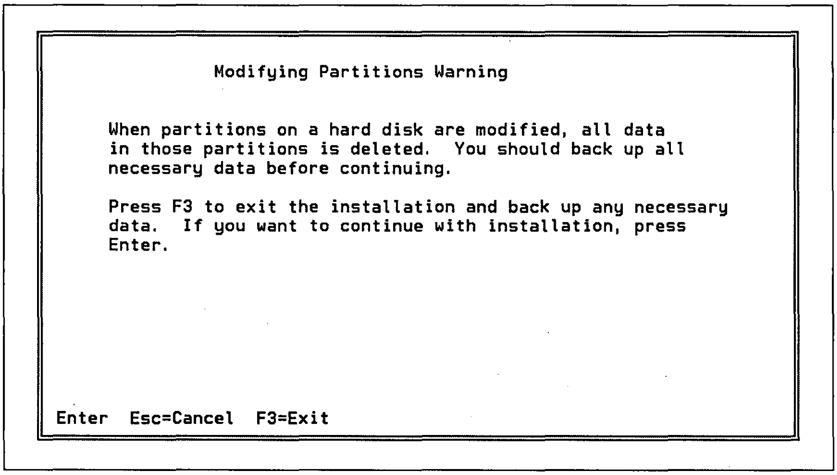
The Advanced Installation option will present you with the option of installing OS/2 Warp to another partition. This is shown in Figure 1-2. You should select option 2. This will bring up the warning panel shown in Figure 1-3.

Figure 1-2.
Installation Drive Selection Dialog



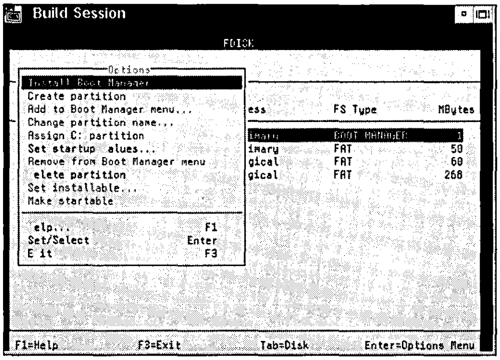
```
                     Installation Drive Selection

       If you want multiple versions of DOS, OS/2 or other operating
       systems on the same hard disk, refer to the OS/2 documentation
       for information on OS/2 hard disk management before continuing.

       If you have multiple primary partitions set up on your hard
       disk, select option 2 to verify that the correct partition is
       active.

       OS/2 will be installed on drive C :


       Select an option:

         1. Accept the drive

         2. Specify a different drive or partition

       If you select option 2, the FDISK screen is displayed.


Enter  Esc=Cancel  F3=Exit  F1=Help
```

Figure 1-3.
Modifying Partitions Warning Dialog



```
                Modifying Partitions Warning

     When partitions on a hard disk are modified, all data
     in those partitions is deleted.  You should back up all
     necessary data before continuing.

     Press F3 to exit the installation and back up any necessary
     data.  If you want to continue with installation, press
     Enter.





Enter  Esc=Cancel  F3=Exit
```

Press **Enter** when this panel is displayed. The next screen that is displayed is the FDISK screen. This will show you a table of the partitions that you have on your hard disk, or an entry called None in the Status column. If you do not have free space at the beginning or the end of the hard disk, you will have to remove one of the partitions, because Boot Manager can only be installed at the beginning or end of the hard disk. Once you have ensured that there is space at the beginning or end of the hard disk, move the highlight bar to either free space location and press **Enter**. This will bring up a submenu, and the first selection will be Install Boot Manager (see Figure 1-4).



```
Build Session
                              FDISK
   ─────────Options──────────
   Install Boot Manager
   Create partition
   Add to Boot Manager menu...     ess        FS Type          MBytes
   Change partition name...
   Assign C: partition             imary      BOOT MANAGER          1
   Set startup  alues...           imary      FAT                  50
   Remove from Boot Manager menu   gical      FAT                  60
    elete partition                gical      FAT                 268
   Set installable...
   Make startable
   ──────────────────────────
    elp...                 F1
   Set/Select           Enter
   E it                    F3

F1=Help        F3=Exit        Tab=Disk        Enter=Options Menu
```

Figure 1-4.
FDISK - Install Boot Manager Option

Once you have created the Boot Manager, you must make the Boot Manager a startable partition. To do this, simply highlight the **Boot Manager** entry and press **Enter**. The last option displayed is Make Startable. Highlight that option and press **Enter**.

Creating other partitions

If you have enough disk space, consider creating a separate partition for OS/2 Warp. I would suggest that you install DOS and Windows on drive C:, and OS/2 on another partition. Create yet another partition for your data and applications. This method minimizes the risk of one thing damaging another.

To create the installation partition, you must highlight the **Free Space** option and press **Enter**. You can decide how large the partition should be, but it must be at least 35MB (40MB for OS/2 Warp with WIN-OS/2). Next you must decide whether to make the partition Primary or Logical.

Keep in mind that you can only see one Primary partition at any given time; however, you can see all Logical partitions at all times. You must have at least one Primary partition. Press **Enter** and select **Set Installable** from the menu.

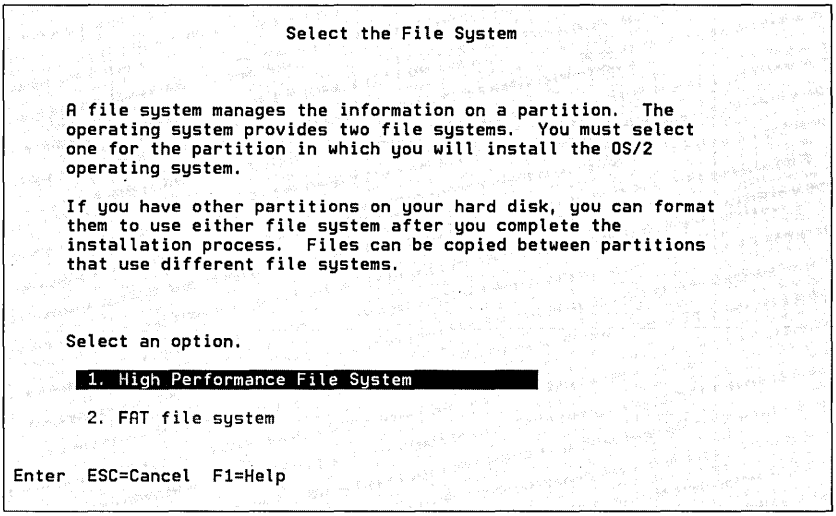
Now you should go about creating any other partitions you may want in the manner described above. Once you have finished, press **F3** twice. When you exit FDISK, you will see a panel instructing you to remove OS/2 Warp diskette 1, and reinsert the OS/2 Warp Installation diskette into drive A:. You have now successfully completed the hard disk partitioning phase.

If you did not repartition your system, you will not be asked to reboot, and the Accept Drive panel will be redisplayed. Press **Enter** at this point. You will be asked to remove OS/2 Warp diskette 1 and insert OS/2 Warp diskette 2 into drive A:.

Formatting the installation partition

The installation program now knows where you want to install OS/2 Warp. If you created the installation partition in the previous step, the next panel you will see is one that asks you what file system type you want installed on the installation partition (see Figure 1-5).

Figure 1-5. Select the File System Dialog



```
                        Select the File System


      A file system manages the information on a partition.  The
      operating system provides two file systems.  You must select
      one for the partition in which you will install the OS/2
      operating system.

      If you have other partitions on your hard disk, you can format
      them to use either file system after you complete the
      installation process.  Files can be copied between partitions
      that use different file systems.


      Select an option.

       1. High Performance File System

       2. FAT file system

 Enter  ESC=Cancel  F1=Help
```

A point to consider in choosing a file system is that if you want to use the dual boot feature, you must format the partition FAT. Consider HPFS if you do not want dual boot and you have at least 8MB of RAM and a large partition (that is, greater than 100MB). In the long run HPFS utilizes disk space better than FAT does. There are some utilities on the market called DEFRAG utilities that help FAT utilize disk space better, but why use these if you can avoid the problem altogether? One limitation of HPFS is that if you boot real DOS, as opposed to the DOS support OS/2 provides, you will not be able to see HPFS partitions.

If the installation partition already existed, the installation program will check for compressed drives and display a warning as needed. Next you will be asked whether you want to format the installation partition. If you elect to format, you will lose all data on that partition, so you may want to back up your data first. If you choose to format, you will see the panel shown previously in Figure 1-5. Your installation partition will now be formatted. If you want to format other partitions, you will be able to do so in the second phase of installation.

If you have a problem formatting during phase 1, do the following:

1. Insert the OS/2 Warp Installation diskette into drive A: and restart your computer.
2. Remove the OS/2 Warp Installation diskette and insert diskette 1 into drive A: when prompted to do so, and then press **Enter.**
3. At the Welcome screen, press **F3** to display a command prompt.
4. Remove OS/2 Warp diskette 1, and insert OS/2 Warp diskette 2 into drive A:.
5. Type **FORMAT** [d:] **/L** and press **Enter.** [d:] represents the drive you are formatting. For example, if you want to format drive D:, you would type **FORMAT D: /L** and then press **Enter.**

Once this is complete, restart the installation.

Informational messages during phase 1

Uncensored

One of the features that was added to the OS/2 Warp installation process was the new look and feel. Aside from more lively colors (we all got tired of the dull gray), the installation program now displays panels during the file copy process that tell you about new features of OS/2 Warp and new offerings from IBM. You may pick up some useful information from these panels.

Dual boot support

To get dual boot support, your installation partition must be formatted with the FAT file system. If you installed OS/2 Warp over DOS version 3.3 or greater, then upon completion of phase 2 of installation, you will have fual dual boot support. If you did not install over DOS, then when you double-click on the dual boot icon, the system will ask you if you want to install DOS at that time. This is another new feature of the OS/2 Warp installation program.

Completing phase 1

At this point you will simply be asked to remove and insert the rest of the diskettes through diskette 6, then reinsert the Installation diskette followed by diskette 1 into drive A:. The final panel you will be shown is one that instructs you to remove the diskette from drive A: and press **Enter**. You have now installed the main part of the operating system. The remainder of the installation is graphical, and it allows you to selectively install software and hardware options.

Phase 2 of OS/2 Warp Installation

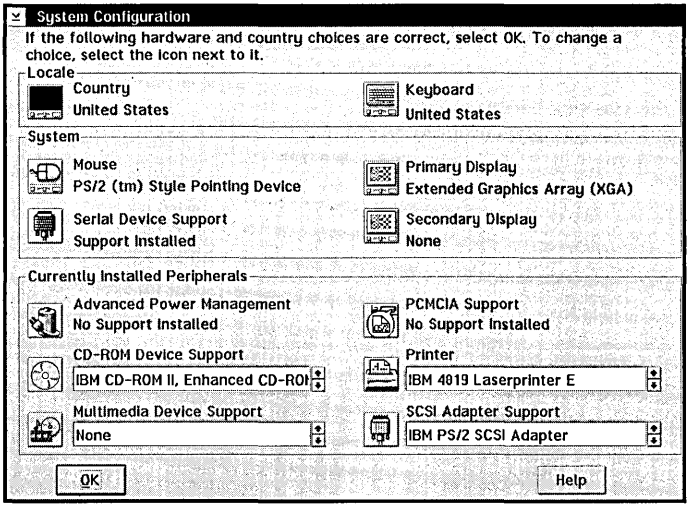
To begin the second phase of installation, remove the diskette from drive A: and press **Enter**. You should be booting off of the hard disk and into a graphical environment.

You have now booted into what is known as the Maintenance Partition. This is a functional Workplace Shell with limited capability. You can open the OS/2 System folder and use any of the objects at the same time installation is running. You can close the installation program at this time and then reinvoke it by either rebooting or clicking on the selective install icon. The Maintenance Partition is persistent, even after you have completed the second phase of installation. More on the Maintenance Partition later.

Each of the dialogs that are displayed during the first and second phases of OS/2 Warp installation has a help panel associated with it. You can press F1 at any time during the installation process to display an associated help panel.

Country information and hardware configuration

Now it is time to select some additional options and confirm that the OS/2 Warp installation program correctly detected the hardware that you have attached to your system. The System Configuration dialog (see Figure 1-6) displays the country configuration and the hardware devices that are currently attached to your system.



System Configuration

If the following hardware and country choices are correct, select OK. To change a choice, select the icon next to it.

Locale

Country
United States

Keyboard
United States

System

Mouse
PS/2 (tm) Style Pointing Device

Primary Display
Extended Graphics Array (XGA)

Serial Device Support
Support Installed

Secondary Display
None

Currently Installed Peripherals

Advanced Power Management
No Support Installed

PCMCIA Support
No Support Installed

CD-ROM Device Support
IBM CD-ROM II, Enhanced CD-ROI

Printer
IBM 4019 Laserprinter E

Multimedia Device Support
None

SCSI Adapter Support
IBM PS/2 SCSI Adapter

Figure 1-6.
System Configuration Dialog

Uncensored

OS/2 Warp goes through this confirmation process because, though it usually recognizes all compatible devices, every once in a while there is the possibility that a device will be incorrectly detected due to bad hardware or incorrectly configured devices (for example, IRQs are not correct). In addition, there is the possibility that you will use a different device after the installation is complete than you did during the installation process; therefore, instead of making you come back into selective install, we allow you to select any and all hardware the first time around. You will probably only need to hit **Enter** on this panel without changing anything. If you do have a problem or need to change something, simply single-click on the smart icon representing the device.

Uncensored

Two of the new features that were added to the OS/2 Warp installation program are the ability to install and configure multimedia support during the main installation process and the ability to install more than one printer during the initial installation. In previous releases of OS/2, you had to first install the OS/2 Warp base product, reboot your system, and then invoke MINSTALL.EXE, the multimedia installation program. Similarly, you could only install one printer during the initial installation. If you wanted to install more printers, you had to invoke the print object after the installation was complete.

Installing and configuring printers

After the System Configuration dialog is displayed, the Select System Default Printer dialog is shown (see Figure 1-7). This dialog is an alphabetical list box containing all the printers OS/2 Warp supports. The easiest way to find the printer you want to install support for is to type the first letter of the name of the manufacturer. This will display the first printer that it matches, not necessarily the one you want selected, but the list of printers to go through will be shortened considerably. Making a selection on this dialog sets the system default printer. You will also have to select the port that this printer is attached to.

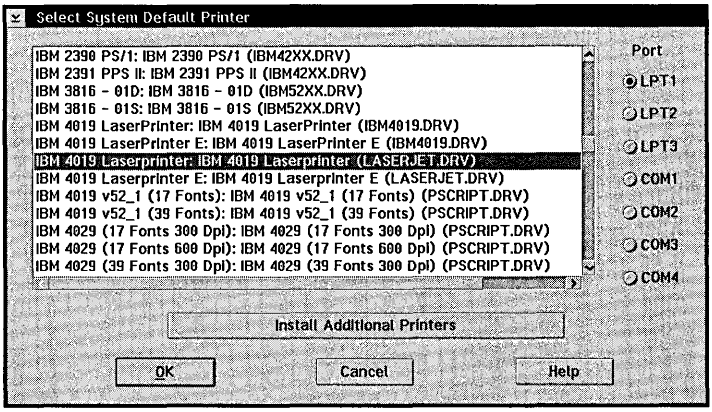


Figure 1-7. Select System Default Printer Dialog

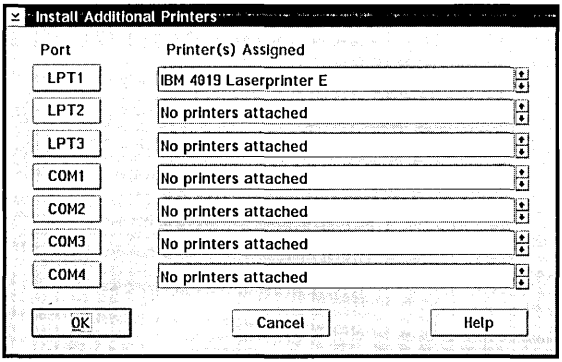
Select System Default Printer

IBM 2390 PS/1: IBM 2390 PS/1 (IBM42XX.DRV)
IBM 2391 PPS II: IBM 2391 PPS II (IBM42XX.DRV)
IBM 3816 - 01D: IBM 3816 - 01D (IBM52XX.DRV)
IBM 3816 - 01S: IBM 3816 - 01S (IBM52XX.DRV)
IBM 4019 LaserPrinter: IBM 4019 LaserPrinter (IBM4019.DRV)
IBM 4019 LaserPrinter E: IBM 4019 LaserPrinter E (IBM4019.DRV)
IBM 4019 Laserprinter: IBM 4019 Laserprinter (LASERJET.DRV)
IBM 4019 Laserprinter E: IBM 4019 Laserprinter E (LASERJET.DRV)
IBM 4019 v52_1 (17 Fonts): IBM 4019 v52_1 (17 Fonts) (PSCRIPT.DRV)
IBM 4019 v52_1 (39 Fonts): IBM 4019 v52_1 (39 Fonts) (PSCRIPT.DRV)
IBM 4029 (17 Fonts 300 Dpi): IBM 4029 (17 Fonts 300 Dpi) (PSCRIPT.DRV)
IBM 4029 (17 Fonts 600 Dpi): IBM 4029 (17 Fonts 600 Dpi) (PSCRIPT.DRV)
IBM 4029 (39 Fonts 300 Dpi): IBM 4029 (39 Fonts 300 Dpi) (PSCRIPT.DRV)

Port
● LPT1
○ LPT2
○ LPT3
○ COM1
○ COM2
○ COM3
○ COM4

If you want to install additional printers, you should click on the **Install Additional Printers** button. This will bring up a dialog like the one shown in Figure 1-8. Click on the button for the port to which your printer(s) are attached. This will bring up a dialog identical to the dialog shown in Figure 1-7. This is a multi-select list box — that is, you can select more than one entry. When you are through selecting the printers, you should click on the **OK** button. Go through the same process for all ports that you have printer(s) attached to.

Figure 1-8.
Install Additional Printers Dialog

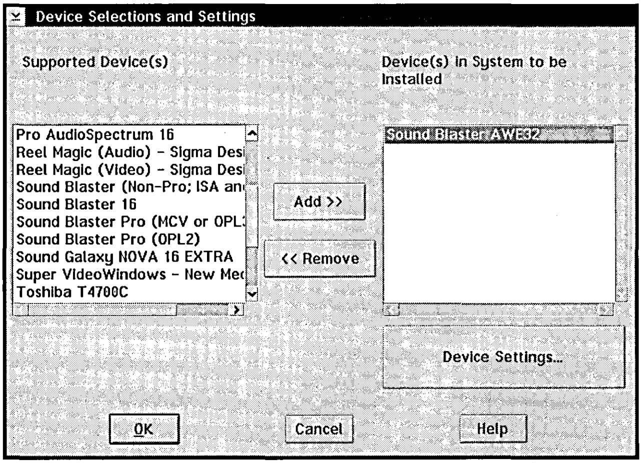


Port	Printer(s) Assigned
LPT1	IBM 4019 Laserprinter E
LPT2	No printers attached
LPT3	No printers attached
COM1	No printers attached
COM2	No printers attached
COM3	No printers attached
COM4	No printers attached

Installing and configuring multimedia hardware

As I mentioned earlier, the latest version of the OS/2 Warp installation program added the ability to install and configure your multimedia hardware, specifically your sound card(s). The OS/2 Warp installation program will automatically detect the sound card that you have attached to your system. However, if you want to select any additional cards or change the settings that have been made for your particular sound card, you can do so by clicking on the **Multimedia Device Support** icon. This will bring up the Device Selections and Settings dialog, shown in Figure 1-9.

Figure 1-9.
Device Selections and Settings Dialog



Supported Device(s)

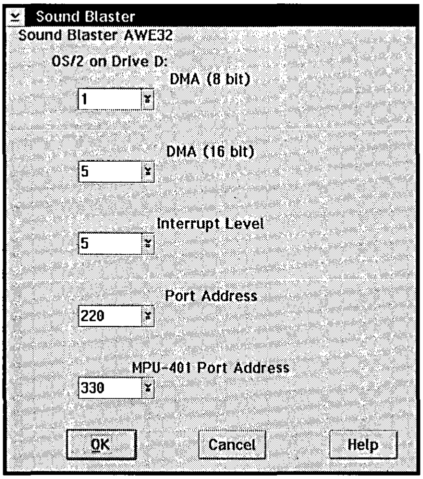
- Pro AudioSpectrum 16
- Reel Magic (Audio) - Sigma Desi
- Reel Magic (Video) - Sigma Desi
- Sound Blaster (Non-Pro; ISA an
- Sound Blaster 16
- Sound Blaster Pro (MCV or OPL:
- Sound Blaster Pro (OPL2)
- Sound Galaxy NOVA 16 EXTRA
- Super VideoWindows - New Me
- Toshiba T4700C

Device(s) in System to be Installed

- Sound Blaster AWE32

To add any sound card support, select the sound card from the list box labeled Supported Devices and click on the **Add** button. You should now see the sound card in the right list box labeled Device(s) in System to be Installed. To remove support for a card (not uninstall the support), you should click on the sound card in the right list box and then click on the **Remove** button.

You can also view and/or change the device settings for a sound card by clicking on the **Device Settings . . .** button. This will bring up a dialog similar to the one shown in Figure 1-10.



Sound Blaster AWE32

OS/2 on Drive D:

DMA (8 bit): 1

DMA (16 bit): 5

Interrupt Level: 5

Port Address: 220

MPU-401 Port Address: 330

Figure 1-10. Sound Blaster Configuration Dialog

Use the drop-down list to change the values of the different settings for your particular card. If the card does not require any settings, a dialog telling you so will be displayed when you click on the **Device Settings . . .** button.

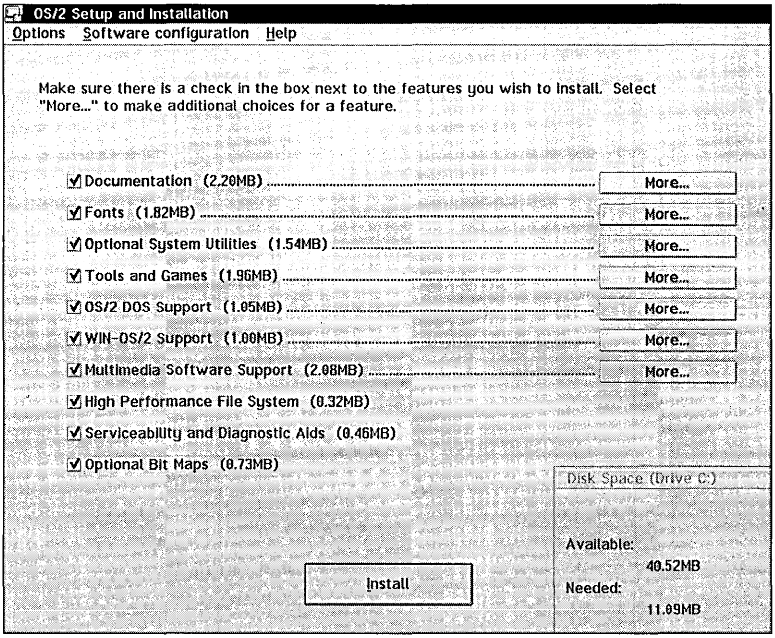
Installing and configuring other hardware

Like the Printer Selections dialog and Multimedia Selections dialog, the rest of the icons in the System Configuration dialog each bring up one or more dialogs that take you through the configuration process for that hardware.

If you are familiar with a previous version of OS/2, one difference you will notice on the System Configuration dialog is that we moved the APM selection, PCMCIA selection, and Serial Support selection to this panel because they are more hardware-related than software-related.

Software selections

When you are through selecting and/or confirming the choices on the System Configuration dialog, click on the OK button. This will bring up the OS/2 Setup and Installation dialog as shown in Figure 1-11.



Make sure there is a check in the box next to the features you wish to install. Select "More..." to make additional choices for a feature.

- ☑ Documentation (2.20MB)
- ☑ Fonts (1.82MB)
- ☑ Optional System Utilities (1.54MB)
- ☑ Tools and Games (1.96MB)
- ☑ OS/2 DOS Support (1.05MB)
- ☑ WIN-OS/2 Support (1.00MB)
- ☑ Multimedia Software Support (2.08MB)
- ☑ High Performance File System (0.32MB)
- ☑ Serviceability and Diagnostic Aids (0.46MB)
- ☑ Optional Bit Maps (0.73MB)

Available: 40.52MB
Needed: 11.09MB

Figure 1-11.
OS/2 Setup and Installation Dialog

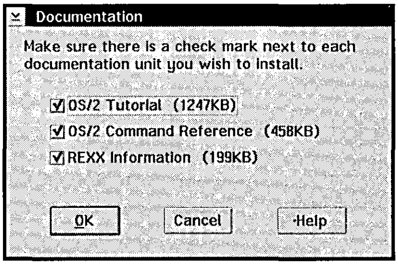
The OS/2 Setup and Installation dialog lets you select the software features that you want to install by checking the appropriate boxes. You will notice that the amount of hard-disk space each feature requires is displayed to the right of the feature in parentheses. The total amount of disk space needed to install all the selected features and the amount of free space you currently have in the partition are shown in the lower-right corner of the OS/2 Setup and Installation dialog. A More . . . button to the right of a selection indicates that you can choose to install only part of that selection; click on the More . . . button to see the available options.

You can save hard-disk space by choosing not to install some features. However, if you have plenty of hard-disk space, it's best to install everything so that you don't have to worry about coming back and installing an option you decide you want after installation is complete. The next sections describe the available software features.

Installing documentation

Tip

There are three selectable options associated with Documentation, and these are displayed in the Documentation dialog, shown in Figure 1-12. It's a good idea to select the **Tutorial** so that you can learn about all the new things in OS/2 Warp. However, if you do not use REXX or the command-line interfaces much, you should deselect the OS/2 Command Reference and REXX Information selections. Remember that each of these selections refers to on-line documentation and not the features themselves. If you deselect REXX Information, you still can use REXX; you just won't have access to the on-line reference.



Documentation

Make sure there is a check mark next to each documentation unit you wish to install.

☑ OS/2 Tutorial (1247KB)
☑ OS/2 Command Reference (458KB)
☑ REXX Information (199KB)

Figure 1-12. Documentation Dialog

Installing fonts

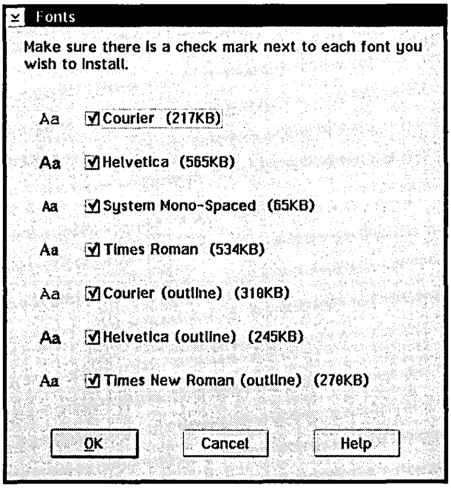
OS/2 Warp includes Adobe Type Manager (ATM) fonts that work with both Presentation Manager applications and Windows applications. OS/2 Warp does not install True Type fonts during the initial install. You can install the True Type fonts for Windows support from a WIN-OS/2 session through the Control Panel icon. Similarly, you can install additional ATM fonts for Windows support through the ATM Control panel.

OS/2 Warp installation gives you an option to choose between seven different fonts, as shown in the Fonts dialog (see Figure 1-13). However, you always get the Helvetica font, as well as the Courier (outline) font. If you deselect the Courier font, you will get the System Mono-Spaced font.

Uncensored

The reason for this is that the OS/2 Book Manager requires the Helvetica font, and the 32-bit Graphics Engine needs the Courier (outline) font to run correctly. In addition, OS/2's Help Manager needs either the Courier font or the System Mono-Spaced font to run correctly.

Figure 1-13.
Fonts Dialog



Make sure there is a check mark next to each font you wish to install.

Aa ☑ Courier (217KB)

Aa ☑ Helvetica (565KB)

Aa ☑ System Mono-Spaced (65KB)

Aa ☑ Times Roman (534KB)

Aa ☑ Courier (outline) (310KB)

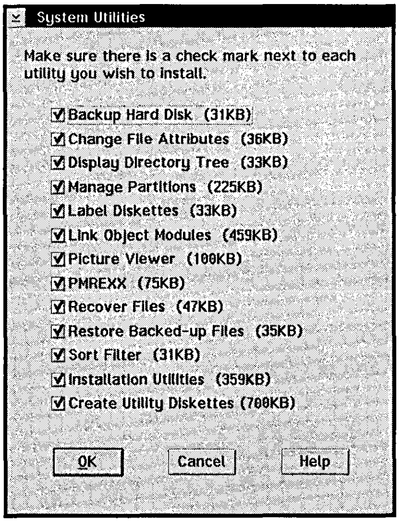
Aa ☑ Helvetica (outline) (245KB)

Aa ☑ Times New Roman (outline) (270KB)

Installing optional system utilities

OS/2 Warp ships with several useful utilities. The System Utilities dialog, shown in Figure 1-14, displays all the selectively installable utilities. You probably will not have a need for the Installation utilities, unless you are using some old software like version 1.x OS/2 Developers Toolkit. Likewise, you may not need the Link Object Modules utility, especially if you have purchased the OS/2 Developers Toolkit.

Figure 1-14.
System Utilities Dialog

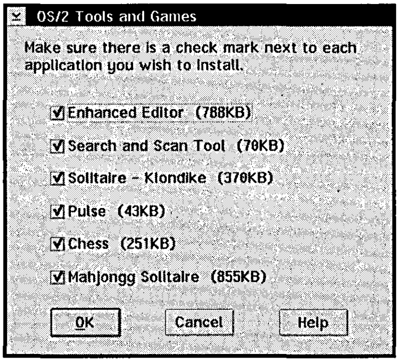


Make sure there is a check mark next to each utility you wish to install.

☑ Backup Hard Disk (31KB)
☑ Change File Attributes (36KB)
☑ Display Directory Tree (33KB)
☑ Manage Partitions (225KB)
☑ Label Diskettes (33KB)
☑ Link Object Modules (459KB)
☑ Picture Viewer (100KB)
☑ PMREXX (75KB)
☑ Recover Files (47KB)
☑ Restore Backed-up Files (35KB)
☑ Sort Filter (31KB)
☑ Installation Utilities (359KB)
☑ Create Utility Diskettes (700KB)

Installing tools and games

Figure 1-15 shows the OS/2 Tools and Games dialog. You are presented with an array of tools and games that you can selectively install. You can save some hard-disk space by deselecting any number of these, though they are quite entertaining.


OS/2 Tools and Games

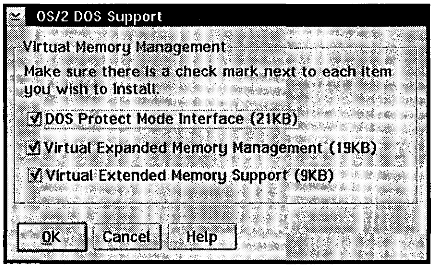
Make sure there is a check mark next to each application you wish to install.

☑ Enhanced Editor (788KB)
☑ Search and Scan Tool (70KB)
☑ Solitaire - Klondike (370KB)
☑ Pulse (43KB)
☑ Chess (251KB)
☑ Mahjongg Solitaire (855KB)

Figure 1-15. OS/2 Tools and Games Dialog

Installing OS/2 DOS support

The OS/2 DOS Support dialog is shown in Figure 1-16. You need to select this option if you want WIN-OS/2 support. It is good idea to install all three DOS memory management options.


OS/2 DOS Support

Virtual Memory Management

Make sure there is a check mark next to each item you wish to install.

☑ DOS Protect Mode Interface (21KB)
☑ Virtual Expanded Memory Management (19KB)
☑ Virtual Extended Memory Support (9KB)

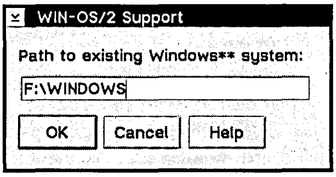
Figure 1-16. OS/2 DOS Support Dialog

Installing OS/2 WIN-OS/2 support

Uncensored

One of the new features of OS/2 Warp is its ability to run WIN32s applications. If you are installing OS/2 Warp without WIN-OS/2, you will get a dialog that prompts you for the path to your existing version of Windows 3.1, Windows 3.11, or Windows for Workgroups. See Figure 1-17. Choosing this selection will cause the OS/2 Warp installation program to prompt for the Windows diskettes. If the OS/2 Warp installation program did not find Windows anywhere on your hard disk, then you will not see the WIN-OS/2 Support selection on the OS/2 Setup and Installation dialog.

Figure 1-17.
WIN-OS/2 Support Dialog



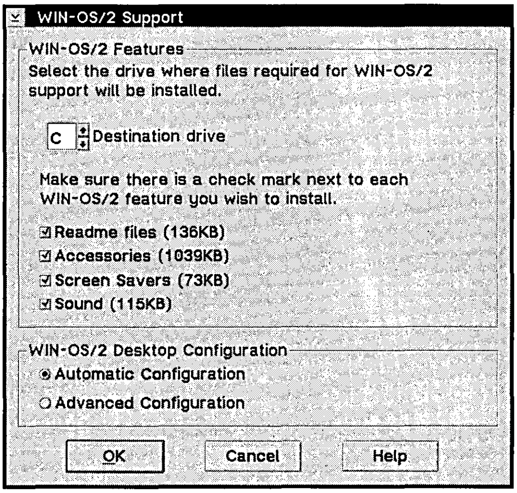
WIN-OS/2 Support

Path to existing Windows** system:

F:\WINDOWS

If you are installing OS/2 Warp with WIN-OS/2 support, you will see the WIN-OS/2 Support dialog, as shown in Figure 1-18. WIN-OS/2 is one of the components you can install to a drive other than the installation drive. You will see a list of features to install, as well as two radio buttons that allow you to configure your WIN-OS/2 Desktop or take the default. If you want to configure your WIN-OS/2 Desktop, you should click on the **Advanced Configuration** button.

Figure 1-18.
WIN-OS/2 Support Dialog (OS/2 Warp with WIN-OS/2)



WIN-OS/2 Support

WIN-OS/2 Features

Select the drive where files required for WIN-OS/2 support will be installed.

C Destination drive

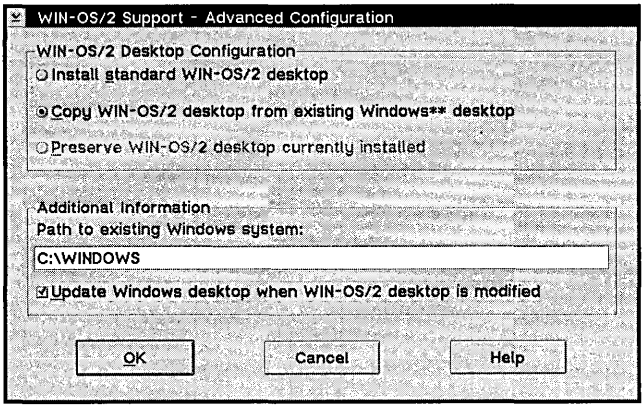
Make sure there is a check mark next to each WIN-OS/2 feature you wish to install.

☑ Readme files (136KB)
☑ Accessories (1039KB)
☑ Screen Savers (73KB)
☑ Sound (115KB)

WIN-OS/2 Desktop Configuration

● Automatic Configuration
○ Advanced Configuration

This will bring up the WIN-OS/2 Advanced Configuration dialog (see Figure 1-19). You have the choice of either using your existing Windows Desktop as your WIN-OS/2 Desktop or keeping your WIN-OS/2 Desktop the same as you had it prior to installing OS/2 Warp. Furthermore, you can keep your Windows Desktop in sync with your WIN-OS/2 Desktop — that is, whenever the WIN-OS/2 Desktop changes, appropriate modifications will be made to the Windows Desktop to keep the look the same — by checking the **Update Windows desktop when WIN-OS/2 desktop is modified** box.



WIN-OS/2 Support - Advanced Configuration

WIN-OS/2 Desktop Configuration

○ Install standard WIN-OS/2 desktop

● Copy WIN-OS/2 desktop from existing Windows** desktop

○ Preserve WIN-OS/2 desktop currently installed

Additional Information

Path to existing Windows system:

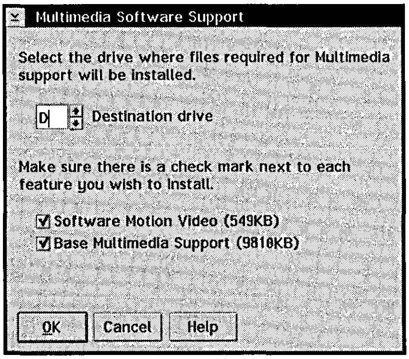
C:\WINDOWS

☑ Update Windows desktop when WIN-OS/2 desktop is modified

Figure 1-19. WIN-OS/2 Support - Advanced Configuration Dialog

Installing multimedia software support

This selection gives you the software support necessary to run OS/2 multimedia applications. You do not need to have a sound card in order to view software motion video; however, you do need to install the Base Multimedia Support if you want to view Software Motion Video or high-resolution digital movies. You can choose to install this material on a drive other than the installation partition if you wish. Use the spin box to select the drive you want to install multimedia software support to. Figure 1-20 shows you the Multimedia Software Support dialog.



Multimedia Software Support

Select the drive where files required for Multimedia support will be installed.

D Destination drive

Make sure there is a check mark next to each feature you wish to install.

☑ Software Motion Video (549KB)

☑ Base Multimedia Support (9810KB)

Figure 1-20. Multimedia Software Support Dialog

Installing high performance file system

Choose this option if you intend to format one or more of your partitions with HPFS. If you do not intend to use HPFS at all, you can deselect this option.

Installing serviceability and diagnostic aids

This option is very important in error determination. If you ever have a problem using OS/2 and you call on IBM to investigate the problem, generally the support person or consultant will use one of these utilities to determine the cause of the failure.

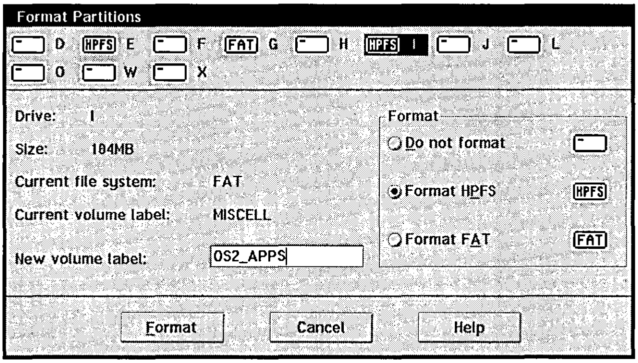
Installing optional bitmaps

When you select this option, you will get some bitmaps that you can use to change your background. Remember that displaying a bitmap does use some system memory. If you are happy with the solid background color, deselect this option.

Formatting drives other than the installation partition

You can format other drives while you are installing OS/2 Warp. To do so, click on the **Options** choice and select **Format.** The Format Partitions dialog, shown in Figure 1-21, will be displayed. All the drives the installation program can find will be listed. You can format any of these drives as either HPFS or FAT. As you select the file system to format a partition, you will see the bitmap of the drive updated to reflect the selection. You can also set the volume label for that partition.

Figure 1-21.
Format Partitions Dialog



Drive: I

Size: 104MB

Current file system: FAT

Current volume label: MISCELL

New volume label: OS2_APPS

Format

○ Do not format

● Format HPFS

○ Format FAT

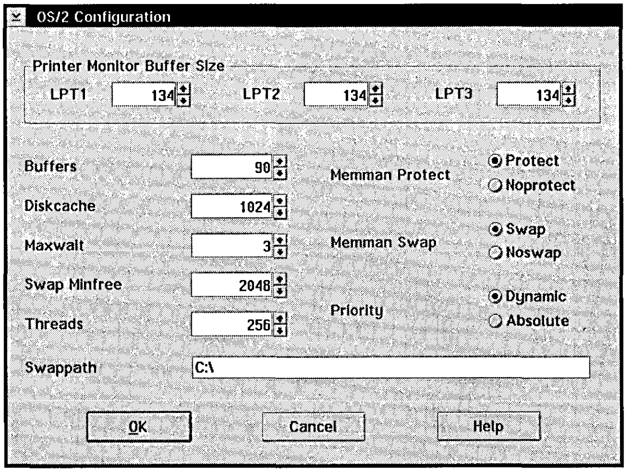
Bringing up a command prompt

You can start an OS/2 Windowed session in one of two ways. The easiest way is to press F3 at any time during the second phase of installation. The other way is to click on the **Options** selection on the OS/2 Setup and Installation dialog, then select **Command Prompt.**

Configuring software

OS/2 Warp installation gives you the ability to customize your OS/2 CONFIG.SYS settings and OS/2 DOS CONFIG.SYS settings. See Figures 1-22 and 1-23. Pay particular attention to the path where you want your OS/2 swapper file to reside (SWAPPATH). Make sure the drive you put it on has an ample amount of free disk space.

Figure 1-22. OS/2 Configuration Dialog



OS/2 Configuration

Printer Monitor Buffer Size

LPT1 134 LPT2 134 LPT3 134

Buffers 90

Diskcache 1024

Maxwait 3

Swap Minfree 2048

Threads 256

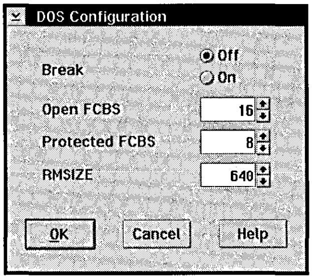
Memman Protect: ● Protect ○ Noprotect

Memman Swap: ● Swap ○ Noswap

Priority: ● Dynamic ○ Absolute

Swappath C:\

Figure 1-23. DOS Configuration Dialog



DOS Configuration

Break: ● Off ○ On

Open FCBS 16

Protected FCBS 8

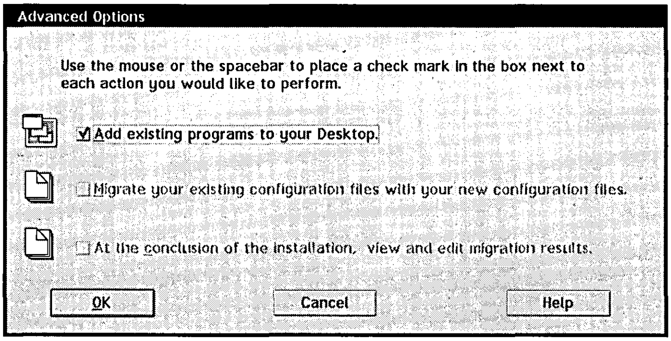
RMSIZE 640

Using advanced options

Click on the **Install** button when you are through making your software selections. This will bring up the Advanced Options dialog, shown in Figure 1-24. (If you are performing an Easy Installation, you will not see this dialog.) This gives you the ability to migrate your existing OS/2, DOS, and Windows applications to your OS/2

Desktop. The applications will be put into folders called OS/2 Programs, DOS Programs, and Windows Programs. If you find that some of your applications are not displayed, click on the **Add** button. You can also do the application migration after the OS/2 Warp installation is complete by clicking on the **Add Programs** icon in the System Setup folder.

Figure 1-24.
Advanced Options Dialog



Advanced Options

Use the mouse or the spacebar to place a check mark in the box next to each action you would like to perform.

☑ Add existing programs to your Desktop.

□ Migrate your existing configuration files with your new configuration files.

□ At the conclusion of the installation, view and edit migration results.

You also get two other options, both of which deal with migrating your old CONFIG.SYS and AUTOEXEC.BAT statements into your new configuration files. If you select the second option, your old CONFIG.SYS statements will be preserved. This is especially useful if you installed some support for an application that had CONFIG.SYS and/or AUTOEXEC.BAT modifications and you want to preserve these changes. The third option will show you the modifications that the installation program intends to make. At this point you can either make your own modifications or take the changes that are presented to you.

After you have made these selections, you are ready to start feeding diskettes. The file transfer process will be taking place as well as some configuration. If you chose Windows support and you are installing OS/2 Warp without WIN-OS/2, you will be prompted for the Windows diskettes after the file transfer is complete. Upon completion of this, you will be asked for printer and display driver diskettes.

Note

Keep in mind that if you chose the Easy Installation option in phase 1, you will get VGA support. If you want to upgrade your support to SVGA, click on the **Selective Install** icon, which resides in the System Setup folder.

Using the OS/2 Maintenance Partition

Uncensored

One of the most common problems that came across my desk was "I have a problem in installation at the very end, and now I have to start all over again." Well, the Workplace Shell team and I got together to see if we could come up with a software safe state during the installation process. First we had to come up with a place where you have a somewhat workable system and can do some error recovery or error determination. Next we had to make sure that this place or state was persistent (that it would always be there).

The result was what we call the Maintenance Desktop (this is the same as the beginning of the second phase of installation). So if you have a problem during the second phase of installation, or even if your system gets to a state where it is unusable, you can always get back to the beginning of phase 2.

Shhh! Tip

You do not have to go back through the somewhat painful process of preparing your hard disk. Perform the following steps to get to the Maintenance Partition:

1 Remove the diskette from drive A:.

2 Turn on your computer if it is not on, or press **Ctrl-Alt-Del** if it is.

3 When the white block is displayed at the top-left corner of your screen, press **Alt-F1**. This will bring up a menu of choices.

4 Press M for Maintenance Desktop. This will bring up the Maintenance Desktop.

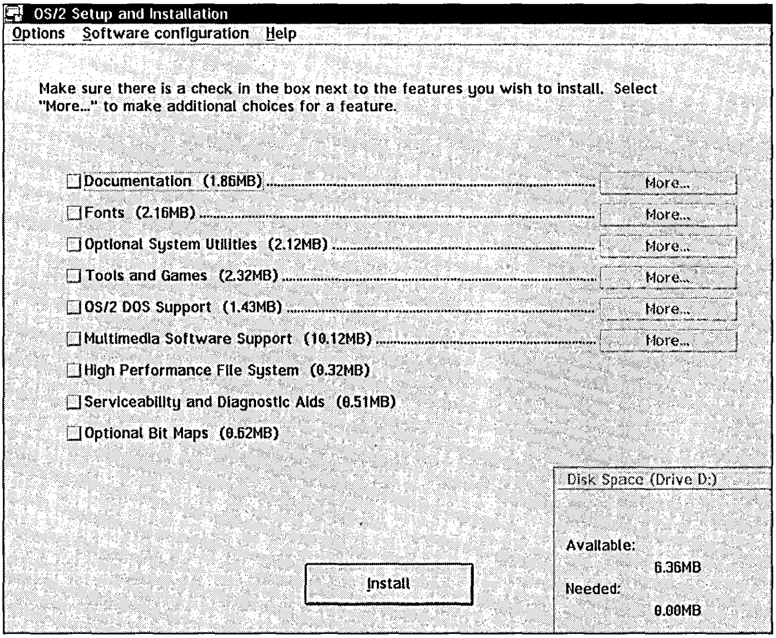
Response Files and Installing over the Network

Shhh! Tip

OS/2 Warp expands OS/2's support of response file installation. This topic could be an entire book by itself, and in fact it is. You can get excellent information in the IBM Red Book titled *OS/2 Installation Techniques: The CID Guide* or IBM Document GC24-4295-00. Simply put, for large accounts where someone has the responsibility of installing hundreds and sometimes thousands of systems, a quick and uninterrupted installation process is desirable. Combining network and response file installation gives the user precisely this ability.

Installing Features Selectively

If you decided not to install some features the first time around or you changed or added hardware, you can install the additional support that you want by double-clicking on the **Selective Install** icon in the System Setup folder. The first dialog you will see is identical to the System Configuration dialog. The next dialog that appears is exactly the same as the OS/2 Setup and Installation dialog that you saw the first time you installed OS/2 (see Figure 1-25). The only difference is that all the software features are deselected, and you must check the features that you want to install.


OS/2 Setup and Installation
Options Software configuration Help
More...
Install

Make sure there is a check in the box next to the features you wish to install. Select "More..." to make additional choices for a feature.

- □ Documentation (1.86MB)
- □ Fonts (2.16MB)
- □ Optional System Utilities (2.12MB)
- □ Tools and Games (2.32MB)
- □ OS/2 DOS Support (1.43MB)
- □ Multimedia Software Support (10.12MB)
- □ High Performance File System (0.32MB)
- □ Serviceability and Diagnostic Aids (0.51MB)
- □ Optional Bit Maps (0.62MB)

Disk Space (Drive D:)

Available: 6.36MB

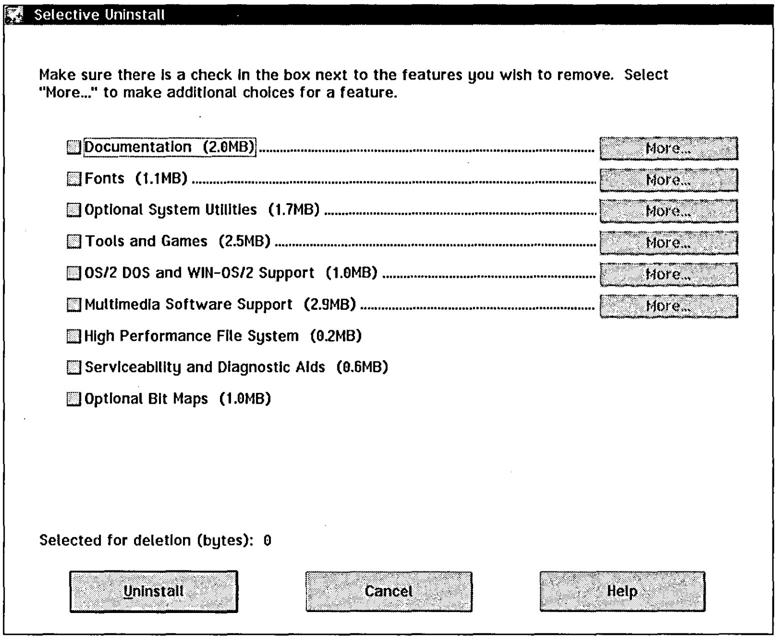
Needed: 0.00MB

Figure 1-25. Selective Install Dialog

Uninstalling Features

Similar to Selective Install is Selective Uninstall. You can remove support for any software feature that you installed (that is, anything selectable in the OS/2 Setup and Installation dialog). Simply check any combination of the features that you want removed on the Selective Uninstall dialog (see Figure 1-26). Keep in mind that you cannot uninstall hardware support.

Figure 1-26.
Selective Uninstall Dialog



Selective Uninstall

Make sure there is a check in the box next to the features you wish to remove. Select "More..." to make additional choices for a feature.

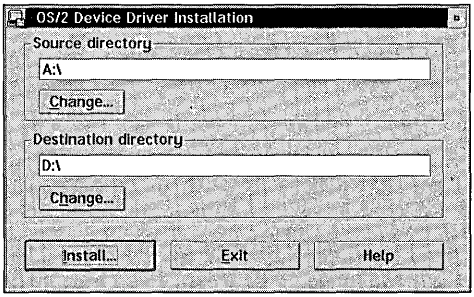
☐ Documentation (2.0MB)
☐ Fonts (1.1MB)
☐ Optional System Utilities (1.7MB)
☐ Tools and Games (2.5MB)
☐ OS/2 DOS and WIN-OS/2 Support (1.0MB)
☐ Multimedia Software Support (2.9MB)
☐ High Performance File System (0.2MB)
☐ Serviceability and Diagnostic Aids (0.6MB)
☐ Optional Bit Maps (1.0MB)

Selected for deletion (bytes): 0

Using the Device Driver Installation Program

OS/2 Warp still includes the device driver installation program, more commonly known as DDINSTAL.EXE (see Figure 1-27). This is a convenient installation program with which third-party hardware vendors can install their device drivers. DDINSTAL.EXE uses Device Driver Profile (DDP) files as the installation mechanism. Documentation on DDP files can be found in I/O subsystems in the device driver kit.

Figure 1-27.
Device Driver Installation Dialog



OS/2 Device Driver Installation

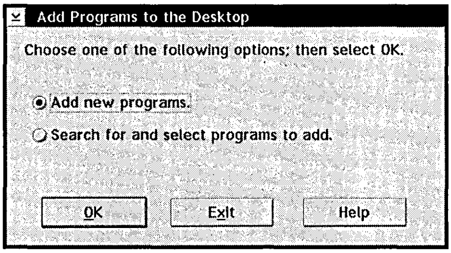
Source directory: A:\

Destination directory: D:\

Adding Programs to Your Desktop after Installation

I mentioned that the OS/2 Warp installation program will add program objects that you installed prior to installing OS/2 Warp to the Desktop. If you chose the Advanced Installation option, you were presented the option of adding program objects (icons) to the Desktop. If you elected not to install the icons or if you have installed additional software since you installed OS/2 Warp, you can migrate these objects to the Desktop by double-clicking on the **Add Programs** object in the System Setup folder. Simply click on the **Add New Programs** radio button and then click on the **OK** button (see Figure 1-28).

Figure 1-28.
Add Programs Dialog



Add Programs to the Desktop

Choose one of the following options; then select OK.

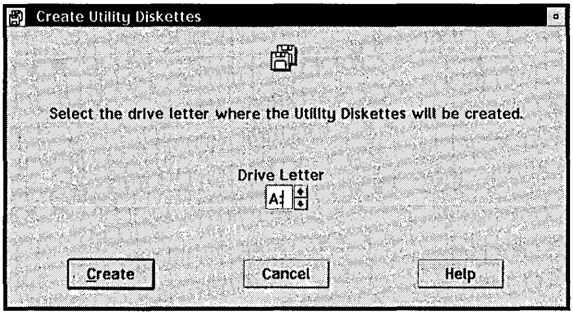
● Add new programs.

○ Search for and select programs to add.

Creating Utility Diskettes

You can use the Create Utility Diskettes feature to create three 1.44MB diskettes or one 2.88MB diskette. These diskettes are bootable OS/2 diskettes and also contain some useful tools such as FORMAT, CHKDSK, and other tools to back up and restore your entire system. There is also a README file copied to the utility diskette that contains some valuable information.

To start the Create Utility Diskette program, double-click on the **Create Utility Diskette** icon in the System Setup folder. Simply choose the drive where you want to create the diskettes, and then click on the **Create** button (see Figure 1-29).



Create Utility Diskettes

Select the drive letter where the Utility Diskettes will be created.

Drive Letter

A:

Figure 1-29. Create Utility Diskettes Dialog

Troubleshooting

The *User's Guide* that ships with OS/2 Warp is an excellent reference for determining the cause of installation problems (in particular, see chapters 14 and 15 of the *User's Guide*).

Following are a few troubleshooting tips that are not mentioned in the *User's Guide:*

What do I do if I get a fatal error (red panel) during the first phase of installation?

I have added some troubleshooting code into phase 1 of the OS/2 Warp installation. To get a more detailed explanation of an error during phase 1 of installation, follow these steps:

1. Insert the OS/2 Warp Installation diskette into drive A: and restart your computer.
2. Remove the OS/2 Warp Installation diskette and insert diskette 1 into drive A: when prompted to do so, and then press **Enter.**
3. At the Welcome screen, press **F3**, which will display a command prompt.
4. At the command prompt, type **COPY CONFIG.SYS DEBUG.SJK** and press **Enter.**

Now restart your installation process. You will still get the same fatal red error screen, but this time, when you press **Enter** to display the log file, you will see a more detailed explanation of the cause of the error. With this information, you may be able to fix the problem yourself; otherwise, you should call the OS/2 Technical Support Center and give them this information. It will be easier for them to diagnose the problem.

What do I do if I have a problem with my SCSI adapter or devices?

The OS/2 Warp installation program installs a set of SCSI autodetection programs into the \OS2\DRIVERS subdirectory on the installation partition. These programs all end with an .EXE extension. A simple way to find out if OS/2 Warp has correctly identified your SCSI adapter is to do the following:

1 Insert the OS/2 Warp Installation diskette into drive A: and restart your computer.

2 Remove the OS/2 Warp Installation diskette and insert diskette 1 into drive A: when prompted to do so, and then press **Enter.**

3 At the Welcome screen, press **F3**, which will display a command prompt.

4 At the command prompt, type **PROMPT ($R) [$P]** and press **Enter.**

   Before you see the directory location, you will see a number in parentheses. This number is the return code from the previous command that you typed in. You should see a 0 (zero) in parentheses at this point.

5 Type **D:**, where D: is the installation drive, and press **Enter.**

6 Type **CD \OS2\DRIVERS** and press **Enter.**

7 Type **DIR *.EXE** (This will give you a list of the SCSI autodetection programs that are shipped.) The list will look something like this:

```
3177  10-07-94   5:54p —  d:\os2\drivers\ibm16afs.exe
3035  12-01-94   5:39p —  d:\os2\drivers\ibm2scpr.exe
3263   3-02-95   3:14p —  d:\os2\drivers\fd8xx.exe
2801   3-02-95   3:12p —  d:\os2\drivers\fd7000ex.exe
3175   3-02-95   3:10p —  d:\os2\drivers\fd16-700.exe
3463  10-05-94  10:37p —  d:\os2\drivers\dptpres.exe
2925  10-03-94   1:56p —  d:\os2\drivers\btkpres.exe
3517  10-03-94   1:48p —  d:\os2\drivers\7870pres.exe
3141  10-03-94   1:46p —  d:\os2\drivers\7770pres.exe
2933  10-03-94   1:44p —  d:\os2\drivers\174xpres.exe
3245  10-03-94   1:43p —  d:\os2\drivers\154xpres.exe
3125  10-03-94   1:45p —  d:\os2\drivers\152xpres.exe
2595  10-03-94   1:33p —  d:\os2\drivers\tmv1pres.exe
2733  10-03-94   1:41p —  d:\os2\drivers\delivprc.exe
2773  10-03-94   1:43p —  d:\os2\drivers\164xpres.exe
```

8 You want to execute each of these programs until you see a 0 (zero) in the parentheses. If the adapter exists and OS/2 Warp recognizes it, then after you run its autodetection program, you will see a 0 in the parentheses. If it does not exist or OS/2 Warp has a problem recognizing it, you will see a non-zero number. Do not be alarmed if you see the screen black out and then come back because that is normal behavior for some of these programs.

After you get a 0, note the name of the program you ran and continue to step 9. If you did not get a 0, then OS/2 Warp did not autodetect the SCSI adapter that you have, in which case you should check to see if the adapter is listed in the book titled *OS/2 Hardware Compatibility List.* If it is listed, you need to call the IBM OS/2 Technical Support Center.

9 At the command line, type **CD \OS2\INSTALL** and press **Enter.**

10 Type **TYPE SCSI.TBL** and press **Enter.** You will see something similar to following on your display:

```
; This list is to be used to display the SCSI adapters to the user.
; The format will be :
;      <manufacturer model>;<presence check executable>;<NULL>
; The semicolon will be the delimeter between the fields.
;
; Need to add:
; IBM PS/1 or Valueline (Actual name for Future Domain adapter for IBM)
; IBM Int 13 driver.
;
Adaptec 1510,1520,1522;152XPRES.EXE;AHA152X.ADD;Prod-ID;None;
Adaptec 1540,1542;154XPRES.EXE;AHA154X.ADD;Prod-ID;None;
Adaptec 1640;164XPRES.EXE;AHA164X.ADD;Prod-ID;None;
Adaptec 1740,1742,1744;174XPRES.EXE;AHA174X.ADD;Prod-ID;None;
Adaptec 2840VL,2842VL,2740,2742,AIC7770;7770PRES.EXE;AIC7770.ADD;Prod-ID;None;
Adaptec 2940,2940W,AIC7870;7870PRES.EXE;AIC7870.ADD;Prod-ID;None;
BusLogic BusMaster SCSI Adapters;BTKPRES.EXE;BTSCSI.ADD;Prod-ID;None;
DPT PM2011,PM2012;DPTPRES.EXE;DPT20XX.ADD;Prod-ID;None;
Future Domain 845,850,850IBM,860,875,885,TMC
9C50/C950;FD8XX.EXE;FD8XX.ADD;Prod-ID;None;
Future Domain 16xx,1790,1795,MCS600/700,TMC
1800/18C30/18C50/3260/36C70;FD16-700.EXE;FD16-700.ADD;Prod-ID;None;
Future Domain 7000EX;FD7000EX.EXE;FD7000EX.ADD;Prod-ID;None;
IBM PS/2 SCSI Adapter;IBM2SCPR.EXE;IBM2SCSI.ADD;Prod-ID;None;
IBM 16-Bit AT Fast SCSI Adapter;IBM16AFS.EXE;FD16-700.ADD;Prod-ID;None;
ProAudio Spectrum 16 with Trantor SCSI;TMV1PRES.EXE;TMV1SCSI.ADD;Prod-ID;None;
```

Each field in this file is separated by semicolons:

- The first field is the description or name of the adapter.
- The second field is the name of its associated autodetection program.
- The third field is the name of the device driver for that SCSI adapter.
- The fourth and fifth fields can be ignored.

Find the name of the autodetection program you noted earlier in this file. Remember, it's the second field. Now get the name of the device driver (the third field). Note the name of this device driver. Now make a list of all the other device drivers in the third field of each line. You will REM these drivers out of the CONFIG.SYS on your hard drive.

11 Bring up the editor by typing **A:\TEDIT CONFIG.SYS** at the command prompt.

12 Find each BASEDEV=XXXXX.XXX statement for the list of SCSI adapters that you made (the ones that were not autodetected), and insert **REM** and a space character in front of each one.

13 After you have done this for each SCSI adapter from your list, make sure that the BASEDEV=XXXXX.XXX statement for the one that was autodetected does not have a REM note in front of it.

14 Save the file by pressing **F4.**

15 Remove the diskette from drive A:

16 Press **Ctrl-Alt-Del** to restart your system.

If you are still having problem with your SCSI adapter or SCSI devices you will need to call the OS/2 Technical Support Center.

Salil Kulkarni is a Staff Programmer and Technical Lead for the OS/2 Installation Department. He has been with IBM since 1989 and has worked on all versions of OS/2 System Installation from OS/2 Version 1.2 through OS/2 Warp. Salil also worked on the improved diskette compression techniques that have been used since OS/2 2.1.

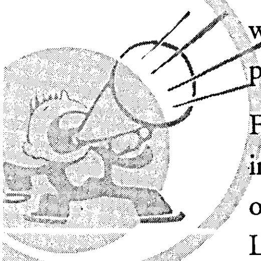
Chapter 2

The Workplace Shell Metaphor

Aha! The Desktop — More Clues Here!

Uncensored

Legend has it that someone at Black and Decker once remarked, "It is about holes and not about drills." What did he mean? Basically, if you want to know how to make a good drill, first find out what kinds of holes people need to make. The same principle was applied to the development of the Workplace Shell model. This focus on the users' tasks was the fundamental principle driving many aspects of the Workplace Shell design.



Even though it was a major paradigm shift, the development of the Workplace Shell was a direct response to customer feedback. It was the only way to solve the underlying problems that customers had been having with earlier versions of OS/2.

Figure 2-1 shows how the Workplace Shell appears after the first boot following installation of OS/2 Warp. By default, the LaunchPad, and OS/2 System folder appear open. During installation, I selected as my Desktop printer, the IBM 4019 LaserPrinter. I also selected Easy Installation, so my DOS and Windows programs were automatically installed and migrated and Multimedia was installed.

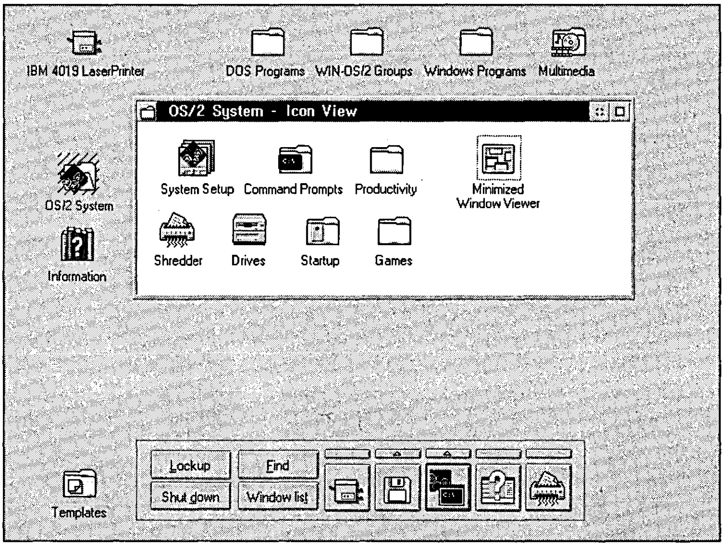


Figure 2-1.
The Workplace Shell in OS/2 Warp

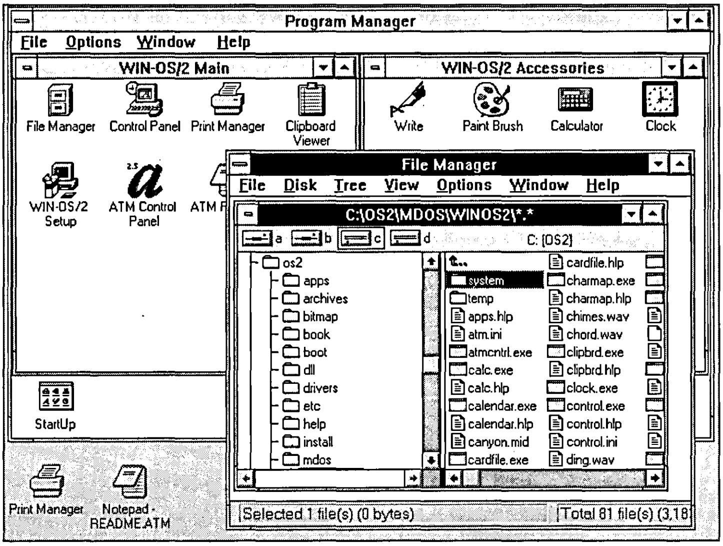
Why the Workplace Shell?

When IBM first delivered the Workplace Shell with OS/2 2.0, many of our customers asked us why we did not just fix the problems they were having with the old Program Manager/File Manager/Print Manager interface that existed in OS/2 versions 1.x. To see what some of those problems were, let's take a look at the old interface using the Win-OS/2 code that ships with the OS/2 Warp.

Figure 2-2 shows the following user interface elements:

- Two minimized windows (Notepad and Print Manager)
- A File Manager multidocument interface window with three subwindows filled with icons
- A Program Manager multidocument interface window with two open windows filled with icons and its own minimized window
- The Desktop or background

Figure 2-2. The Windows-OS/2 1.x User Interface



So what are the problems associated with this interface? The simplest way to highlight the problems is to ask you where the following tasks can be performed:

- ✗ Which windows can you drag and drop to?
- ✗ Which icons have properties notebooks attached?
- ✗ On which window backgrounds can you set a background bitmap?
- ✗ Which icons are used to execute a program? Which icons are used to display an already running program? If you were a novice end user, would you even know the difference between these two concepts?
- ✗ Is there any one window in which you can put your printer, programs, and related data files?

Well, if you are finding it all pretty confusing, so did our users. Although all the elements looked the same, in practice they were fundamentally different. You could not perform the same actions on similar-looking objects. So a group of User Interface specialists at IBM, in conjunction with the IBM Common User Access (CUA) group, decided that the only way to really solve the problem was to change the metaphor. The interface would have to be turned inside-out and sideways, but the end result would be a more consistent and easier-to-use interface. Every graphical icon would represent an object. It would be possible to perform similar actions on all objects. The windows in which the user could place those objects would, for the most part, not depend on the type of object.

As our group proceeded with the design of OS/2 Warp, we made sure that the Workplace Shell, as implemented, followed all the rules established by IBM's CUA group for user-interface controls. These rules helped to ensure that the user would be presented with a consistent and easy-to-use interface. This book presents a subset of the CUA guidelines, but for full documentation of IBM's user-interface guidelines, I recommend *Object-Oriented Interface Design, IBM Common User Access Guidelines.*

What Is the Workplace Shell model?

Now, you may be wondering why I want to spend all this time explaining the Workplace Model in all its gory detail rather than just getting on with the business of telling you how to do all the cool stuff. If you can understand what our goals were in creating the Workplace Shell model, then you can understand why we implemented things the way we did and how they fit together, so that you can better utilize the features.

As I guide you through the complete description of the Workplace Shell model, I will point out the design decisions supporting that principle. But I want to highlight a few of the more important ones now.

Uncensored

The principle underlying the design of the Workplace Shell is that the interface is object-oriented and graphical. Furthermore, the interface is data-driven instead of program-driven. Our design goal was to allow users to focus on the tasks they want to perform and not on which application they use to perform those tasks. This goal is the impetus for OS/2 Warp features such as pop-up menus, the Templates folder, the graphical user interface, and data-driven program launch.

Pop-up menus

Pop-up menus (see Figure 2-3) allow you to stay focused on your task without having to look through a menu bar structure that may be visually very far away from the work at hand. Also, a pop-up menu is a context menu, which means that all the choices on the menu are available given your current context (that is, the part of the document that you are currently working on). You will not be presented with choices that are not available or do not apply.

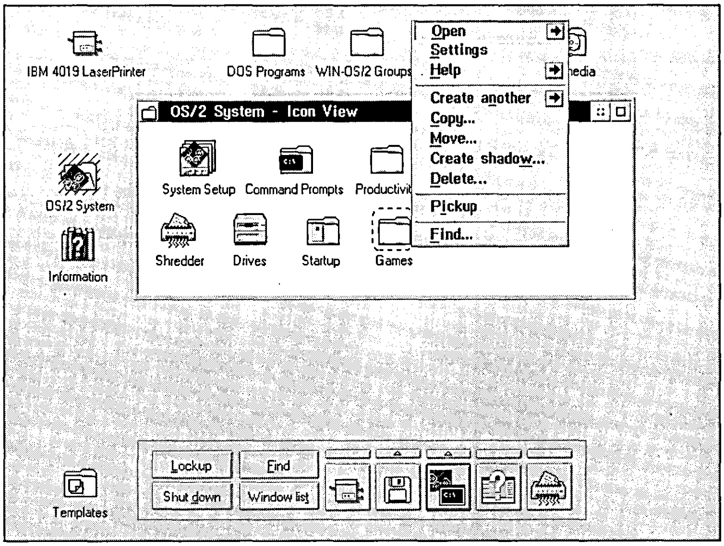
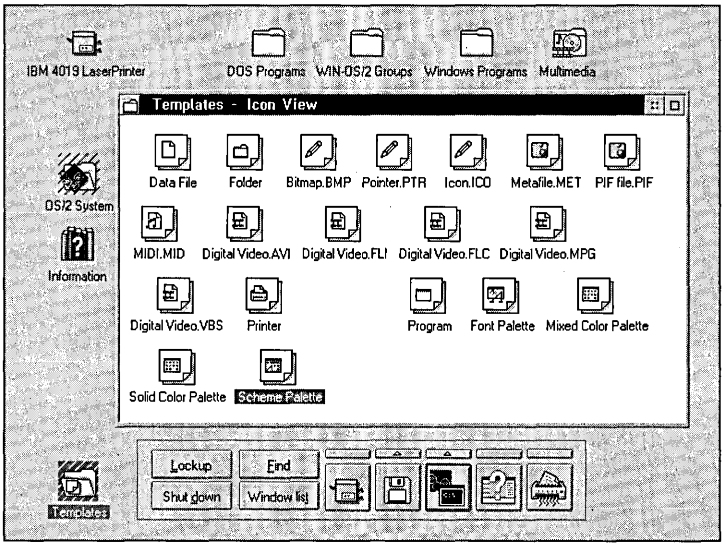


Figure 2-3.
A Pop-Up Menu of the Games Folder

Templates

When you perform a task using the computer, your first thought is about what you want to accomplish, not how you want to accomplish it. You may want to write a letter or a book. Or you might need to calculate a mortgage payment for that new house you want to buy. You may even have to prepare an important presentation. Under DOS and Windows most users make the leap in a nanosecond to the program they need to start in order to accomplish that task. But this is only after they've been trained to do so. What if you could just get a fresh copy of the type of document you wanted to work with and the application that supported that document came along like magic? You would then be working at the computer the same way you were thinking. Templates, therefore, provide a place to get a clean version of the type of file you want to work on (see Figure 2-4).

Figure 2-4.
The Templates Folder



Graphical User Interface

On systems that support both a graphical user interface such as the Workplace Shell and a command-line interface, the system default will be for the graphical user interface to be displayed. Open windows may also be displayed. For example, a Getting Started tutorial window might be displayed immediately following installation.

The start-up configuration of the Workplace Shell contains icons representing objects presented to you in a set of windows called folders. On systems like OS/2, which supports both a graphical user interface and a command-line user interface, the graphical user interface must be the shipped default. Furthermore, some objects may be prestarted or preopened, such as the tutorial and the OS/2 System folder in OS/2 Warp.

This design is not limited to OS/2; it is a broad design that can be used across operating systems.

Hidden Programs

Programs appear only in views of folders and can be started only from those views. Although users can place programs on the Workplace Shell Desktop, the default Desktop does not contain programs.

Uncensored

In our data-driven view of the world, programs (that is, stand-alone executables) are not the primary focus. Actually, we would prefer that users forget all about the concept of programs so that they can be free to concentrate entirely on their data. That is why we did not want to see programs appearing directly on the Desktop. You can find them buried in folders somewhere, but the system should not ship with programs "in your face." However, there is no restriction on your ability to place a program anywhere you like after the system has been installed and booted the first time.

Associations between Data Objects and Programs

One of the most important components of the Workplace Shell model is the concept of *associations*. Associations between data objects and programs exist for all user-accessible data objects in the default configuration. For example, .TXT, .BAT, and .SYS files are associated with the OS/2 System Editor.

In a perfect world and in a perfectly implemented Workplace Shell-based application, you would never find an executable program. The application would be implemented as a Dynamic Link Library (DLL) and would have registered itself with the Workplace Shell as a Workplace Shell extension, also known as a subclass.

However, we do not live in a perfect world. In fact we have a world populated with legacy applications for DOS, Windows, and OS/2. So the problem is to how to get something useful to happen when you attempt to open your data files that were created by one of these applications.

Uncensored

We use the concept of associations to link groups of data with the applications needed to process that data. Data may be grouped in one of the following ways:

- **By file extension (for example, .TXT or .DOC)** — You must include the file extension in the title of a Workplace Shell data object so that association by extension will continue to work.
- **By file type** — File types are used to describe a data file object according to the type of application that created it. For example, Lotus 1-2-3 spreadsheet or Ami Pro document types are added to the system when you install the corresponding applications. You can then link those file types to documents that already exist so that those documents can be processed by the appropriate applications.

- ✗ **By Workplace Shell class** — Data file documents may belong to a specific Workplace Shell class, such as Bitmap, Icon, or Pointer, which correspond to the Workplace Shell classes WPBitmap, WPIcon, and WPPointer, respectively. In this case the application developer (sometimes known as the class developer) maintains more control over the object to ensure that it is processed by the correct application or that processing can be done internally within the class handler. A class handler is the DLL the class developer writes instead of an application to provide Workplace Shell integration.

Default associations

We want something useful to happen when you attempt to open any object in the system, so we set up a *default association.* Unless another association is found, the object will be opened into the OS/2 System Editor. The default association assigns all Workplace Shell objects that are specifically of the Data File (WPDataFile) class an imaginary file type of Plain Text. I say imaginary because we do not physically write a Plain Text file type for the object but just treat the file internally as if it has this type. Since the OS/2 System Editor is associated with data file objects of the Plain Text file type, all default objects without a specific association default to the Plain Text type. If you want to change the default association from the OS/2 System Editor to another editor, such as the Enhanced Editor, do the following:

1. Remove Plain Text from the Association page of the OS/2 System Editor object's Settings Notebook (see Figure 2-5) by selecting the Plain Text entry in the Current types list box and pressing the **<<Remove** push button.
2. Add Plain Text to the Association page of the Enhanced Editor object's Settings Notebook (see Figure 2-6) by selecting the **Plain Text** entry in the Available types list box and pressing the **Add>>** push button.

If you want to keep the OS/2 System Editor as a choice under the Data File object's Open menu, then add back Plain Text to the Association page of the OS/2 System Editor after you have performed the two steps outlined above. Follow the instructions in Step 2 above, but on the OS/2 System Editor instead of the Enhanced Editor. Yes, that means you are removing it and then adding it back.

Figure 2-5.
OS/2 System Editor Association Page

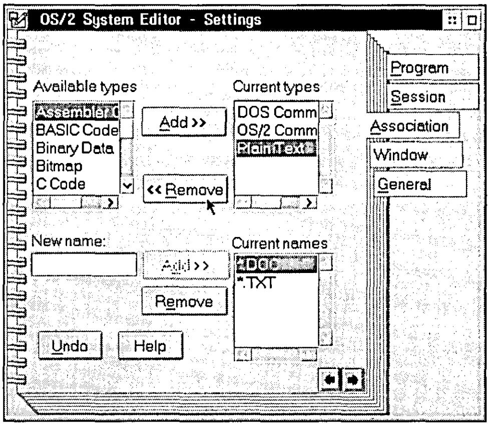
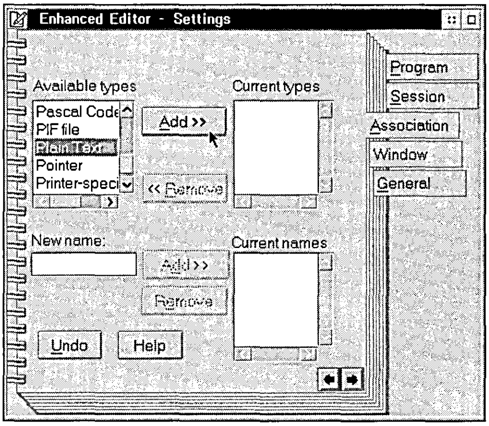


Figure 2-6.
Enhanced Editor Association Page



Establishing and changing associations

A mechanism is provided to allow applications to establish and change associations — during application installation, for example. We certainly would not want to burden our users with having to set up the associations for every application by hand. A mechanism is provided for applications developers to set up these associations during the installation of their applications. The effect of this is that Data File objects for those applications that you find in a folder or on your Desktop should open right into the applications without any additional effort on your part.

When you install an application, you may notice that additional types are added to your system, supporting your ability to associate currently existing data file objects with the newly installed application. For example, if you install Lotus 1-2-3, you may find a Lotus 1-2-3 type added to your system. You can then add the Lotus 1-2-3 type as the first type in the Current Types list box of the Data File notebooks for other spreadsheet data files, including those files that can be read by Lotus 1-2-3 but were not created in Lotus 1-2-3.

Establishing and changing associations through the user interface

Uncensored

A user interface is provided to allow users to establish and change associations. Users might need to change associations for the following reasons:

- ✗ To allow an existing Data File object to work with a new or different application
- ✗ If they want data files of an existing application to open directly into that application when opened from a folder or the Desktop and the application did not set up this association on the user's behalf

A more in-depth discussion of the practical use of associations is provided in the section on Associations in the Workplace Shell later in this chapter.

Printing Data File Objects

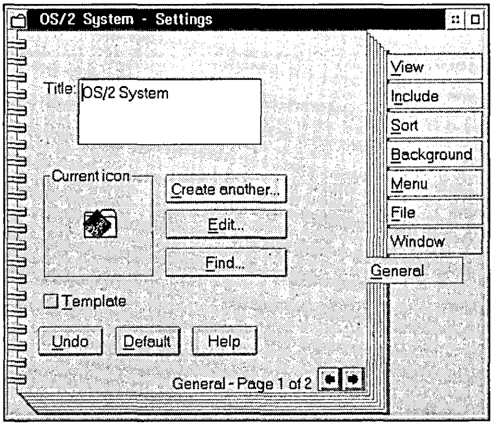
As you may know, most applications, such as word processors, spreadsheets, or presentation graphics, store their data files in a binary (nonreadable) format. Naturally we would like you to be able to print these documents by dragging their icons from your Desktop or folder directly to your printer icon. However, since the Workplace Shell does not understand the internal representation of the data, it does not know how to print it for you. The association mechanism that we provide for users does not entirely solve this problem.

Uncensored

The architecture of the Workplace Shell requires that application developers provide a Workplace Shell class for their application to intercept the print request so that the application developer can provide a printing routine. One of the finest examples of this feature is WordPerfect's Workplace Shell Integration Tools for WordPerfect 6.0 for Windows. I encourage you to request this function from the providers of your favorite applications. It does not matter if the application is DOS-, Windows-, or OS/2-based.

Settings Notebooks

The pervasive properties of any object are contained in a *Settings Notebook* that is accessed from the object's pop-up menu. The idea behind the Settings Notebook concept is to reduce the frustration you feel when you want to browse through an object's properties or set more than one piece of information about an object (see Figure 2-7).



Title: OS/2 System

☐ Template

Figure 2-7. A Settings Notebook Page for the OS/2 System Folder.

Think about the applications you use today that have no equivalent to the Settings Notebook. If you want to make a change, you have to first traverse one or more levels of menus and then open a single dialog. You either view or change that one item in the dialog and then close the dialog. You have to repeat these steps for every item you want to view or change. This is much too cumbersome and inhibits exploration.

Because the information about an object in a Settings Notebook is collected into one set of interrelated dialogs or windows, you are better able (encouraged, actually) to explore all the information and make multiple changes at one session with the Notebook.

The Settings Notebook also provides a certain level of consistency by introducing the concept of a standard view for all objects. In other words, for any object you find either on your Desktop or in a folder, you can be assured that there is at least a Settings Notebook view. Even though the Settings choice has been moved out of the Open menu because of overwhelming user requests, we User Interface purists still consider the Settings to be a view of the object.

The following options are available when you're using the Settings Notebook:

✗ **Undo** — Pressing the Undo push button causes all fields on that Settings Notebook page to change back to the state and values they had when that Settings Notebook page was first opened for that session. It also causes the page to be marked as not having changed so that the information pertaining to that page does not have to be written out to persistent storage.

✗ **Default** — Pressing the Default push button causes all fields on that Settings Notebook page to change to the default state. The side effect of the default feature is that the object does not have to save this information to its persistent storage. One of the important features we need to provide in the future is a way for the user to discern which items of information on that page are being inherited and which items have a value that was actually set by the user.

✗ **Help** — Every Settings Notebook page provides a Help push button to describe that page. You can also access help by pressing **F1**. You may wonder if there is a way to get field-specific help. Currently, the answer is no. Since the vast majority of users want a general overview of the function of the Settings Notebook page, the Overview page provides a hypertext link for users to access information about specific fields.

The Workplace Shell Settings Notebook pages operate under the philosophy of *perfect save.* This philosophy says that the user does not have to take a secondary action to cause the property change to take effect. For example, if I change the folder background from one color to another, I see the effect immediately. I, as the user, am not required to press a second button, such as OK or Apply, to cause that change to take effect.

The fact that the change takes effect immediately does not necessarily mean that the persistent state of the object has changed. In most cases, the object state is not saved until the Settings Notebook is closed. Even then, the work is placed on an internal queue to be done when the system is not so busy.

Uncensored

The rationale behind this delay is to encourage you to freely explore the settings. We want you to experiment with the values of your Settings Notebook. Since the changes are not saved right away, you are not burdened with waiting for the system to write out the changes each time you change a value. Also, you may decide after all of your experimenting that you liked the original values best. In that case, you would press the Undo push button and the original settings would be restored.

Note

However, there are situations in which perfect save would have such a negative impact on overall system performance that it would be counterproductive. In those cases, you may well find an Apply push button on the Settings Notebook page. An example of this is objects that represent information stored on a LAN-based server or perhaps on a mainframe host machine. Even though these systems may not have to save the information, they still have to communicate the change. And because of the hardware implications of sending that information over a LAN line or a modem connection to a mainframe host, you would want to know that your action will trigger this time-consuming event.

System component Settings Notebooks

Some Workplace Shell objects correspond to the physical components of a user's computer. The properties of system components, such as the keyboard, mouse, and display, are accessed from pop-up menus of objects that represent them. Users can change the characteristics of their hardware via Settings Notebook pages for those objects.

This setup ensures consistency, since you can always go to a Settings Notebook to change information about an object. It also allows you to have fast paths to the most common settings directly on the pop-up menus of those objects. For example, the Power object lets you set the Refresh state to on or off directly on the pop-up menu as well as in the Settings Notebook.

Standard Objects and User Interface Controls

In addition to the Workplace Shell Desktop, which is aWork Area folder, several standard objects are supplied. They may be either ready-to-use samples or templates allowing users to create their own objects.

Folders

A folder is a file system object that contains other objects. A folder is always a file system directory. One important point here is that a folder, unless it has been specialized, can contain any object whatsoever. This solves the major problem of users who are used to working with Program Manager, File Manager, and Printer Manager, each of which displays what looks to the user like a window full of objects, but is set up in such a way that the user has to remember which kinds of objects can go in which window. Under the Workplace Shell model, an object is an object and any folder can contain any object, including another folder.

System folder

A System folder provides a collection of objects that are either used to configure the system or specific to the operating system. These objects are different from the Workplace Shell model objects, which are always on the Desktop. In addition to the Startup folder, the System folder also contains objects specific to OS/2 Warp, and those objects are discussed in detail in Chapter 9.

Startup folder

The Startup folder is identical to any other folder except that all of its objects are opened when the system is started.

This is different from a STARTUP.CMD file in that the STARTUP.CMD file contains only applications, and those applications are started in a synchronous manner, unless the START command is used within STARTUP.CMD. The Startup folder can open any object as well as open objects to their default views. You change the default view for an object via the Menu page.

Warning

In most cases, you will want to put *shadows* (pointers) of objects into the Startup folder rather than the objects themselves. This is important. Otherwise you will end up moving or copying data or programs to the Startup folder from their original locations, which may have been on a different drive. Often this will cause the application to run improperly, if at all.

For Program objects, however, it is safe to make a copy or to even move the Program object into the Startup folder. Since a Program object is just a pointer (or reference) to the executable program, you are not at risk of causing problems with your application or accidentally moving a large amount of data to your Startup folder directory.

Information folders

The Information folder on the Desktop provides system reference information. This will be discussed in more detail in Chapter 9.

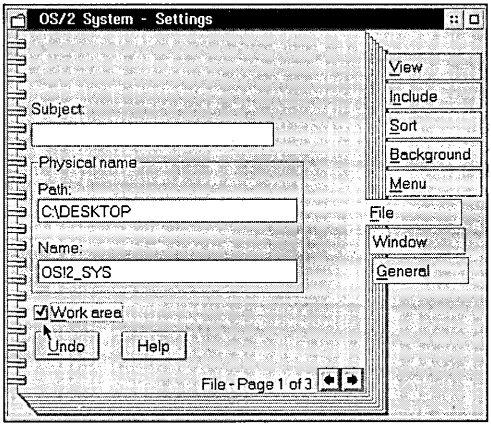
Work Area folders

A Work Area folder has the property that all windows opened from objects on the Work Area

- Are closed when the Work Area is closed
- Are reopened when the Work Area is reopened
- Are hidden when the Work Area is either hidden or minimized

The Workplace Shell Desktop is a Work Area folder. It is the only folder preconfigured as a Work Area in OS/2 Warp. You can turn any folder into a Work Area by turning on the Work Area attribute in the folder's Settings Notebook pages (see Figure 2-8). You cannot change the Work Area attribute of the Desktop.

Figure 2-8. Setting the Work Area Attribute for a Folder



OS/2 System - Settings

Subject:

Physical name

Path: C:\DESKTOP

Name: OS!2_SYS

☑ Work area

File - Page 1 of 3

A typical use of Work Areas is grouping objects and their open windows by the type of task they perform. When you need to switch your attention from one task to another, you simply minimize, hide, or close the Work Area folder to put aside (or put away) all the other windows associated with that Work Area until you are ready to bring your attention back to it.

Tip

Another possible example of the use of Work Areas is to provide a simple serial multiuser capability (that is, when more than one person uses the same machine but not at the same time) to your OS/2 Warp setup. You might create Work Area folders on your Desktop that represent each of the users of your system. You can then move objects from the Desktop to the Work Area of one of the users, and you can create shadows to the high-level folders or other specific objects in the Work Area(s) of the other users. Now each user for all practical purposes has his or her own Desktop.

This example demonstrates the utility of shadows. You can have what appears to be the same object in more than one place at the same time without actual copies of the object in all those places consuming storage. The storage size of a shadow object in the OS2.INI file is negligible.

The Shredder

What's the difference between a trash can and a shredder? You can retrieve things from a trash can; once you put something through the Shredder, it is gone forever. Hopefully we will be able to provide a trash can in OS/2 in the future, but for now all we have is the Shredder, so be careful. The Shredder object is discussed in Chapter 9.

Printer objects

A Printer object provides the user with an obvious drag-and-drop target for initiating a print request, for displaying a set of print job objects that are scheduled to be printed, and as a means of customizing the options of the printer device. Printing and the Printer objects are covered in full detail in Chapter 14.

The LaunchPad

In OS/2 Warp, the front panel that is required in the Workplace Shell model was named the LaunchPad. We developed the LaunchPad to provide a convenient place for users to gain fast access to their most commonly used objects and the most common Desktop actions. The LaunchPad is discussed in detail in Chapter 4.

Color, font, and scheme palettes

Uncensored

Because the designers of the Workplace Shell model wanted to encourage you to explore your system, they felt it was important at the architectural level to provide a way for you to make the system feel like your very own. For this reason a set of palettes is provided, allowing you to customize your system using the same metaphor (drag and drop) that is used to perform other actions in the system.

Since the color or font change becomes a property of the object that was changed, it is also possible to change that property from the Settings Notebook of the object. Although we sometimes receive complaints that the Workplace Shell offers more than one way to do similar things, I hope you can see that we did it that way to provide consistency with the Workplace Shell model. That way, the user is free to use the technique that works best for him or her.

Help

Every Workplace Shell object needs to present three types of help about itself.

1. The first type of help is referred to as *closed object* help. This is the help you receive when you press the F1 key with an object selected or choose the General Help menu choice for the object. This help gives you a general overview of the purpose and function of the object in question.

2. The second type of help is referred to as *menu choice* help. The user receives a specific help description of the highlighted menu choice when the user presses F1 while in that object's pop-up menu.

3. The third type of help is referred to as *open view* help. This is the help you receive about an object when you are in an open view of that object. This help is more detailed and gives specific information about how to work with the open view of that object. For example, the open view help for a spreadsheet object would tell you how to function in the spreadsheet and provide information about all the features available to you. Another example of this type of help is the help available on each of the Settings Notebook pages that discusses the functions available to you on that specific page.

The Getting Started tutorial

In my world of User Interface design, I have to deal with the conflict between "easy to learn" and "walk up and use." Our goal is to make the system as easy to use as possible, but "walk up and use" is the holy grail of User Interface design, and we do not think we are there yet. So instead, the design point for the Workplace Shell was "easy to learn."

The phrase itself implies, obviously, that users have to learn something before they can productively use the system. All of our studies have indicated that we have 10 to 20 minutes to teach the user about the system with a tutorial. So that is enough time to teach the user the basics of how to get around in the Workplace Shell and discover objects, help, and menu items. But the fact remains that we still expect that the user will have done at least this much before using the system.

Templates

The design of the Workplace Shell requires that we provide representative objects, or templates, from several different object classes. The user can create a new empty object of a particular object class from the template of that class.

By collecting these objects in a Templates folder, we have provided a place from which the user can always get a new object. Remember that in our view of the world, the user does not start an application to create a new document; the user gets a new document from the templates folder and that document starts the application.

With the introduction of the Templates folder, we also introduce a new drag metaphor called *create on drag.* This means that the new object dragged from a template is neither copied nor moved. Instead, a new operation is carried out, called *create from template.* This gives the application developer a chance to do special processing on the creation, such as prompting the user for additional information.

Pop-up menus

I mentioned earlier that one of the most important aspects of the Workplace Shell model was allowing users to stay focused on their work. Our goal was to bring the menu choices to where the user's attention is focused. Also, we wanted the menu choices to be relevant to the current state (or context) of the user's work. That is why pop-up menus are also referred to as context menus. We made sure you wouldn't find grayed-out items on the pop-up menu; these options are best left to the menu bars, where users can have access to them when needed.

To be useful, pop-up menus must meet several requirements:

- ✗ The user must be able to get to all views of the object.
- ✗ The user also must be able to perform the typical object manipulation functions such as Copy, Move, and Create another.

- The user should be able to access from the pop-up menu actions specific to an object, such as Refresh on the Folder menu.

Characteristics of Standard Objects

All standard objects have a Settings Notebook. At the very least, the Settings Notebook must provide the user with the ability to view and modify that object's title and icon.

Uncensored

To prevent users from getting lost in a sea of icons, every class of objects has a unique icon associated with it to help users distinguish between objects. For example, Ami Pro document icons should look different from Lotus 1-2-3 spreadsheet icons even though they are both subclasses of Data File objects. Each class of objects must also have a unique title, because there are several places in the Workplace Shell where the user is asked to pick from a set of objects by the default object title for a class of objects.

The icon and title also need to be unique for each class of objects because the templates folder automatically generates a template object for each class registered to the system. (A few classes do not need templates; thus, the system does not generate templates for every class.) It would be very confusing to find two or more different objects in the Templates folder with the same icon and title. Similarly, confusion would also result if multiple classes had the same title.

Direct manipulation of objects

All standard objects support drag-and-drop for Create, Move, Copy, Create shadow, Print, and Discard operations. To really make the system easy to use, we provided a well-defined set of mouse-related operations because you can perform them much quicker than menu-based operations. Some of these direct manipulation operations are self-contained, including Copy, Move, and Create shadow. Others require a special source or a special target. For example, in the case of Create, a special source, namely a template, is required. For Print and Discard, you must have a special target, known as a Device object, to drag to in order to complete the operation.

Views of objects

The following is a set of rules that govern how objects should present views of themselves.

Views for each object type

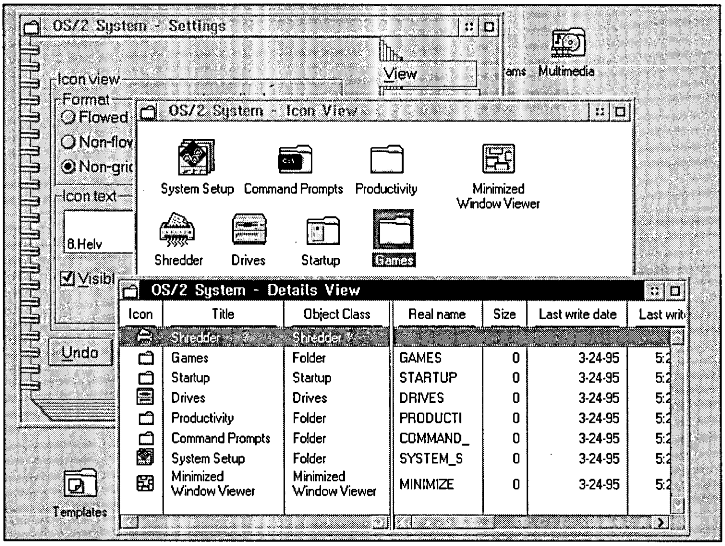
The standard objects provide multiple views of themselves. Other objects must provide at least a Settings view. Remember that the Settings Notebook is considered a view even though you navigate to it through a choice on the primary pop-up menu and not under the Open conditional cascade menu. So objects like data files that open into a view of a running program, like the OS/2 System Editor, still have a second view, which is the Settings Notebook view. For objects that perform a setup-type functionality, you will routinely find just a Settings Notebook view.

You will also notice that all open views of an object are named corresponding to the menu choice that you used to open them. That is why, for example, the LaunchPad appears in the Window List as LaunchPad - Palette. You can also open up the Settings view of the LaunchPad, and it will be named LaunchPad - Settings.

A neat trick is to keep a Details view of the Desktop always open so that you have a view of the Desktop that you can bring to the front of all the other windows. This gives you fast access to your Desktop objects.

Multiple views

Multiple views can be opened concurrently in different windows or consecutively within the same window (see Figure 2-9). The minimum requirement is to resurface the same type of view if one is already open. As a system option, a user preference can be provided to open additional views of the same type.



OS/2 System - Details View

Icon	Title	Object Class	Real name	Size	Last write date	Last writ
	Shredder	Shredder				
	Games	Folder	GAMES	0	3-24-95	5:2
	Startup	Startup	STARTUP	0	3-24-95	5:2
	Drives	Drives	DRIVES	0	3-24-95	5:2
	Productivity	Folder	PRODUCTI	0	3-24-95	5:2
	Command Prompts	Folder	COMMAND_	0	3-24-95	5:2
	System Setup	Folder	SYSTEM_S	0	3-24-95	5:2
	Minimized Window Viewer	Minimized Window Viewer	MINIMIZE	0	3-24-95	5:2

Figure 2-9. Multiple Views of the OS/2 System Folder

It is perfectly acceptable to have more then one view of an object open at the same time. You can have a folder open to its Icon view and also display its Settings view. You may also have an Icon and a Details and/or Tree view all open simultaneously.

Figure 2-2, at the beginning of this chapter, shows the Program Manager interface. You can see the Print Manager icon in two places. One is inside the WIN-OS/2 Main Group window and the other is in the bottom-left corner of the Desktop. Quick: What is the difference between the two? Well, as a power user you already know that the one inside the WIN-OS/2 Main Group window is a program icon and the one in the bottom left-hand corner of the Desktop is a minimized window of an already running Print Manager program.

This example illustrates the type of problem we faced using the similar interface in OS/2 1.x. Our novice users could not make the distinction, so they just double-clicked on the first icon they found that looked like what they wanted. In the majority of cases, that would be the icon inside the Group window. The end result was that the user would launch multiple copies of the application without knowing it. The user only wanted to launch the application if it was not already running; otherwise, they only want the application to come to the surface.

Uncensored

So that is what we did with the Workplace Shell. By default we changed the "put aside" visual from a minimized window to a hidden window. Then we changed the definition of the Open operation to first attempt a resurface of that view if it is open, and only then to open the view. You can change this behavior with options in the object's Settings Notebook. For OS/2 Warp we changed the default on the command prompt objects to always open a new instance since that is the expected behavior for those types of objects.

Displaying updates in multiple views. Updates are displayed simultaneously in multiple concurrent views of an object. This ties in with the idea of perfect save that I discussed previously, that is, that all changes appear to take effect immediately. If you have multiple views of the object open and they each present the same information but in different formats (i.e., an Icon and Details view of a folder), you do not want to have to keep track of which view you made a change in, which view you are currently in, and whether or not a view needs to be refreshed. All changes should be reflected immediately in all open views of that object.

Container object views

In addition to the Settings view and help information that all objects must provide, objects that hold other objects (i.e., container objects or folders) must also provide three standard views:

- ✗ **Icon** — In Icon view, all objects are represented by their object icon and title. Optionally, one or the other may be turned off so that you are displaying just the icon or just the title.
- ✗ **Details** — In Details view, all objects are displayed in a columnar format. At a minimum, you will find the object's icon and title. Additionally you may find other information about the object that you would typically have to go into the Settings Notebook to find. This allows you to see that information all at one time.

  In OS/2 Warp the folder is configured to show a file system level of information about objects. You can change that configuration to see details of other object classes that may have been provided separately.
- ✗ **Tree** — The Tree view is provided so that you can see a hierarchy of container objects (that is, what other containers exist and which containers those containers contain).

Shadow objects

All standard objects support multiple occurrences of their icons. In OS/2 Warp the ability to add multiple occurrences of an icon is provided through the Shadow object class. A *shadow* is an object that allows you to access an object from multiple folders without having to put a copy of that object in each folder. Instead, a shadow serves as a pointer that, when called upon, simply tells the system where to find the original object.

Developer's Desktop

In my own personal configuration, I keep all my applications on a different physical drive from my OS/2 Warp boot drive. I keep shadows of my application data directories on my Desktop. The objects in the LaunchPad are also Shadow objects.

Tip

You can use shadows to help group your work. If you are working on a particular project, you may wish to create a project folder that contains shadows of all the objects you need to work on that project, such as your printer, your data files, and maybe even the Program object. If you use those same objects in different projects or other contexts, you do not have to move them from their original locations, and you will not have to remember where they are every time you change the project you are working on.

Standard visual and interaction paradigms

Uncensored

As I mentioned earlier, the design of the Workplace Shell was greatly influenced by the CUA guidelines for interaction. Whether you are using the Workplace Shell or an ordinary Presentation Manager program, you want to be able to perform similar actions or be presented with similar visual clues that have the same meaning across the operating system. Therefore, we decided that the Workplace Shell model would encompass and use the following CUA items:

- Emphasis
- Cursoring/navigating
- Selection
- Direct manipulation
- Mouse mappings
- Keyboard mappings
- Text editing
- Contextual help

Window features

The graphical user interface of OS/2 presents you with many types of windows. CUA has established some rules about what types of window the user should be presented with and under what circumstances these windows should be presented.

In our design of the OS/2 Warp interface we have endeavored to follow the standard CUA window usage guidelines for views of objects, action options, and messages, for example. Following is a discussion of some of the window features and why we created them the way we did.

Overlapping windows. Think about the messy desk metaphor: You want to have more work open and in progress than you can possibly see at one time, but you want to be able to arrange and order your work so that you can get at it.

We created windows that overlapped so that you could have all your work out on the Desktop but would be able to easily bring the current work to the top of the pile.

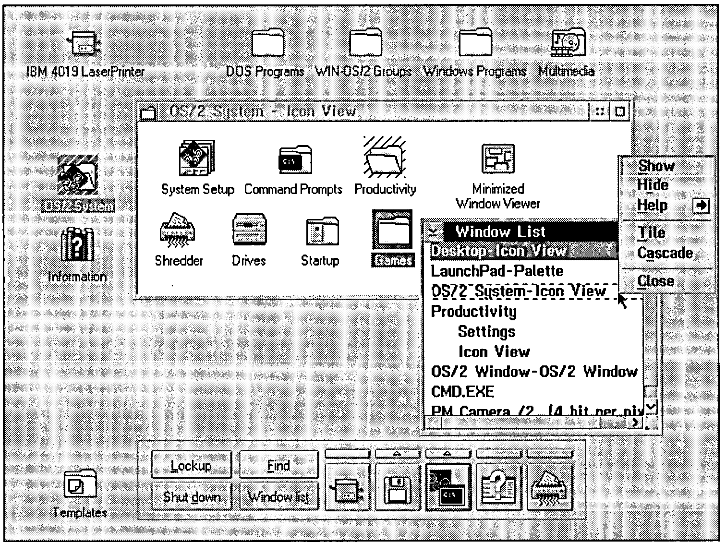
Uncensored

Movable and sizable windows. There is nothing more frustrating in my daily use of a system than when I am presented with a window that I cannot move. Often it is a dialog of some type, asking me a question or warning me about a situation, that is covering up a piece of information on the window below that pertains to it. So I have to dismiss the dialog, get the information I need, and cause the dialog to be displayed again. We strived in this release of OS/2 Warp to make all our windows moveable. They are sizable for the same reason.

Modeless windows. A modeless window is one that you can move out of the way to work on the window behind it. Windows used for action options and messages are as modeless as the situation permits. Sometimes the situation requires that you be presented with a window that must be dismissed before you are allowed to take further action on the window behind it. This type of window is called modal, and typically they are dialog or message boxes.

Window Lists. To facilitate easy access to all open windows, a special window called the Window List is provided (see Figure 2-10). The Window List shows a text-only list of all open windows in the system regardless of their state (minimized, maximized, hidden, or restored). The Window List supports switching and restoring.

Figure 2-10.
Window List



For windows opened as a result of opening objects from the Workplace Shell, the Window List displays the window's title in the format Object Name - View. If there is more than one view of that object open concurrently, the object name is shown and each view name is listed indented under the object name.

Later in this chapter, I will discuss the Menu pages of File System objects. In that section, you will learn how to add programs to the Menu pages of the Desktop or other folders. What you will notice if you do this is that when you run those programs, they show up in the Window List indented under the name of the Folder object that you launched them from. This makes sense when you understand that the Workplace Shell model considers them to be just another view of that object.

For windows opened from applications that were either started from another application or started from the command line using the Start command, the Window List displays the name of each window with no special formatting style. When you run applications from the command line (OS/2 Window or OS/2 Full-screen), the protect mode command processor (CMD.EXE) updates the Window List to display the name of the running program. When that program ends, the Window List entry for the command prompt session is restored to its original title. There are some things that you may do that will disable this behavior:

- ✗ Change either the path and filename or the parameters entry field on the Program page of the OS/2 Window or OS/2 Full-screen Program object.
- ✗ Use the Title parameter on the Start command.

There are several actions you can perform on the Window List. You access these functions from a pop-up menu on the windows in the Window List that you want to affect. Just as with objects in a folder, you can bring up a pop-up menu either on one or more selected windows or on a single window that is not part of the selected set. We will refer to the former as the selected windows and to the latter as the cursored window. If you bring up the pop-up menu on a selected set, the action applies to all selected items. If you bring up the pop-up menu on a cursored item that is outside the scope of the selected set, the action selected from the pop-up menu applies only to the cursored item. The available actions on a Window List entry are described below. All actions may not be available for all windows.

- ✘ **Show** — Makes the selected or cursored window the top-most window. If the window was hidden or minimized, it is first made visible and then it is restored to the state it was in prior to being minimized or hidden.
- ✘ **Hide** — Hides the selected or cursored window. When the window is hidden, you cannot find any type of minimized window icon representing that window. The only way to reaccess that window is either from its entry in the Window List or by resurfacing the view from the icon on the Desktop or folder that launched that window originally.
- ✘ **Minimize** — Minimizes the selected or cursored window. This option is only available if the Minimize push button behavior is set to either Minimize window to viewer or Minimize window to desktop in the Window page of the System object.

  When this behavior is set to Minimize window to viewer, the minimized window icon can only be seen if you open the Minimized Window Viewer folder. When this behavior is set to Minimize window to desktop, the minimized window icons appear on the Desktop with a yellow border around them to help distinguish them from Desktop icons. This is a source of some confusion to many users. The Workplace Shell does not arrange the minimized window icons with Desktop icons. It is as if the minimized window icons exist on an invisible plane above the Desktop window.
- ✘ **Help** — Shows the general help for the object represented by the selected or cursored window.
- ✘ **Tile** — Arranges the selected or cursored windows such that each window is visible in a checkerboard fashion.
- ✘ **Cascade** — Arranges the selected or cursored windows such that only a portion of each window is visible and each window is diagonally offset from the preceding window.
- ✘ **Close** — Closes the window. It should be noted that for Presentation Manager applications the effect of choosing Close from the Window List menu may not be identical to choosing Close from the System menu of the application. If the application was developed properly, it will appear identical; however, if the application developer failed to account for this difference, you may find that when you closed the application from the Window List, your application data was not saved. If this happens, you should contact your application provider to have them update the application.

If you want to perform an action on all open views of an object, just select that object name in the Window List and all open views that are indented underneath it will be selected automatically.

If you want to perform an action on all windows, simply select all windows in the Window List and choose the desired action from the pop-up menu. Special handling is done for the Desktop window to ignore those commands that it does not support.

Hidden and minimized windows

Hidden windows are provided as the default; minimized windows are a system option. Hidden windows are the default style because of the confusion associated with minimized windows on the Desktop. Minimized icons tend to look too much like icons on the Desktop. They are provided as an option because users of OS/2 1.x may wish to continue using that style. Also, some applications may require that their icons be minimized instead of hidden because they communicate status to the user by changing the minimized window graphic. The application is therefore relying on the ability of the user to see the changed graphic in the minimized window. For example, a mail program might change the minimized window graphic from a closed mailbox to an open mailbox when new mail has arrived.

Putting the Workplace Shell Model to Work

Now that the Workplace Shell model has been explained in some detail, we can explore how it was implemented specifically in OS/2 Warp.

Objects in the Workplace Shell

Everything in the Workplace Shell is represented as an object. Every object has a graphical icon and a title associated with it. All objects reside in a folder (the Desktop is a special folder, but it is a folder nonetheless). For every object, you can bring up its pop-up menu, open up the various views, and manipulate that object using drag and drop.

Opening the object pop-up menus

Every object in the Workplace Shell has a pop-up menu. At a minimum, you should be able to open up a view of the object, open up the settings for the object, and receive general help about the object.

You can get the pop-up menus for the objects in a selected set, or you can get the pop-up menu for an object outside of the selected set. To surface the pop-up menu for the objects in a selected set, first select the objects you want, then use either the manipulation button on the mouse or the Menu keyboard command. The default for the manipulation button is mouse button two. The default for the Menu keyboard command is Shift-10. These defaults can be changed in the Mouse object in the System Setup folder. If you bring up a pop-up menu for an object that is part of a selected set, the pop-up menu shows the intersection of all available menu choices for the selected objects. This means that the menu choice must be a valid menu choice on all the selected objects in order to be shown. This can cause a problem on Data File objects that do not share any common Open actions; no Open menu choice will be displayed. If you want to Open each of these selected objects to its default view, simply press **Enter.**

You may wish to raise a pop-up menu on an object outside of the selected set. You can do this via the mouse using the manipulation button as described above. In this case the actions selected from the pop-up menu apply only to that object.

Using the pop-up menus

In order to help you determine what objects actions on the pop-up menu apply to, a source emphasis is drawn around each object that would be affected by the pop-up menu choice. If hundreds of objects are selected (see Figure 2-11), it may take long enough for the intersection of all the menu choices to be calculated and for the source emphasis to be drawn that you perceive the delay. Do not request the pop-up menu again, because that will have the effect of dismissing the menu and you will have to start again. If the pop-up menu is for the folder background (remember that the Desktop is a folder), then the source emphasis is drawn around the inside of the folder window.

Once you have surfaced the pop-up menu, you can traverse its menu structure using either the keyboard or the mouse. Using the mouse, you press and hold the selection button (the default for the selection button is mouse button one) once the mouse pointer is over the menu. As you move over the various menu choices, they are highlighted. You choose the highlighted menu choice by releasing the selection button. If you release the button over a cascade menu, as indicated by a flat arrow, then the submenu is displayed. You can then repeat the steps above.

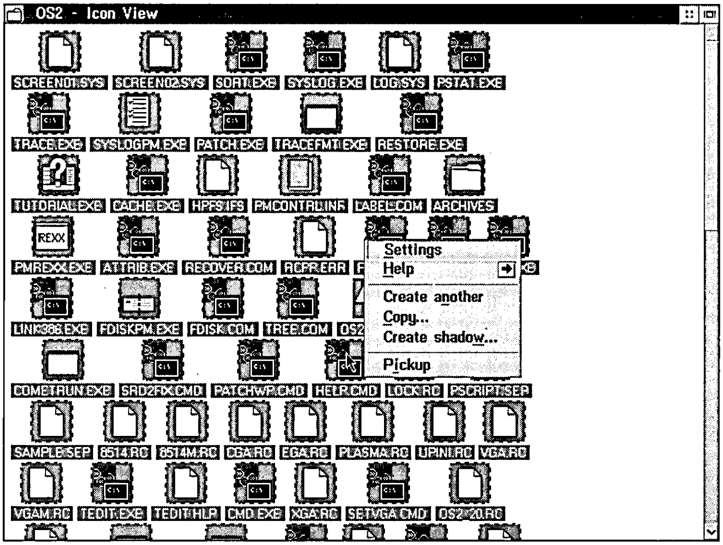


Figure 2-11.
Source Emphasis on Many Objects

By highlighting a menu choice outside of the submenu or the menu choice that displayed it, you close the submenu. You can dismiss the menu by releasing the mouse button outside the menu window. If you release the button over a conditional cascade menu, as indicated by the raised arrow, you have chosen the default menu choice for the conditional cascade menu. If you want to traverse the conditional cascade menu, then you need to move the arrow pointer over the raised arrow. This causes the submenu to be displayed. You can release the button at this point, in which case it will behave just like the cascade menu.

You can traverse a pop-up menu via the keyboard by using the up- and down-arrow keys to move up and down through the menu and using the left- and right-arrow keys to move into and out of submenus. To choose the highlighted item, press **Enter.** To choose any item, highlighted or not, that has an underlined letter, just press that letter on the keyboard. To dismiss a menu, press **Esc.**

To view help for the highlighted menu item, press **F1** when using either method.

Folders in the Workplace Shell

All Workplace Shell objects exist in a folder. A folder is the same as a file system directory, but since not all Workplace Shell objects have their persistent storage in the file system, you may not see all objects when looking at a directory from the command line. Worse, a directory at the command line may appear to be empty when it in fact has many Workplace Shell objects in it.

Since all folders are directories and all directories are folders, you can open any directory as a folder. This can be done either by finding that directory in the Drives folder or by using the **Find** push button from the LaunchPad.

As discussed previously in this chapter, the Workplace Shell provides three standard views of a folder: Icon, Details, and Tree views. The following features are common to all views of a folder:

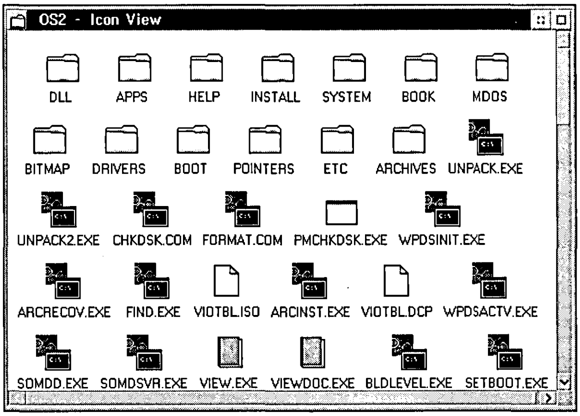
- ✗ The folder's pop-up menu can be accessed from the folder background or from the integrated folder system menu.
- ✗ The graphical icons and descriptive titles of the objects in the folder are displayed.
- ✗ Objects in a folder can be renamed by holding down the **Alt** key and pressing mouse button one.
- ✗ A default size for the folder window is picked the first time it is opened, based on the view type, the number of objects in the folder, and how long it takes to find all the objects in the folder.
- ✗ The folder is filled asynchronously to the display of the folder window. This means that even though you can see the open folder window, all the objects in that folder may not as yet be displayed.
- ✗ Many customizations of a folder are inherited from the parent folder. So, for example, if you want to change the default view for all folders on a drive, try changing the default on the Drive object.

Icon view

Icon view is the default view for all Workplace Shell folders. Icon view supports three formats: non-gridded, non-flowed, and flowed.

Non-gridded Icon view. Non-gridded is the default Icon view format (see Figure 2-12). This is sometimes referred to as Messy Desk view, because icons and their titles can overlap. In this view, the object title text always appears under the icon. If all of the objects can fit into the view, no scroll bars are displayed. If you wish to move objects around within the view or into the view, drag the object using the manipulation button. When you see target emphasis around the inside of the folder, then you know that the object is being moved into the folder area. If you see the target emphasis around the outside of an object's icon, then you know that you will be dropping the object onto that targeted object. You can optionally choose to turn off the object title text and to display the object icon using the small icon instead of the normal size icon. In this format neither scroll bar is displayed unless it is required.

Figure 2-12. Non-Gridded Icon View



Non-gridded Icon view supports the Arrange menu option on its pop-up menu. (Actually, the Arrange option is really there for the other views as well, it is just that its effect only applies to Icon view. If you do an arrange in another view, you will only see the effect after you switch to Icon view.) The system performs your arrange request by aligning the bottom-right edge of each icon with the bottom-left edge of the icon to its right. It continues to do this in a curved-S fashion for all the icons. After you do an arrange, you will find an Undo arrange menu choice on the pop-up menu. The Undo arrange option only remembers the last arrange you performed. Also, closing the folder view clears the Undo arrange buffer. The effect of Undo is affected by your choices for displaying icon text and your choice of which size icons to display.

Non-flowed Icon view. Non-flowed view is like an endless stack of object icons with the title text displayed to the right (see Figure 2-13). The vertical scroll bar is always displayed. The horizontal scroll bar is only displayed if you make the window width too small to display all the information. This view is considered an ordered view; none of the objects can overlay each other. When you want to move objects around within the folder, you use the manipulation button to drag the object to the desired location. You can also add an object at the top or the bottom.

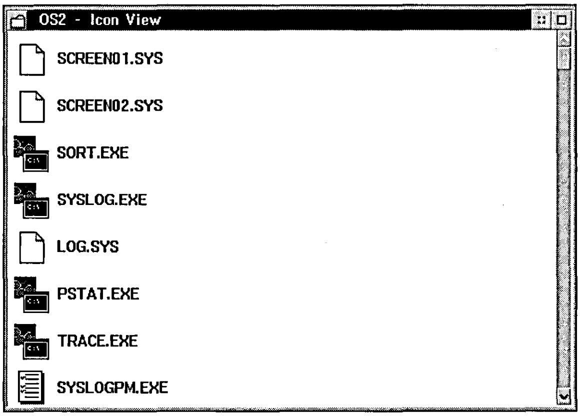


Figure 2-13.
Non-flowed Icon View

Your target emphasis for an ordered view of the folder is a horizontal black bar. If the target emphasis is around an object's icon (or title if the icon is invisible), then you know that you would be dropping the source object onto the target object instead of into the folder. You can choose to turn off the display of an object's title. You may also choose to display the object's icon at either normal or small size. Additionally, you can choose to have the icon not displayed, in which case the title text visibility is forced on.

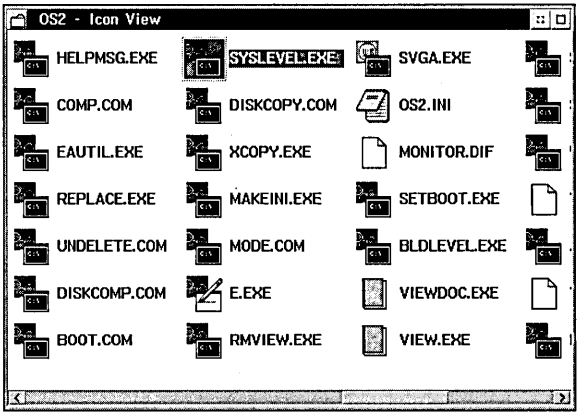
Flowed Icon view. Flowed format is extremely similar to non-flowed format, with one important difference (see Figure 2-14). You have multiple columns of stacked icons and titles instead of just one column. The number of rows is determined by the height of the window. There is always a horizontal scroll bar; the vertical scroll bar is only displayed if all the columns cannot be seen.

You can choose to sort the icons in Icon view for each of the three formats. For non-gridded view, the sort is treated like an arrange, but with a sort order. You cannot undo this sort. In the other views the objects are sorted from the top-left object, continuing down and then up to the top of the next column if you are in flowed format. The choices for the Sort menu are dependent on the object type selected on the Sort page of the Settings Notebook. This is discussed in more detail in Chapter 5. The shipped default for OS/2 Warp is to allow you to sort by object and file system information.

Details view

The Details view of a folder is like a columnar report of the most interesting information about the objects in the folder (see Figure 2-15). This is useful if you are looking for all objects in a folder that meet some criteria and you wish to then sort the folder by the criteria. For example, you might be looking for all File objects that were created today.

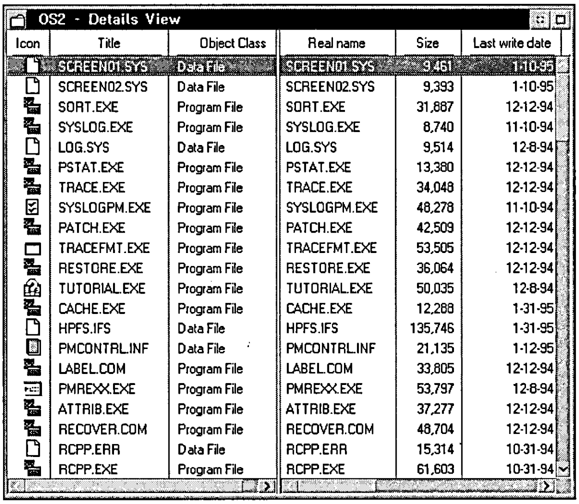
Figure 2-14. Flowed Icon View



OS2 - Icon View

HELPMSG.EXE
COMP.COM
EAUTIL.EXE
REPLACE.EXE
UNDELETE.COM
DISKCOMP.COM
BOOT.COM
SYSLEVEL.EXE
DISKCOPY.COM
XCOPY.EXE
MAKEINI.EXE
MODE.COM
E.EXE
RMVIEW.EXE
SVGA.EXE
OS2.INI
MONITOR.DIF
SETBOOT.EXE
BLDLEVEL.EXE
VIEWDOC.EXE
VIEW.EXE

Figure 2-15. Details View



OS2 - Details View

Icon	Title	Object Class	Real name	Size	Last write date
	SCREEN01.SYS	Data File	SCREEN01.SYS	9,461	1-10-95
	SCREEN02.SYS	Data File	SCREEN02.SYS	9,393	1-10-95
	SORT.EXE	Program File	SORT.EXE	31,887	12-12-94
	SYSLOG.EXE	Program File	SYSLOG.EXE	8,740	11-10-94
	LOG.SYS	Data File	LOG.SYS	9,514	12-8-94
	PSTAT.EXE	Program File	PSTAT.EXE	13,380	12-12-94
	TRACE.EXE	Program File	TRACE.EXE	34,048	12-12-94
	SYSLOGPM.EXE	Program File	SYSLOGPM.EXE	48,278	11-10-94
	PATCH.EXE	Program File	PATCH.EXE	42,509	12-12-94
	TRACEFMT.EXE	Program File	TRACEFMT.EXE	53,505	12-12-94
	RESTORE.EXE	Program File	RESTORE.EXE	36,064	12-12-94
	TUTORIAL.EXE	Program File	TUTORIAL.EXE	50,035	12-8-94
	CACHE.EXE	Program File	CACHE.EXE	12,288	1-31-95
	HPFS.IFS	Data File	HPFS.IFS	135,746	1-31-95
	PMCONTRL.INF	Data File	PMCONTRL.INF	21,135	1-12-95
	LABEL.COM	Program File	LABEL.COM	33,805	12-12-94
	PMREXX.EXE	Program File	PMREXX.EXE	53,797	12-8-94
	ATTRIB.EXE	Program File	ATTRIB.EXE	37,277	12-12-94
	RECOVER.COM	Program File	RECOVER.COM	48,704	12-12-94
	RCPP.ERR	Data File	RCPP.ERR	15,314	10-31-94
	RCPP.EXE	Program File	RCPP.EXE	61,603	10-31-94

As you can see from Figure 2-15, the Details view window is divided into two panes. The information in the left pane is the object information that is common to all classes of objects. The information on the right can change, depending upon the object type you selected on View - Page 3 of 3 in the folder's Notebook.

The shipped default is to display file system information. Additionally, you can choose whether to display certain columns by selecting or deselecting field names in the Details to display list box on that same Notebook page. If a particular object does not support that detail, then the cell is left empty. None of the abstract classes of objects would display any information in the right pane. The column descriptions are:

- ✘ **Icon** — Displays the object's icon in its small size.
- ✘ **Title** — Displays the object's descriptive title. You can change the object title in place just as you can for Icon view.
- ✘ **Object Class** — This is the descriptive name for the Workplace Shell object class that this object is an instance of. Each Workplace Shell object class has certain functional characteristics. These are described in more detail in a later chapter. This column is new to OS/2 Warp; the information is useful when you are confronted with choices that involve specifying the object class. Object class information can be found on Find, Sort, Include, and Details Settings pages.
- ✘ **Real name** — This is the physical filename of the object as it would appear when viewing that folder as a directory from the command line. It is not possible to change the real name of an object from the Workplace Shell. The real name is always derived from the title.
- ✘ **Size** — This is number of bytes of storage the object takes up in the file system.
- ✘ **Last write date** — This is the date of the last modification to the file.
- ✘ **Last write time** — This is the time of the last modification to the file.
- ✘ The following information is only available if the file system that the folder resides on uses the high performance file system (HPFS):
    - **Last access date** — This is the date of the last file access. Unlike Last write date, this tells you when the file was last accessed even if it was not changed.
    - **Last access time** — This is the time of the last file access.
    - **Creation date** — This is the date on which the file was created.
    - **Creation time** — This is the time at which the file was created.

✘ Flags —

- A — Archive

  This means that the file has been written to since it was last backed up.

- S — System

  This means that the file has been marked as an important system file. Normally you would not see objects with this flag on unless you changed the include criteria for the folder.

- H — Hidden

  This means that the file has been marked as a hidden file. Normally you would not see objects with this flag on unless you changed the include criteria for the folder.

- R — Read-only

  This means that the file is not deletable under normal circumstances.

The Details view of a folder is an ordered view. You move objects around within it and to it the same way you do in the Non-flowed Icon view of a folder. You also sort objects within it as you would for Non-flowed Icon view.

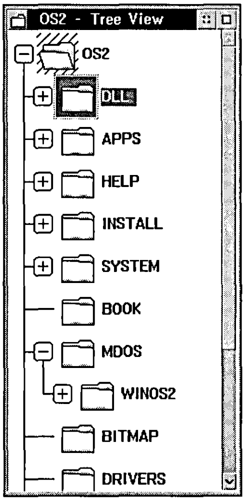
Tree view

The Tree view of a folder is meant to show you what other folders are contained within the folder and so on down the line (see Figure 2-16). This is known as the *hierarchy* of folders. This view of a folder does not show individual objects.

You can customize this folder view by choosing whether to display the tree lines. You can also choose to display the icons in normal or small size, or you can turn them off altogether. Unless you turn off the icons, you may choose not to display the object title text. I recommend changing the Tree view to display small icons.

If you worked with earlier versions of OS/2, you may have been bothered by your inability to sort in Tree view. This has now been fixed. There is no real strategy for rearranging objects within the Tree view; if you want to move one folder into another, simply drag and drop it using the manipulation button.

Figure 2-16.
Tree View



Templates in the Workplace Shell

The Templates folder is a special folder where the Workplace Shell collects object templates. The Templates folder is special because it queries every registered Workplace Shell class to determine if that class has any templates to add. Not every Workplace Shell class wants a template provided, while some classes provide more than one template. For example, the Color Palette class provides both the Mixed Color Palette and the Solid Color Palette.

In our data-driven world, we do not start an application and choose New from the File menu. Instead we create a template and drag a new instance of the object that we want to work with from the Templates folder into the folder where we want it to reside.

You drag a new instance of an object from the Templates folder by using the manipulation button on the mouse. To move the template itself you have to hold down **Shift** while you are dragging.

You can create templates for documents that do not already exist in the Templates folder. Simply save an empty document from your application. Open the Settings Notebook for the Workplace Shell object that represents that document and enable the **Template** check mark on the General page. You can move that template into the Templates folder or you can leave it in a document- or project-related folder. You can create as many templates as you need, one for each type of document that you regularly create. For example, you could have different word processing documents: one might be a memo template, one might be a status report template, and one might be a template for a book chapter.

Uncensored

One of the questions that I have been asked many times regarding templates is, "How do I set the directory associated with a document on my Desktop?" This user has created a document from a template onto their Desktop when they really wanted the document to exist in a specific data directory. What the user needs to do is move the document into that directory. If they want access to the document from the Desktop, they should create a shadow of the document on the Desktop.

Associations in the Workplace Shell

Because not all applications that you may wish to run on OS/2 Warp will be converted to the Workplace Shell model, we have provided a way for you to use your existing applications within the Workplace Shell.

Tip

This is accomplished through association. Association allows you to connect a data file to the application that is responsible for processing or editing that data file. You form a connection between the data file and its application by matching the data file's extension or type to the list of file extensions or types that a program is associated with.

Association using Association pages

You can tell a program which data files it is associated with on the Association page of either the Program or the Program File object. Figure 2-17 shows you the Association page of the OS/2 System Editor. As you can tell from the contents of the Current types list box, the OS/2 System Editor is associated with all data files that have the following types:

- DOS Command File
- OS/2 Command File
- Plain Text

File types are just strings of text that help you to categorize your data files. You can even add new types on the File pages of the data file. Additionally, the OS/2 System Editor is associated with all data files that match the file filters specified in the Current names list box:

- *.DOC
- *.TXT

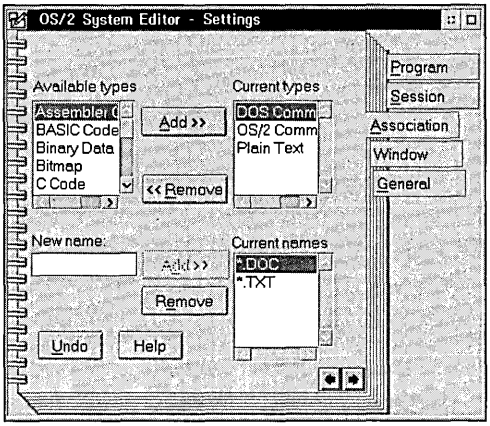


Figure 2-17. OS/2 System Editor — Association Page.

That is, it is associated with all files with extensions of .DOC or .TXT. File filters are generally used to match the data file to a specific file extension, but they can be used in other ways as well. You can specify the file filter the same way you specify filenames to be found using the directory command. For example, a file filter of *STATUS* will match all files that have the word STATUS anywhere in the name.

You add additional types that you want a program to be associated with by highlighting the type name in the Available types list box and pressing the **Add>>** push button. You remove types from being associated with a program by highlighting the type name in the Current types list box and pressing the **<<Remove** push button. Remember that by default all data files have an implicit type of Plain Text unless another type has been assigned either by the application developer or by yourself.

You add additional name filters by typing into the New name entry field and pressing the **Add>>** push button. To disassociate this program from a name filter, select that name filter in the Current names list box and press the **Remove** push button.

If you now go to a data file that matches one of the program's associated types or name filters, you will find the title of the program object listed in the Open menu's conditional cascade submenu. One way that you can tell if the association is being matched to an actual type or to an implicit type is by whether or not the icon on the data file matches the icon that is used on the Program or Program File object.

Suppose that several Program and Program File objects have specified the same file type or file extension, or the user has specified more than one file type on the data file. How does the system know which program to associate the data file with? The Workplace Shell uses the following rules to match the data file to its associated program:

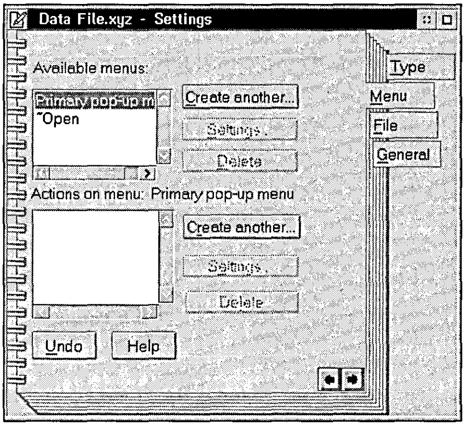
- ✗ The Workplace Shell finds all Program and Program File objects that have a file filter that matches the object title for the data file. It searches the list of programs stored in the association database in the order the programs were entered into the database. (A program is entered into the database when a name filter is entered for that program on the program's Settings Notebook Association page.) So the first program object that you entered the matching name filter for is found before the second program you entered it for, and so on. If you want to reverse the order, you have to go back to the oldest program, remove it, and then add it back again to change the chronology.
- ✗ For each type in the Current types list box of the data file, the Workplace Shell searches for all matching Program or Program File objects, searching in the same order that the types are listed in the list box. For example, if you have a data file associated with both DOS Command File and Assembler Code, in that order, the Workplace Shell searches for all Program objects that match DOS Command File, and then it searches for all Program objects that match Assembler Code.

The programs are added to the Open conditional cascade menu in the order in which they were found by the rules above. The first program found becomes the default. You can change the default to a different program by using the Menu page to change the default view of the data file. The Menu page is described next.

Association using Menu pages

The use of associations as described above is meant to be a flexible means of associating a group of like data files with the program that is responsible for processing them. What if you want to associate just one data file or even a folder with a program that can be used to process it, and you wanted to be 100 percent sure of which program would be assigned to do the processing? You can do this through the Menu page (see Figure 2-18). Also through the Menu page, you can change the default view of that data file or folder to be any one of the object views or any one of the associated programs or one of the programs you added to the Menu page. You can even make the Settings view be the default.

Figure 2-18.
Menu Page of a Data File Object



The simplest way to add a Program or Program File object to the Menu page is to drag that object into the Actions on menu list box after first selecting either the **Primary pop-up menu** or the **Open** menu in the Available menus list box. That selection will decide what menu the program gets added to.

Tip

Remember that selecting this or any program name from the Open menu opens a view of the data file, not of the program. That is why you will find in use emphasis around the data file and not around the program. Also remember that you set the characteristics of how you want the program to run by changing the settings of the Program object. The program's settings are used when this data file is launched into the program.

You can create your own additional cascade menus on the primary menu by pressing the **Create another...** push button next to the Available menus list box. Give your new menu a name in the Menu name entry field. Then decide if you want it to be a cascade or a conditional cascade menu. You can ignore the Default action drop down box for now. Press **OK** when you're done.

You can then add programs to that menu or any of the other menus in the Available menus list box by pressing the **Create another...** push button next to the Actions on menu list box. This leads to the Menu Item Settings dialog.

There are two ways to fill in this dialog:

1 Use the **Find program...** push button to find the program you want. Select the program you want in the Find Results dialog and press **OK**. This fills in the Menu item name and the Program name. It is very important that you do not

attempt to edit the Name field. If you do, it will break the link to the program and you will have to search for the program as described in step 2.

2 Fill in a menu name in the Menu item name entry field. Then type in the name of the program file that you wish to add in the Name field. Let me repeat, type in the name of the program file. You cannot specify a Program object here, only the name of an actual executable file. You can either type in the fully qualified file name, such as C:\OS2\CMD.EXE, or just the filename, such as CMD.EXE. If you do the latter, the PATH as set in CONFIG.SYS is used to find the program.

Tip

I repeat again that if you want to specify special characteristics of how a program is to be run, you have to find that executable and set its properties on its Settings Notebook pages. That is why I highly recommend that you only use Program objects for menu items. You can create as many of them as you need with the special characteristics for the different menu pages you wish to add them to.

Well, now that you have added programs to your menu, and even if you did not, you may wish to set the default view. For example, on a folder you may wish to change the default folder view from Icon view to Details view. Here is how you do it:

1 Select the **Open** menu in the Available menus list box.

2 Press the **Settings...** push button next to the Available menus list box.

3 In the Default action drop-down list box, set the view to the one you wish to be the default and press the **OK** push button.

Naturally you can delete the menus and menu items that you have added using the **Delete** push buttons.

Tip

Many of our customers like to add the programs they use most often to the Menu page of the Desktop. This was never the purpose of the Menu page, nor had we even anticipated that use of it. But who are we to argue with the resourcefulness of our users? The need to do this has been diminished somewhat by the introduction of the LaunchPad, but it still provides a convenient tool if you wish to use it. If you do use it, make sure you place a single percent sign in the Parameters field of the Program object to keep the name of the folder from being passed to the Program object.

Association using drag and drop

There is a third way of creating an association between a data file and a Program object. You simply drag the data file or files to the Program object that you wish to process them and drop them on it. Each one gets launched into its own separate copy of the program. (There is no way to specify that they all get sent to just one copy of the program.) Because you may want to provide for special processing of the dropped files, the Program object supports special parameters to help handle this for you. These parameters are described in the next section.

Program object parameters in the Workplace Shell

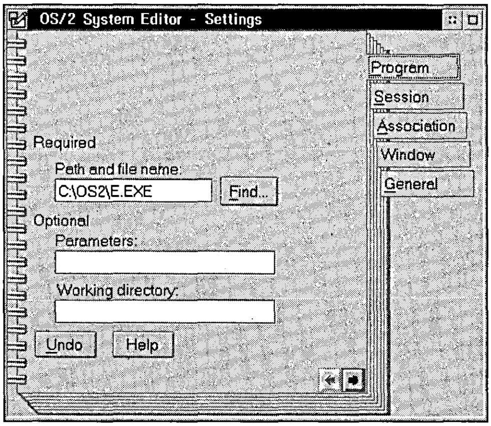
A Program object is a reference to the executable portion of an application. It is useful for pointing to the executable from another folder, like a shadow, but it also adds the ability to set the settings on the Program object differently than they are set on the program file. You can then have multiple Program objects that refer to the same executable, each with different settings for a different purpose.

The program or program file can be launched directly by the user, or indirectly via one of the three association methods described above. You specify parameters in the Parameters entry field and the Working directory entry fields to handle these cases.

The Program Settings Notebook page

Figure 2-19 shows an example Settings Notebook page for the OS/2 System Editor. Fields on this page include the Path and filename field and an optional Parameters field.

Figure 2-19. OS/2 System Editor — Program Page



OS/2 System Editor - Settings

Required

Path and file name: C:\OS2\E.EXE

Optional

Parameters:

Working directory:

In the Path and file name entry field:

- Use an asterisk (*) to specify that the system command processor should be used. The command processor used depends on the session type specified on the Session page. The session types are as follows:
  - CMD.EXE for OS/2 Full-screen or OS/2 Window. You can specify the name of a different default command processor for this session type by changing the value of the SET OS2_SHELL= statement in CONFIG.SYS.
  - COMMAND.COM for DOS Full-screen or DOS Window. You can specify the name of a different default command processor for this session type by changing the value of the SHELL= statement in CONFIG.SYS.
  - WINOS2.COM for WIN-OS/2 Full-screen.
- To specify the program to be used, enter the fully qualified path and filename that points to the executable, such as C:\OS2\E.EXE. You can use the **Find...** push button to help locate the program file and to fill in the name for you.

  Uncensored

  When you use the fully qualified path and filename, the Workplace Shell automatically keeps track of changes to that filename that may occur elsewhere in the system. For example, if the program file is renamed or moved either in the Workplace Shell, at the command line, or in another application, the Workplace Shell will update this field so that it still points to that executable.

  If you don't know the full path, just enter the program filename, such as E.EXE. In this case the PATH statement specified in CONFIG.SYS is searched to find the program. The program is not tracked for changes to the filename as it is if you had specified the fully qualified path and filename.

In the Parameters entry field:

- If the field is left blank, then nothing is passed if the Program is launched directly. If the program is launched indirectly via one of the association methods, then the fully qualified path and filename of the data file or folder will be passed as the parameter to the program.
- Entering a question mark (?) in the Parameters field causes the user to be prompted for input. This can occur multiple times within this field and in

combination with the other special parameter entries. The user's input is substituted in place of the ? with no additional blanks added either before or after.

✗ You can specify a prompt to be passed to the user by entering the prompt text into the Parameters field between brackets ([]).

The following parameters are only meaningful if you are using this program object for association. Remember that there were three forms of association. If the program is simply launched directly, then these special parameters will be stripped from the parameter string before the program is launched. All of these parameters can be used in conjunction with each other and the preceding parameters.

- %*edit_string*— A percent sign followed by any number of characters, up to but not including the next blank or the end of the line, is treated as a name editing string. The best way to describe an editing string is that it is what you might type as the second parameter on a copy or rename command.

  For example, if you typed the command COPY *.DOC *.TXT, then the extensions of all files that match the filter *.DOC would be changed to .TXT. The extension *.TXT is the editing string for that COPY command. You would use an edit string parameter in a similar fashion. For example, if you specify %*.TXT, then the name of the data file or folder passed in will be changed to end with .TXT.

  Since this parameter is used to indicate where the name of the file being dropped should be inserted into the parameter string, the name of the file is not added to the end of the parameter string. This can be useful if you need the parameter to be somewhere other than at the end. For example, you might need to type additional parameters after the name of the data file. You can pass files with embedded blanks or special characters through to the program by enclosing this special parameter in quotes. If the program is launched directly, the quotes will be stripped along with the rest of this special parameter.

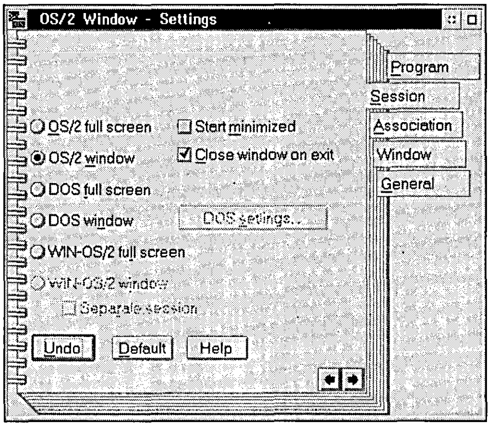
- %* —This is an example of this parameter that means to just accept the parameter as passed. It is useful for positioning the passed data file somewhere other than at the end of the parameter string.

- % — This is an example of this parameter that means to suppress passing the parameter to the program. This is useful when adding programs to the Menu page of the Desktop when you do not want the path of the Desktop passed to the program.
- %**D — Specifies that just the drive letter and the colon of the associated data file or folder are passed to the program. For example, if the data filename is C:\DATA\FILE.TXT, then just C: is passed to the program.
- %**P — Specifies that just the path of the associated data file or folder is passed to the program. For example if the data filename is C:\DATA\FILE.TXT, then just C:\DATA is passed to the program. If it is an associated folder with the name C:\FOLDER1\FOLDER2, then just C:\FOLDER1 is passed to the program.
- %**F — Specifies that just the filename of the associated data file or folder is passed to the program. For example, if the data filename is C:\DATA\FILE.TXT, then just FILE.TXT is passed to the program.
- %**N — Specifies that just the filename without the extension of the associated data file or folder is passed to the program. For example if the data filename is C:\DATA\FILE.TXT, then just FILE is passed to the program.
- %**E — Specifies that just the file extension of the associated data file or folder is passed to the program. For example if the data filename is C:\DATA\FILE.TXT, then just TXT is passed to the program.

You can string the various %** parameters together to form combinations of interest. For example, you can specify a parameter of %**P\%**N.BAK to cause the passed parameter to be sent with a renamed extension of .BAK.

The Session Settings Notebook page

When you enter the name of a program file on the Program page, the Workplace Shell goes out and determines what are the compatible session types for that program. You can view or change that determination on the Session Settings Notebook page (see Figure 2-20) as long as what you want to change it to has been determined to be compatible.



○ OS/2 full screen □ Start minimized
● OS/2 window ☑ Close window on exit
○ DOS full screen
○ DOS window
○ WIN-OS/2 full screen
○ WIN-OS/2 window
□ Separate session

Figure 2-20.
OS/2 Window — Session Settings Page

The following options are available on this page:

- **OS/2 full screen or OS/2 window** — You can choose to run an OS/2 mode application either in a window or in its own full-screen session.
- **DOS full screen or DOS window** — You can choose to run a DOS mode application either in a window or in its own full-screen session. Additionally, you specify DOS settings, which are explained in their own chapter later in this book.
- **WIN-OS/2 full screen or WIN-OS/2 window** — You can choose to run a WIN-OS/2 mode application either in a window or in its own full-screen session. Additionally, you specify WIN-OS/2 settings, which are explained in their own chapter later in this book. If you choose to run WIN-OS/2 in a window session you can choose to have it assigned to its own protected window session by checking the Separate session box. For performance reasons do not choose this check box unless you are finding that one WIN-OS/2 application is interfering with another WIN-OS/2 application running in the same shared WIN-OS/2 session.
- **Start minimized** — For windowed sessions (OS/2, DOS, or WIN-OS2), this command will start the window session minimized or hidden, depending on the global setting or the setting in that individual application.
- **Close window on exit** — This causes the windowed session to be automatically closed by the system when the session has ended. If you want to be able to view the information in the windowed session even after the session has ended, then check this box.

Summary

This chapter was meant to give you an overall feel for how the Workplace Shell was designed and some of the ways in which that design was implemented in OS/2 Warp. We will explore many more aspects of the Workplace Shell throughout this book. As you learn about the features of the Workplace Shell, you will better understand why it works the way it does.

Peter G. Magid is an Advisory Programmer on the Workplace Shell Development Team. As the User Interface Design Lead for the OS/2 Workplace Shell, Peter has had chief design responsibilities for what users see and how they interact with the Workplace Shell. Peter also plays an active role in reviewing and consulting on the content and direction of the Workplace Shell programming model, and he developed the code for the LaunchPad.

Chapter 3

Object Management

It's As Though Something Was Dragged and Dropped Over Here

In this chapter, I discuss basic object management and show you how to map the types of functions that you may be used to doing in the Windows File Manager to the equivalent functions under the Workplace Shell. I also discuss in some detail the various object selection techniques and the many drag and drop options available to you.

Selecting Objects

Uncensored

Whenever you want to manipulate more than a single Workplace Shell object in a folder, you must select the set of objects that you want to work with. The Workplace Shell allows you to use several different selection techniques, depending on the folder view you are interacting with. All Workplace Shell folders operate under what the IBM Common User Access Group defines as extended selection capability, except for the Tree View, which only allows single selection capability. These selection techniques are also applicable to many of the Presentation Manager Controls. Table 3-1 indicates which selection techniques are valid for each folder view.

Table 3-1. Valid Selection Techniques by Folder View

Selection Type	Non-Grid	Non-Flowed	Flowed	Details	Tree
Point	✓	✓	✓	✓	✓
Area (Marquee)	✓				
Range swipe (ordered)		✓	✓	✓	
Range swipe (unordered)	✓				
Range click		✓	✓	✓	
Add mode	✓	✓	✓	✓	

Using point selection

Point selection selects a single object and deselects all other objects in the folder.

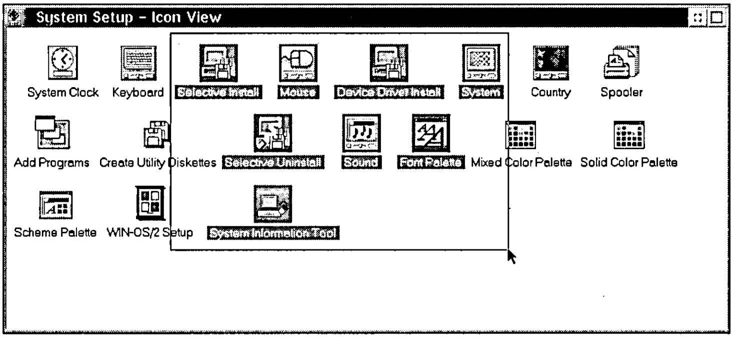
- ✗ **With the mouse** — Use mouse button one to click on the object in the folder.
- ✗ **With the keyboard** — Use the arrow keys. The cursor emphasis will change from object to object as you navigate through the objects. The selected object is the one on which the cursor emphasis is seen. You can use the Spacebar to remove selection from the cursored object.

Using area (marquee) selection

Area (marquee) selection selects all objects within a rectangular area; all other objects outside of the rectangle are deselected (see Figure 3-1).

This selection method can only be performed using the mouse. Press mouse button 1 on the white space of a folder, and then drag the mouse pointer, with the button still depressed, across the folder. A box will be drawn around the objects you wish to select.

Figure 3-1.
Area (Marquee) Selection

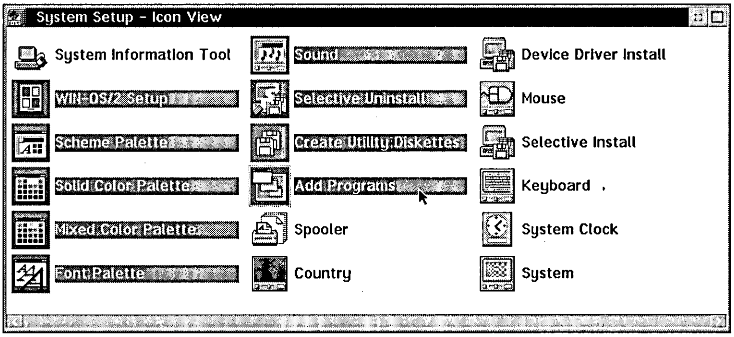


Using range swipe (ordered) selection

Range swipe (ordered) selection selects a set of objects from an ordered set of objects; all other objects are deselected (see Figure 3-2).

- **With the mouse** — Press and hold mouse button 1 on an object, and then move the mouse to the last object of the range you want included in the selection. Release mouse button 1 to end your selection.
- **With the keyboard** — Press and hold the **Shift** key while using the arrow keys to select objects. All objects between the first and the last object are selected.

Figure 3-2.
Range Swipe (Ordered) Selection

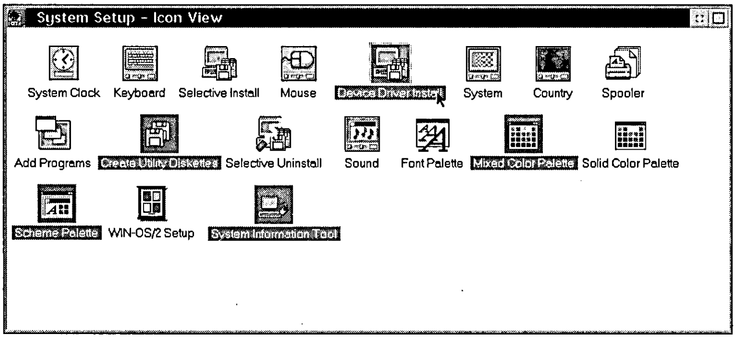


Using range swipe (unordered) selection

Range swipe (unordered) selection selects a random set of objects from an unordered set of objects; all other objects are deselected (see Figure 3-3).

This selection method can only be performed using the mouse. Press mouse button 1 over the first object you wish to select, and keep the mouse button pressed while you move the pointer over those objects you wish to select. If you accidentally select unwanted objects, see the Add Mode section for how to deselect these objects.

Figure 3-3: Range Swipe (Unordered) Selection



Using range click selection

Range click selection selects a set of objects from an ordered set of objects; all other objects are deselected.

This selection method can only be performed using the mouse. Click mouse button 1 on the first object in the range you wish to select, and then press and hold the **Shift** key while you use mouse button 1 to click on the last object in the range that you wish to select. All objects between the first and the last object are selected. The final selection will look the same as it does in Figure 3-2.

Using add mode selection

Add mode selection allows you to add or remove individual objects to or from the current selection. The other objects are not deselected.

- ✗ **With the mouse** — Press and hold the **Ctrl** key and click with mouse button one on the object you wish to have selected or deselected.

- **With the keyboard** — Press **Shift-F8,** which puts you into the Add mode. You will know that you are in Add mode when the cursor blinks at you. Once in Add mode, use the arrow keys to navigate to the object whose selection state you wish to change. When your cursor is at the desired object, use the **Spacebar** to toggle the selection state of that object. Press **Shift-F8** again to get out of the Add mode. The final selection will look the same as it does in Figure 3-3.

Understanding Drag Concepts

Once you have determined the object or objects that you want to work with, you will need to drag the object to a target in order to perform any type of operation with that object. In OS/2 Warp, drag is performed by pressing and holding the manipulation button (mouse button 2) over an object and then moving the mouse pointer with the manipulation button still depressed. The drop occurs when the manipulation button is released. To cancel a drag operation, either press the Esc key or drop the object(s) over a target that does not allow the drop, such as the title bar of a window.

The target of the drag can be one of the following:

A container of objects, such a another folder or the LaunchPad

A device object, such as a printer or the Shredder

In most cases, the source and the target agree on the proper drag operation to be performed under the current circumstances, in which case the default drag operation will be performed. A default drag means that you don't have to hold down any of the drag modifier keys (see Table 3-2) to have a specific drag operation performed. Following are some examples of default drag operations:

- **Drag move** — default for dragging objects between folders on the same hard disk or for dragging an object to the Shredder
- **Drag copy** — default for dragging objects between a folder on the hard disk and a folder on a diskette, or for dragging a file to a printer object
- **Drag create shadow** — default for dragging objects between a folder and the LaunchPad
- **Drag create from template** — default for dragging objects from a template to a folder

If you want a drag to do something other than the default operation, you need to use one of the drag modifier keys as described in Table 3-2. These keys override the default behavior.

Table 3-2. Drag Modifier Keys

Drag Modifier Key	Effect on Drag Behavior
Ctrl	Forces a drag copy
Shift	Forces a drag move
Ctrl-Shift	Forces a drag create shadow

Tip

You do not have to hold down the drag modifier key at the initiation of the drag operation. You can even change which modifier key, if any, is used during the drag. The modifier key in use at the time the drop operation occurs determines the drag mode that will be attempted.

Following are some examples of when you might want to use a drag modifier key:

- If you want to copy objects between folders on the same hard disk, you need to use the copy (**Ctrl**) modifier key.
- If you want to move objects from a folder on the hard disk to a folder on a diskette, you need to use the move (**Shift**) modifier key.
- If you want to add a template to the LaunchPad, you need to use the create shadow (**Ctrl-Shift**) modifier keys.

Introducing Pickup and Drop

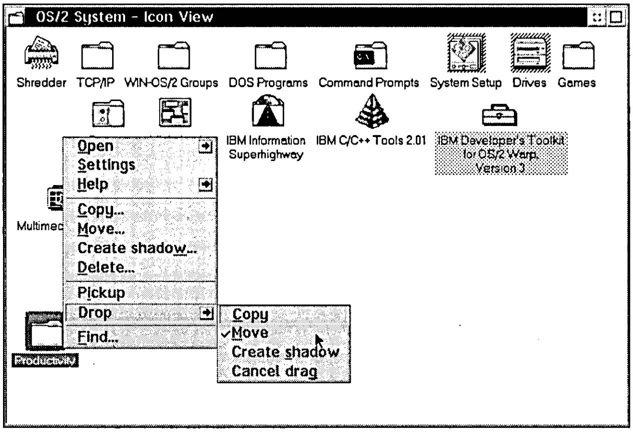
Uncensored

OS/2 Warp introduced a new data-transfer mechanism named pickup and drop. We developed the pickup and drop feature to

- Allow you to drag objects without having to continually keep mouse button 2 depressed.
- Provide a menu and keyboard mechanism to performing drag operations.
- Allow you access to other system functions during a drag.
- Allow you to perform multiple pickups of objects within the same source window.

Figure 3-4 illustrates the pickup and drop techniques using menus. In the figure, the IBM Developer's Toolkit object has already been picked up as indicated by the pickup emphasis on the object's title. The figure also shows the pop-up menu for the Productivity folder, which contains both the Pickup and the Drop menu choices.

Figure 3-4.
Pickup and Drop Menu Choices



Using pickup

If you want to use the mouse to pick up a single object or a selected set of objects, press and hold the **Alt** key and click with mouse button 2 on the single object or on any of the selected objects in the set. After the pickup operation, the mouse pointer becomes augmented with a lock symbol.

You can also perform the pickup operation from the pop-up menu of the object(s), using the mouse or the keyboard. Use **Shift-F10** to raise the pop-up menu for the selected object(s). If no objects are selected, the pop-up menu for the folder will be presented. If objects are selected and you still want to raise the folder's pop-up menu, press **Ctrl-\** before pressing **Shift-F10** (this clears the selection state from the objects).

Canceling pickup

To cancel the pickup operation, press and hold the Alt key and use mouse button 2 to click on the white space of the Desktop. You can also cancel a pickup operation, using either the mouse or the keyboard, by choosing the **Cancel** menu from the **Drop** conditional cascade. If **Cancel** is the only valid drop choice, then it will appear on the primary menu instead of within the **Drop** conditional cascade.

Dropping from a pickup

If you want to use the mouse to drop objects that were picked up, press and hold one of the drag modifier keys described in Table 3-2 as you use mouse button 2 to click on the drag target.

Using either the mouse or the keyboard, you can drop the objects that were picked up by choosing the **Drop** menu choice from the target's pop-up menu. If you choose the **Drop** menu, the default drop operation will be performed. You can also select a specific drop action from the **Drop** conditional cascade, for example, move, copy, create shadow, or cancel drag.

Object Management Functions

Now that you understand how to select and drag objects, let's turn our attention to the types of operations that can be performed on Workplace Shell objects.

Copying and moving objects

You copy and move files and other objects between folders in the Workplace Shell in much the same way that you copy and move files between directory windows in the File Manager. Every folder in the Workplace Shell is a file system directory.

To copy or move objects, you either drag a single object or a selected set of objects from the source folder to the target folder.

Note

You cannot copy or move objects (or folders containing objects) descended from the WPAbstract class into folders that exist either on removable media or on network drives. For information about which objects are descended from the WPAbstract class, please see Chapter 5. You also cannot copy or move objects to a read-only drive such as a CD-ROM drive or to a network drive to which you do not have write access.

If you attempt to copy or move a folder or individual object that does not fit on the target, you will be prompted to retry or cancel the operation. You may choose to make space at the target by deleting some objects and then retrying the operation, or you may choose to cancel the operation. If you cancel the operation, any portion of the folder or individual object that had been created at the target as part of the copy or move operation will be deleted from the target. This way, you will not be left with an incomplete copy or move.

Shadowing objects

Uncensored

There is no File Manager equivalent to creating an object shadow. This function is unique to the Workplace Shell. To create a shadow of a single object or a selected set of objects, press and hold Ctrl-Shift while you drag the objects to the target folder or the LaunchPad.

Note

You cannot create a shadow on a drive that contains read-only media, such as a CD-ROM drive, or on a network drive to which you do not have write access.

Printing objects

You print file objects by dragging them to the printer object or to the open printer object window. Dragging to the printer object can only be a copy operation. If you attempt to use any of the other drag modifiers, the drop will not be accepted.

Deleting objects

Uncensored

You delete objects by dragging them to the Shredder. If you drag a folder to the Shredder, the entire folder subdirectory structure will be deleted. The following enhancements to the delete function were added for OS/2 Warp:

- ✗ The ability to be prompted before each and every object is deleted for confirmation of the deletion of that object.
- ✗ The ability to select **Yes to all** in response to the confirmation prompt, to prevent further prompting for individual objects.
- ✗ The ability to delete non-deletable objects. To delete non-deletable objects, move the objects into a folder that is deletable and then delete the folder. As the non-deletable objects are encountered by the delete processing, you will be prompted to confirm that you wish to delete the object. Every attempt will be made to delete the object by modifying the underlying object attributes, such as turning off the read-only attribute, should you select Yes in response to the confirmation prompt.

Creating objects

The primary mechanism for creating new objects in the Workplace Shell is to drag a new instance of the object from a template of the object in the Templates folder to the target folder in which you want the new object.

For example, to create a new directory in a particular folder:

1 Open the Templates folder.

2 Open the folder that you wish to contain the new directory or have the icon for that folder visible to you within its parent folder.

3 Press and hold mouse button 2 on the Folder template in the Templates folder, and drag and drop it into the target folder or onto the target folder's icon.

Note

You can create new instances of other objects in the same way. Under the File Manager, you would have created a directory using the Create Directory menu choice on the File menu.

Naming objects

After an object is created you can rename the object by

1 Pressing and holding the **Alt** key and then clicking mouse button 1 on the title text of the object you wish to rename.

2 Typing in the new object name in the edit window.

3 Clicking on the white space of the folder outside of the edit window to close the edit window.

Note

The physical name of a file system object is always constructed from its title. You cannot enter the physical name separately. Under the File Manager, you would have used the **Rename** command under the File menu to accomplish this task.

Managing disks

The Workplace Shell has several options for managing your disks that correspond to and exceed those capabilities offered by the File Manager.

Partitioning your disk

You can partition your disks through the Workplace Shell by using the **Create partition...** menu choice from the pop-up menu of the Drives folder.

Copying disks

You can copy your disks through the Workplace Shell by using the **Copy disk...** menu choice from the pop-up menu of the disk object that you wish to copy. This is equivalent to the Copy Disk menu choice under the Disk menu in the File Manager.

Checking disks

You can check a disk for errors using Workplace Shell by using the **Check disk...** menu choice from the pop-up menu of the disk object you wish to check.

Formatting disks

You can format a disk using the Workplace Shell by using the **Format disk...** menu choice from the pop-up menu of the disk object you wish to format. This is equivalent to the Format Disk menu choice under the Disk menu in the File Manager.

Starting programs with an object

You can start a program by passing it a data file object. Drop the data file object onto the program object for the program you want to execute.

Confirmation options

The amount of prompting you receive during each of the above functions is a matter of personal choice. To control your level of prompting, use the Confirmations Settings Notebook page of the System object, which resides in the System Setup folder.

Summary

Besides discussing the various selection and drag and drop techniques available in OS/2 Warp, this chapter was also intended to help you map what are typically thought of as file-management functions in the Windows File Manager to their equivalent functions in the Workplace Shell. All the functions listed here are described in more detail in Chapters 5, 9, and 15.

Peter G. Magid is an Advisory Programmer on the Workplace Shell Development Team. As the User Interface Design Lead for the OS/2 Workplace Shell, Peter has had chief design responsibilities for what users see and how they interact with the Workplace Shell. Peter also plays an active role in reviewing and consulting on the content and direction of the Workplace Shell programming model, and he developed the code for the LaunchPad.

Chapter 4

The LaunchPad

Hmmm, This Seems to Be Where It All Starts

The OS/2 Warp Development Team created the LaunchPad (see Figure 4-1) to provide fast and easy access to your most commonly used objects and Desktop functions. You can customize both the display characteristics of the LaunchPad and the objects on the LaunchPad.

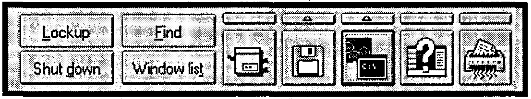
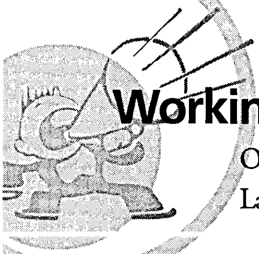


Figure 4-1.
The LaunchPad



Working with the LaunchPad

OS/2 Warp provides fast access to your LaunchPad. When working with the LaunchPad, you need to understand three distinct areas:

- The Action Button area

- The Object Button area
- The Drawer Access Button area

There are also drawers that can contain additional objects. Much like a drawer in a dresser or cabinet, the drawers in the LaunchPad hide their contents until they are needed by the user. This feature enhances the overall goal of the LaunchPad, which is to give users fast access to the objects they need quickly, while at the same time reducing the amount of screen real estate it takes to display those objects.

Fast and easy access

To ensure that you have fast and easy access to your LaunchPad, OS/2 Warp will automatically start the LaunchPad for you each time the system starts.

Tip

You can always surface the LaunchPad by double-clicking mouse button 1 on the white space of any open folder or the Desktop. If required, the LaunchPad will be created and/or opened before it is surfaced. This means that if the LaunchPad is ever deleted, it will be re-created on either the next double-click of mouse button 1 on the Desktop or folder or the next time you restart the system. You can use this feature to have the system re-create the default LaunchPad for you if you want to delete your customized LaunchPad.

You can move the LaunchPad by simply dragging any of its white space with mouse button 2. You can access the pop-up menu of the LaunchPad by clicking mouse button 2 on the white space of the LaunchPad.

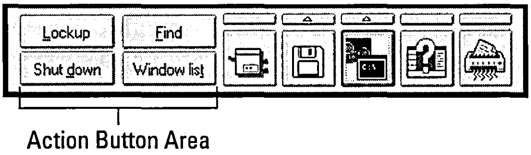
Action Button area

The Action Button area (see Figure 4-2) allows you to activate the most common Desktop functions without having to discover them on the Desktop's pop-up menu. The following actions are provided in this area:

- **Lockup** — Activates the Desktop Lockup utility. This is identical to choosing **Lockup now** from the Desktop's pop-up menu.
- **Find** — Activates the Desktop Find utility. This is identical to choosing **Find...** from the Desktop's pop-up menu.

- **Shut down** — Activates the Desktop Shut down utility. This is identical to choosing **Shut down...** from the Desktop's pop-up menu
- **Window list** — Activates the Window List. This is identical to pressing **Ctrl-Esc**

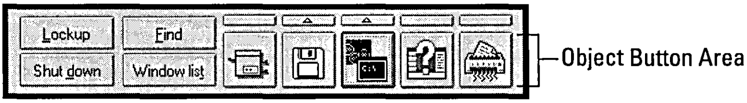
Figure 4-2.
Action Button Area



Object Button area

The Object Button area (see Figure 4-3) allows you to work with your most common Workplace Shell objects faster. Instead of having to double-click on the objects in the Object Button area, you simply single-click on them. Objects in this area are given a push-button appearance to help support this single-click notion. If you double-click on them, the second click is ignored.

Figure 4-3.
Object Button Area



The objects on the LaunchPad are all shadow objects. The LaunchPad itself does not provide object storage like a folder does. The real objects that are shadowed onto the LaunchPad are stored in the Nowhere directory.

LaunchPad objects behave just like any other Workplace Shell objects (except that LaunchPad objects respond to single-clicks, whereas Workplace Shell objects require double-clicking). You can

- Access the object's pop-up menu. Use mouse button 2 to click on the object button to access the object's pop-up menu.
- Use the object as a drag source. You can drag the object from the LaunchPad to any object or folder that would normally accept the drop of that object. Remember that the objects on the LaunchPad are shadows, so you are dragging a shadow object. There is no concept of selection on the LaunchPad, so just start dragging

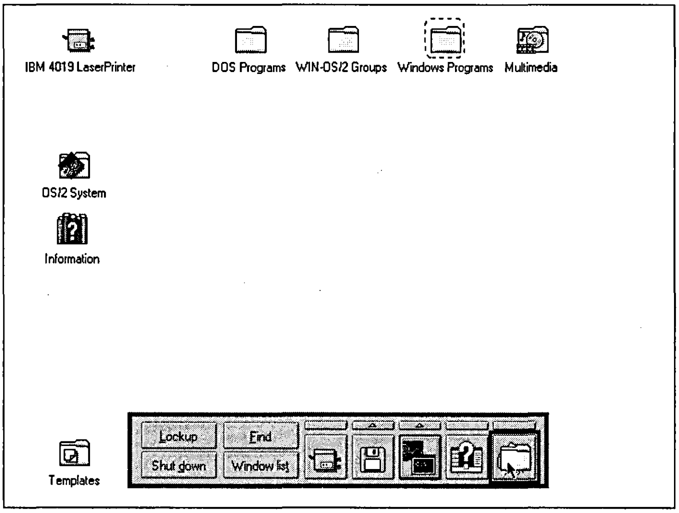
as you would an object in a folder. You do not need to select the object on the LaunchPad before you begin dragging it. You are limited, however, to dragging one object at a time from the LaunchPad.

- Use the object as a drop target. You can use the objects on the LaunchPad as targets for a drag and drop operation. Figure 4-4 shows the Windows Programs folder being dragged to the Shredder on the LaunchPad. The drag and drop operations available between objects in a folder and objects on the LaunchPad are the same operations that would be available if those objects were in folders. The black border around the object button is your visual clue that the source object is going to be dropped onto the target object button and not onto the LaunchPad itself.

When OS/2 Warp is installed on your system, it constructs a LaunchPad with the following set of objects on the LaunchPad:

- **Default Printer** — When the LaunchPad is being constructed it looks for the default printer on the Desktop. If it can, it adds a shadow of the default printer object as the first object. If it cannot find the default printer on the Desktop, it does not add a printer object to the LaunchPad.

Figure 4-4. Dragging to an Object on the LaunchPad



- **Drive A** — This is a shadow of the Drive A object in the Drives folder.
- **OS/2 Window** — This is a shadow of the OS/2 Window object in the Command Prompts folder.
- **Tutorial** — This is a shadow of the Tutorial object in the Information folder. This is the same tutorial that runs when the system starts for the first time.
- **Shredder** — This is a shadow of the Shredder object in the OS/2 System folder.

Tip

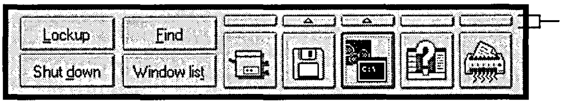
You can use the **Original/Locate** menu choice from the object's pop-up menu to gain fast access to the folder that contains the original of the LaunchPad object. For example, if you want to use the OS/2 Fullscreen object instead of the OS/2 Window object, use the **Original/Locate** menu choice from the OS/2 Window object on the LaunchPad to quickly navigate to the Command Prompts folder, where you will find the OS/2 Fullscreen object.

If the object on the LaunchPad is set to resurface its already open view, as most objects are, then you do not need to access the Window List to resurface the object's window. Simply single-click on the object button, and the Workplace Shell will resurface the view for you.

Drawer Access Button area

Figure 4-5 shows the Drawer Access Button area, the small group of push buttons above the larger drawer graphic buttons. Graphics on the push buttons indicate whether the drawers are open or closed and whether they contain any objects you can use (see Table 4-1).

Figure 4-5. Drawer Access Button Area



Drawer Access Button Area

Table 4-1. Drawer Access Button Indicators

Graphic	What Happens When You Click on the Drawer Access Button
None	The LaunchPad will beep at you because the drawer is empty.
Up-Arrow	The closed drawer will open, and focus will be given to the bottommost object in the drawer. The objects in the drawer are stacked linearly in a perpendicular direction to the LaunchPad. The graphic on the Drawer Access Button will change to a down-arrow to indicate that the drawer is now open.
Down-Arrow	The open drawer will close, and focus will be given to the last button that had focus on the LaunchPad. The graphic on the Drawer Access Button will change to an up-arrow to indicate that the drawer is now closed.

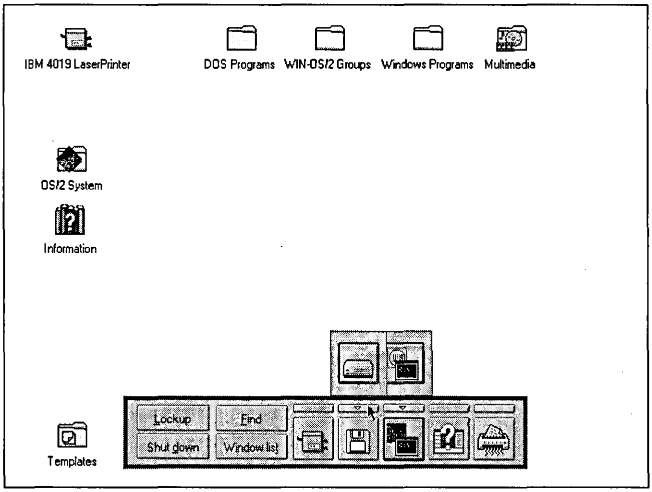
The drawers of the LaunchPad (see Figure 4-6) give you a place to store objects that are related to but are used less frequently than the corresponding LaunchPad objects.

For example,

- ✗ The object that comes preconfigured in the drawer for the Drive A object is the drive object that represents the drive on which you installed OS/2 Warp.
- ✗ The object in the drawer for the OS/2 Window object is the DOS Window object.

Although this was the purpose for which we intended the drawers, there is nothing to restrict you from placing whatever objects you like in any drawer.

Figure 4-6. LaunchPad Drawers



How drawers open

Uncensored

The drawers are designed to open above the LaunchPad when the LaunchPad is configured horizontally. If the LaunchPad is configured vertically, then the drawers are designed to open to the right of the LaunchPad.

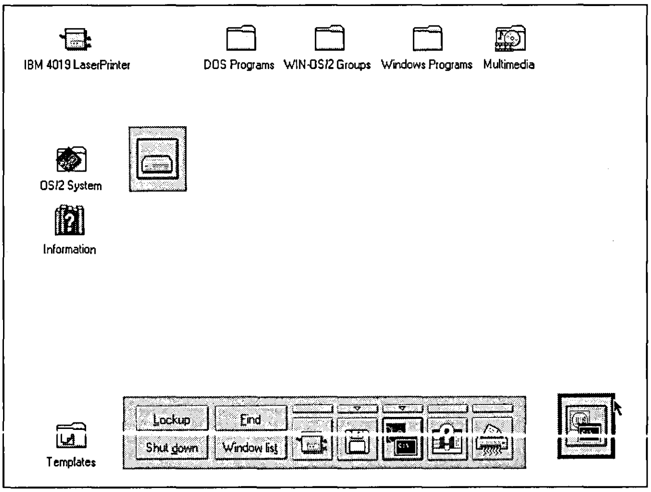
Before opening a drawer, the LaunchPad will check to see if there is enough room to open the drawer in its preferred direction without having any part of the drawer appear off screen. If it cannot open the drawer this way, it will check to see if there is enough room to display the drawer the other way. If it can display the drawer the other way without it going off screen, it will do so. If it cannot open the drawer in either direction without some part of the drawer appearing off screen, it will open it in the direction in which less of the drawer appears off screen.

Breakaway drawers

You can drag a drawer away from the LaunchPad (see Figure 4-7) and move it wherever you like. To break a drawer away from the Launch Pad, press and hold mouse button 2 on the white space of the drawer and move the mouse while keeping the button depressed. Drawers cannot be placed behind the LaunchPad. If you bring the LaunchPad or any drawer to the foreground, all drawers and the LaunchPad will come to the foreground as a unit.

The position of a drawer will not be retained the next time the drawer is closed and reopened. The next time the drawer is opened, it will appear in its default position relative to the LaunchPad.

Figure 4-7.
Free-Standing Drawer



Keyboard access

Table 4-2 lists the keyboard access techniques for the LaunchPad and its drawers:

Table 4-2. Keyboard Access for the LaunchPad and Its Drawers

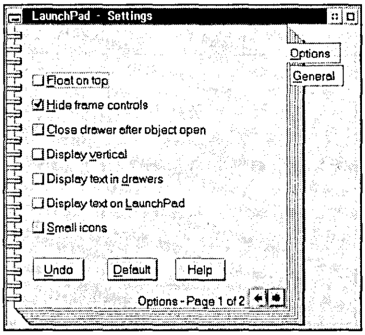
Key	Description
Up-Arrow or Right-Arrow	Moves to the next button in the LaunchPad or drawer in an upward and then right direction.
Down-Arrow or Left-Arrow	Moves to the next button in the LaunchPad or drawer in a downward and then left direction.
Spacebar or Enter	Activates the button that has the focus.
Alt-F7	Puts the LaunchPad or drawer in move mode. Use the mouse or arrow keys to move the window. Use the Enter key or a mouse button click to end the move. Use the Esc key to cancel the move.
Alt-F4	Closes the LaunchPad or drawer.
Alt-F6	Moves the focus from the LaunchPad, through each of the open drawers, and back to the LaunchPad.
L	Activates the Lockup action button
D	Activates the Shut down action button
F	Activates the Find action button
T	Activates the Window list button

Customizing the LaunchPad

You can customize the LaunchPad in several ways: You can use the Settings Notebook to change the visual look of the LaunchPad; you can add and remove objects to and from the LaunchPad; and you can also drop colors and fonts onto the LaunchPad.

Setting LaunchPad options

Figure 4-8 shows you the first Settings Notebook page in its default configuration.



□ Float on top
☑ Hide frame controls
□ Close drawer after object open
□ Display vertical
□ Display text in drawers
□ Display text on LaunchPad
□ Small icons

Options - Page 1 of 2

Figure 4-8. LaunchPad - Options, Page 1 of 2

The following options may be set on this page:

- The **Float on top** option causes the LaunchPad and its open drawers to remain on top of all other open windows in the system.
- The **Hide frame controls** option, which is the default, hides the System menu and the title bar of the LaunchPad and drawers. Turn this option off if you want to see the title bar and the system menu (see Figure 4-9).

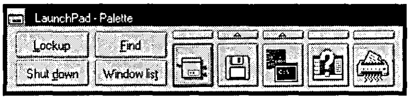


Figure 4-9. LaunchPad with Frame Controls

- The **Close drawer after object open** option automatically closes a drawer after you open an object in the drawer.
- The **Display vertical** option displays the LaunchPad vertically instead of horizontally (see Figure 4-10). Turn this option off if you want the drawers to be displayed horizontally.

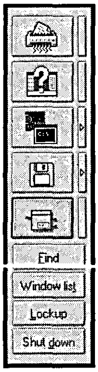


Figure 4-10. Vertical LaunchPad

✗ The **Display text in drawers** option displays object titles for objects in a drawer when the drawer is open (see Figure 4-11).

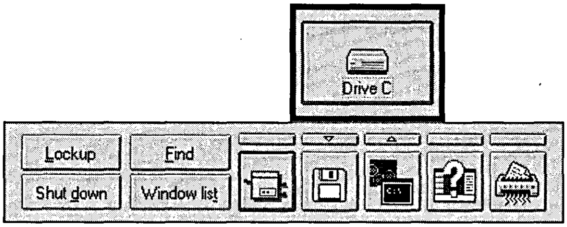


Figure 4-11. Object Title in Drawers

✗ The **Display text on LaunchPad** option displays object titles for the objects on the LaunchPad (see Figure 4-12). You can also edit the object's icon to include more descriptive information about the object in its icon. To edit the object's icon, press the **Edit** push button on the General page in the object's Settings Notebook.

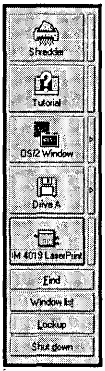
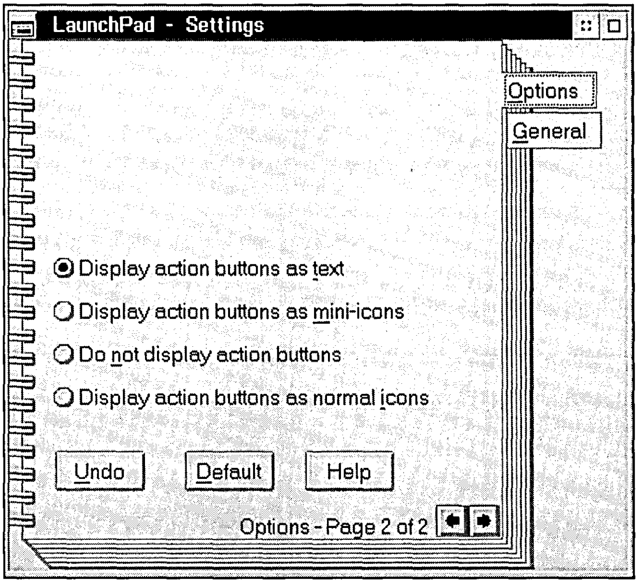


Figure 4-12. Object Titles on the LaunchPad

✗ The **Small icons** option displays the buttons in the Object Button area as mini-icons instead of normal icons.

Several other options can be set on the second Options page of the LaunchPad Settings Notebook (see Figure 4-13):

Figure 4-13. LaunchPad - Options Page 2 of 2



● Display action buttons as text

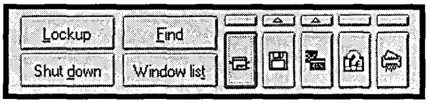
○ Display action buttons as mini-icons

○ Do not display action buttons

○ Display action buttons as normal icons

- ✗ The **Display action buttons as text** is the default. Figure 4-14 shows what the LaunchPad looks like if the **Small Icons** options is enabled. The action buttons are in text, and the object buttons are displayed as mini-icons.

Figure 4-14. Text Action Buttons with Small Icons



Developer's Desktop

- ✗ The **Display action buttons as mini-icons** option does just that. Figure 4-15 shows how the LaunchPad looks when the action buttons are displayed as mini-icons and the object buttons are displayed as normal icons. This is how my LaunchPad is configured.

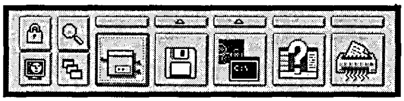


Figure 4-15. Mini-icon Action Buttons with Normal Icons

Figure 4-16 shows how the LaunchPad looks with the action buttons displayed as mini-icons and the object buttons displayed as small icons. This is the smallest configuration for the LaunchPad in terms of screen real estate with the action buttons still displayed.

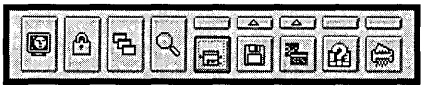


Figure 4-16. Mini-icon Action Buttons with Small Icons

- ✘ The **Do not display action buttons** option turns off the action buttons. Figure 4-17 shows how the LaunchPad looks when the action buttons are turned off and the object buttons are displayed as normal icons.

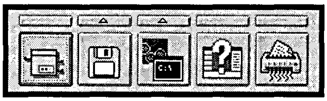


Figure 4-17. No Action Buttons and Normal Icons

Figure 4-18 shows how the LaunchPad looks when the action buttons are turned off and the object buttons are displayed as small icons.

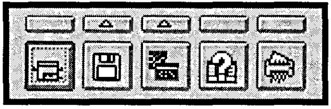


Figure 4-18. No Action Buttons and Small Icons

- ✘ The **Display action buttons as normal icons** option does just that. Figure 4-19 shows how the LaunchPad looks when the action buttons are displayed as normal size icons and the object buttons are displayed as normal icons.

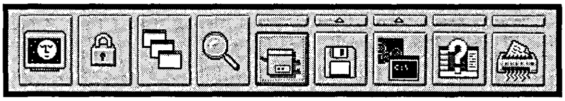


Figure 4-19. Normal Icon Action Buttons and Normal Icons

Figure 4-20 shows how the LaunchPad looks when the action buttons are displayed as normal size icons and the object buttons are displayed as normal icons.

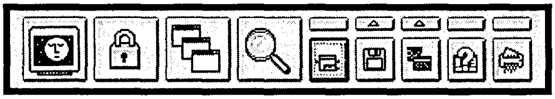


Figure 4-20. Normal Icon Action Buttons and Small Icons

Adding and removing objects

The only one way you can add an object to the LaunchPad or to one of its drawers is by dragging and dropping it, unless you're willing to resort to some type of programming. You can remove an object either by dragging and dropping it to the Shredder or by using the object's pop-up menu.

Adding objects to the LaunchPad

You add objects to the LaunchPad by dragging an object from a Workplace Shell folder and dropping it over any white space on the LaunchPad or white space on a drawer. The LaunchPad will show a solid black line to target the nearest drop zone. Figure 4-21 shows one of the drop zones for adding an object to the LaunchPad.

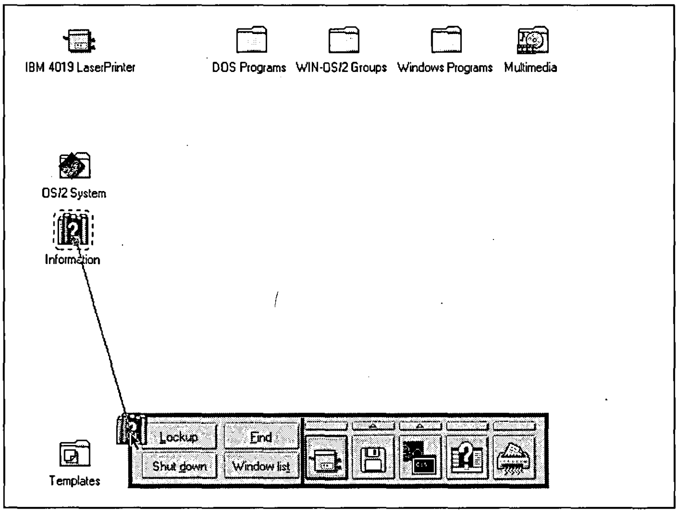


Figure 4-21. Adding an Object at the Beginning of the LaunchPad

Figure 4-21 shows a line from the source to the target. This line indicates that a shadow of the object will be created. If you want to put a template object on the LaunchPad, you have to hold down the **Ctrl** and **Shift** keys while you are dragging the object since dragging from a template usually does not result in the creation of a shadow. There is additional white space on both sides of the LaunchPad to ensure that you have enough space for a drop zone. Figure 4-22 shows what the LaunchPad looks like after the object has been dropped.

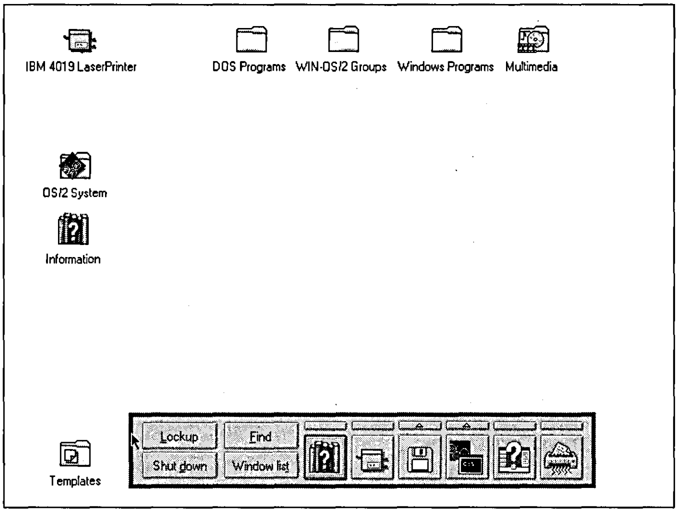


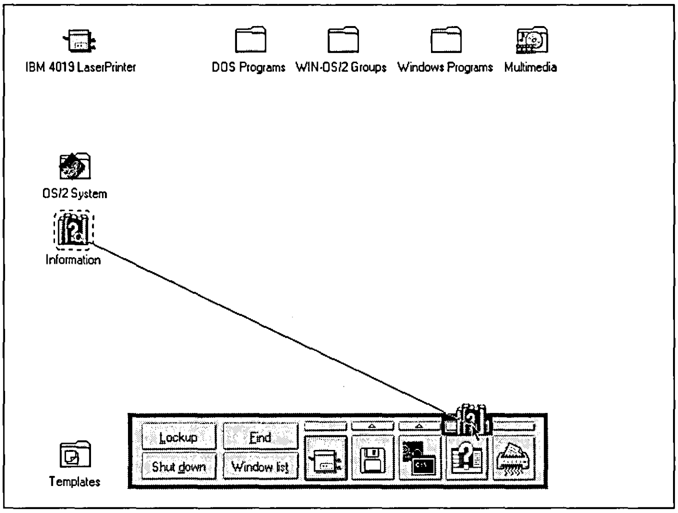
Figure 4-22. The LaunchPad After an Object Has Been Dropped on It

You can also drag objects to the LaunchPad icon on the desktop. This will add objects to the far-right part of the LaunchPad Object Area. You are not limited to dragging a single object at one time to the LauchPad or to one of its drawers. Because the LaunchPad does not have scroll bars, if you add more objects than can fit on screen, you will have to move the LaunchPad around to access the objects off screen.

Adding objects to a drawer

You add an object to a drawer either by dragging the source object to the white space of the open drawer or by dragging the source object to the Drawer Access button for that drawer, as shown in Figure 4-23. You will notice a thick black border drawn around the Drawer Access button to indicate that it is being targeted.

Figure 4-23. Dragging an Object to a Drawer Access Button



Removing objects from the LaunchPad or a drawer

You remove objects from the LaunchPad either by selecting **Delete** from the object's pop-up menu or by dragging the object from the LaunchPad to the Shredder or to a folder such as the Desktop. Since any of these actions are made on a shadow object, the original object is undisturbed. If you delete an original object that has a shadow of itself on the LaunchPad, the shadow object will be deleted automatically from the LaunchPad.

If there is a drawer above the object that is being deleted from the LaunchPad, then the topmost object in the drawer is placed on the LaunchPad in place of the deleted object. If the last object in a drawer is being deleted, then the drawer is removed. If there are no drawers and the last object from the LaunchPad is deleted, the original LaunchPad set of objects will be re-created.

Dropping colors and fonts on the LaunchPad

Shhh! Tip

You can also customize the LaunchPad by dropping colors and fonts from the color palettes and from the font palette to the background of the LaunchPad, to a drawer, or to individual buttons. It should be noted that Drawer Access buttons that are not currently displaying an arrow are disabled. As a result you cannot drop a font or color on them. To customize them, you will need to first add an object to them to create a drawer. You can delete the drawer object and the drawer after you complete your customization of the Drawer Access button. If you drop a font onto the background of the LaunchPad, you will need to close and reopen the LaunchPad to see the effect of the font size change on the text buttons.

Each button can have individual colors and fonts stored on them. Unless the drawer has had a color or font dropped on it, it will inherit the color and font of the LaunchPad. Once you drop a color or font onto the LaunchPad, drawer, or button, there is no way to undo that action. You can always drop the orginal font or color back.

Special Considerations

The following topics represent a set of questions I have been frequently asked about the LaunchPad since OS/2 Warp shipped.

Disabling the LaunchPad

Shhh! Tip

If you want to disable the automatic opening and re-creation ability of the LaunchPad, you·need to remove the LAUNCHPAD option from the AUTOSTART keyword in CONFIG.SYS. The LaunchPad will still open automatically if you double-click on Desktop white space. The only way to get around that is to either delete the LaunchPad or to give it a different Object ID (see Chapter 8 to learn how to write a REXX batch file for this purpose).

Copying the LaunchPad

Warning

There is a bug in the LaunchPad that affects copying of the LaunchPad. For a description of this problem and the solution, see Chapter 5.

Green question marks

Uncensored

You may occasionally find that a button on your LaunchPad shows up as a green question mark. This indicates that the object for that button could not be found. This can result if you deleted the object when the LaunchPad was closed. To remove this button, simply drag it to the Shredder. You cannot get a pop-up menu on a button with a green question mark.

Tip

If you ever find that all the buttons on your LaunchPad show up as green question marks, this indicates that you have deleted the Nowhere directory, which the Workplace Shell created to store hidden and temporary objects. All you need to do is re-create the directory and restart the system. Your objects on the LaunchPad will return. The Nowhere directory may be called Nowhere1 or Nowhere2, depending on the name of your Desktop. For example, if the physical name for the Desktop directory is Desktop1, then your Nowhere directory may have a physical name of Nowhere1.

Summary

As both the author of this chapter and the developer of the LaunchPad, I hope you find it a useful tool and that it improves your enjoyment of OS/2 Warp as much as it does for me. Stay plugged into OS/2 Warp for future developments; there are many improvements planned for the LaunchPad in future releases.

Peter G. Magid is an Advisory Programmer on the Workplace Shell Development Team. As the User Interface Design Lead for the OS/2 Workplace Shell, Peter has had chief design responsibilities for what users see and how they interact with the Workplace Shell. Peter also plays an active role in reviewing and consulting on the content and direction of the Workplace Shell programming model, and he developed the code for the LaunchPad.

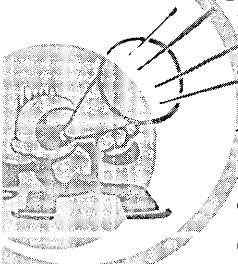
Chapter 5

Workplace Shell Object Types

Hmmm, They SEEM Real, But...

I am sure you are familiar with all of the icons floating around on the Desktop, but do you really know what they represent? In this chapter, I will introduce you to the various object classes you will see when using the Workplace Shell and will explain what they stand for and what they can do for you.

Workplace Shell Objects



Each of the icons you see represents an *object* in the Workplace Shell. But, what is an object?

An object is something that has data associated with it, called *properties,* and that has a particular set of behaviors. For example, all objects in the Workplace Shell have a name, or title, and an icon. These are properties that are common to all objects. Most objects can be copied, moved, or deleted. These are behaviors that are common to all objects, even though some objects have these behaviors disabled.

In an object-oriented system, such as OS/2 Warp, each object in the system is derived from an *object class* that describes the properties and behaviors that all objects derived from that object class can have. The object class defines what properties and behaviors an object has and the object defines the values for the properties.

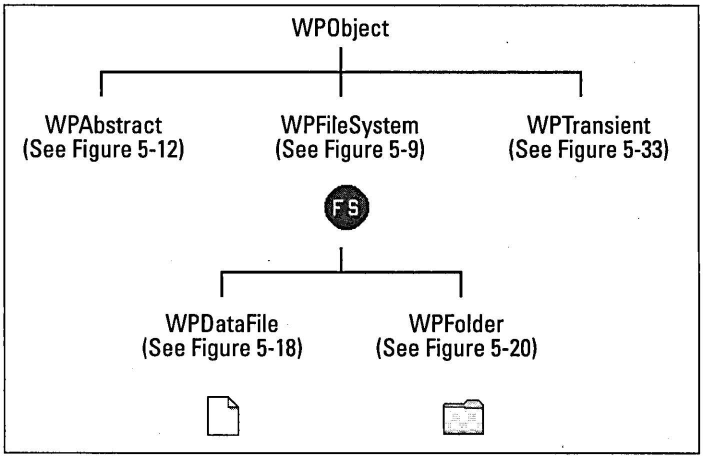
Some of the properties associated with an object are immediately visible to you when you see the object. For example, the object's title and icon are visible in the folder that contains the object. Other properties, such as the viewing characteristics of a folder, can be deduced from the behavior of the object. Others, such as the object identifier, are totally hidden and only affect the use of the object. You will begin to appreciate all you can do with an object once you understand what properties it has.

You can change some of the properties directly, such as changing an object's title by directly editing it; you can change some properties from the Settings Notebook for the object; and you can change others using a REXX script. (See Chapter 8 for information on how to use REXX to change the properties of a Workplace Shell object.)

Object classes are used as building blocks for other object classes. An object class that is built on top of another object class is called a child class, and it inherits all of the characteristics of its parent class. This allows for a hierarchy of object classes, where each object class only adds the properties and behaviors necessary to distinguish it from its parent class.

I am sure this is as clear as mud right now, but as you read this chapter, it should become clearer. To help ease the confusion somewhat, take a look at Figure 5-1. It shows the overall hierarchy of object classes supported by the Workplace Shell. Throughout this chapter, you will find more detailed hierarchies showing each branch of the WPObject family tree.

Figure 5-1.
Workplace Shell Object Class Hierarchy - Overall



WPObject — The Parent of All Workplace Shell Object Classes

The very top of the Workplace Shell object class hierarchy is a class called *WPObject.* This is the parent of all Workplace Shell object classes. It defines the properties and behaviors common to all Workplace Shell objects. You will probably never encounter an object whose class is WPObject – it is just the building block for all other object classes. An object of class WPObject wouldn't be very interesting, since it couldn't do very much.

WPObject – properties

The properties defined by the WPObject class include

- ✘ Class Title — This is the default value for the name of any object created from this class. You can usually see this name under the icon for the template for this object class in the Templates Folder. Normally you assign a different title to the object when you create it.
- ✘ Class Icon — This is the picture that represents each object created from this class. An object class may define its own Class Icon. Each object created from the object class may define its own icon, or may use the default icon defined for the object class. For example, many folders use the Default Folder Class Icon, while some special folders, like the OS/2 System Folder, have their own distinctive icon. You cannot directly change the Class Icons, but you can change the icon associated with any object. A Folder class may actually have two Class Icons – the normal Class Icon and the animation Class Icon, which is displayed when an object of that class is open.
- ✘ Class Style — This is a set of characteristics for all objects that are created from this object class. These characteristics include
  - Whether or not a template is created for this object class
  - Whether or not objects of this class can be moved to another folder
  - Whether or not objects of this class can be copied to another folder
  - Whether or not you can create a shadow for objects of this class
  - Whether or not objects of this class can be deleted or dragged to the Shredder object

- Whether or not objects of this class can be dragged anywhere
- Whether or not objects of this class can accept other objects being dropped on them
- Whether or not objects of this class are visible
- Whether or not the title can be changed for objects of this class
- Whether or not objects of this class can be printed or can be dragged to a Printer object
- Whether or not you can display the Settings Notebook for objects of this class

  Most of these characteristics, except for the template characteristic, can be overridden by the object itself. For example, an object class may have a Class Style that prohibits objects of that class from being printed, but an individual object of that class may have a style that says the object can be printed.

✘ Title — This is the name of the object. You can see this name under or next to the icon in an open view of a folder containing the object. You can directly change this name by pointing the mouse at either the icon or the title, holding the Alt key on the keyboard, and pressing mouse button one. The title is replaced with an edit control containing the title, which allows you to change the object's name. When you are finished editing the name, move the mouse pointer outside the edit control and press either mouse button. The title is also found on the General page of the Settings Notebook for the object. (I will describe the Settings Notebook pages for an object after the description of the object class.) The title can also be changed using the TITLE= setup string. See Chapter 8 for a description of how to use setup strings.

✘ Icon — This is the picture that represents the object. Each object may define its own icon, or may use the default icon defined for its object class. The icon is visible in an open view of a folder that contains the object and it can be changed from the first General page of the Settings Notebook or by using the ICONFILE= or ICONRESOURCE= setup strings. A folder may have a second icon, the animation icon, which is visible in an open view of its containing folder when the folder is open. The animation icon can be changed from the second General page of the Settings Notebook or by using the ICONFILE= or ICONRESOURCE= setup strings.

- ✘ **Default Icon Position** — This is the position within an open view of the containing folder where the icon for this object will be displayed. The Default Icon Position can be set using the ICONPOS= setup string.
- ✘ **Style** — This is a set of characteristics for the object that indicates the behaviors that are allowed for the object. These characteristics are defined by the child object class, and most of them can only be specified using the appropriate setup string (which follows the characteristic in parentheses). These characteristics include
  - Whether or not the object is a template (TEMPLATE= setup string or the General page of the Settings Notebook)
  - Whether or not the object can be moved to another folder (NOMOVE= setup string)
  - Whether or not the object can be copied to another folder (NOCOPY= setup string)
  - Whether or not you can create a shadow (see the description of the WPShadow class later in this chapter) for the object (NOLINK= or NOSHADOW= setup strings)
  - Whether or not the object can be deleted or dragged to the Shredder (NODELETE= setup string)
  - Whether or not the object can be dragged anywhere (NODRAG= setup string)
  - Whether or not the object can accept other objects being dropped on it (NODROP= setup string)
  - Whether or not the object is visible (NOTVISIBLE= setup string)
  - Whether or not the object's title can be changed (NORENAME= setup string)
  - Whether or not the object can be printed or dragged to a printer (NOPRINT= setup string)
  - Whether or not you can display the Settings Notebook for the object (NOSETTINGS= setup string)

Most of these characteristics, except for the template and visibility characteristics, affect the list of items appearing in the pop-up menu for the object. (I will describe the pop-up menu options after the description of the object class.)

- ✗ **Help Information** — This is the name of the help library (HELPLIBRARY= setup string) and the help panel (HELPPANEL= setup string) that define the default help for the object. This panel is displayed if you press the F1 key while the object's icon is highlighted.
- ✗ **Object Identifier** — This is a name that you can assign to an object using the OBJECTID= setup string. This name can later be used to reference this object through REXX commands. Many objects that are created during the installation of OS/2 Warp have predefined object identifiers. See Chapter 8 for the list of predefined object identifiers.
- ✗ **Minimized Window Behavior** — This specifies the behavior of the object when you click on the **minimize** push button in an open view of the object. The behavior may be "hide," "minimize to Desktop," or "minimize to the Minimized Window Viewer." This property may be specified on the first Window page in the Settings Notebook for the object or by using the MINWIN= setup string. If this property is not specified for an object, it defaults to the system default value that is specified on the first Window page of the System object (see Chapter 7).
- ✗ **Button Appearance** — This specifies whether a **hide** push button or a **minimize** push button appears in the title bar of an open view of the object. This property may be specified on the first Window page in the Settings Notebook for the object or by using the HIDEBUTTON= setup string. If this property is not specified for an object, it defaults to the system default value that is specified on the first Window page of the System object.
- ✗ **Concurrent View Behavior** — This specifies what happens when you try to open a view of the object when the specified view is already open. The behavior may be "open another view" or "display existing view." This property may be specified on the first Window page in the Settings Notebook for the object or by using the CCVIEW= setup string. If this property is not specified for an object, it defaults to the system default value that is specified on the first Window page of the System object.

- ✗ Default View — This specifies the view that is opened if you double-click on the icon for the object. This property may be specified on the Menu page in the Settings Notebook for the object or by using the DEFAULTVIEW= setup string. If this property is not specified for an object, it defaults to the system default value, which, for a folder, is specified on the third Window page of the System object. For other objects, the default value is defined by the specified object class.

- ✗ Container — This specifies the folder in which the object resides. This property can be changed only by moving the object to another folder.

WPObject – behaviors

The behaviors that are defined by the WPObject class include

- ✗ **Pop-up menu** — You can display the pop-up menu for any object by moving the mouse over the object and pressing mouse button two. You can select an item from the pop-up menu to specify an action to be performed by the object by clicking mouse button one on the item. I will describe the pop-up menu items supplied by each object class after the description of the class.

- ✗ **Settings Notebook** — You can open the Settings Notebook for any object by choosing the **Settings** item from the object's pop-up menu or by pressing and holding the Alt key while double-clicking mouse button one on the icon for the object. The Settings Notebook contains pages that can be used to display and change the properties of the object. Not all properties are represented in the Settings Notebook. Some properties are specified by the object class and can't be overridden; other properties can only be specified by setup strings when the object is created. I will describe the Settings Notebook pages supplied by each object class after the description of the class.

- ✗ **Move To Another Folder** — You can move an object from the folder it currently resides in to another folder. An object can be moved either by choosing the **Move...** item from the object's pop-up menu or by dragging the object from one folder to another. See Chapter 3 for information on how to use drag and drop or pickup and drop to move an object.

- ✗ **Copy To Another Object** — You can copy an object from the folder it resides in to another folder. An object can be copied either by choosing the Copy... item from the object's pop-up menu or by dragging the object from one folder to another while holding the Ctrl key. The difference between moving an object and copying it is that, after moving the object, it no longer resides in its original folder, while after copying the object the original object still resides in its original folder and a new object that looks just like the original is created in the destination folder.

- **Create Another Object** — You can create another object of the same object class by selecting the **Create another...** item from the object's pop-up menu or by copying a template object.
- **Create A Shadow** — You can create a shadow of an object either by choosing the **Create shadow...** item from the object's pop-up menu or by dragging the object from one folder to another while holding the Ctrl and Shift keys. See the explanation of the WPShadow class later in this chapter for information about using shadows.
- **Delete** — You can delete (destroy) an object either by choosing the **Delete...** item from the object's pop-up menu or by dragging the object to the Shredder.
- **Print** — You can print an object either by choosing the **Print...** item from the object's pop-up menu or by dragging the object to a printer.

Note

The Printer object doesn't necessarily print the contents of the object on paper. There are some products, such as the FaxWorks/2 program that comes in the BonusPak, that create Printer objects that don't print — they may fax the contents of the object, capture it to disk, or do anything else that the Printer object is designed to do with the object.

- **Drag And Drop** — You can drag an object and drop it on another object by moving the mouse over the object's icon, pressing and holding mouse button two, moving the mouse over the target object's icon, and releasing mouse button two. The default operation performed when you drop an object on another object is defined by the target object. For example, the default operation when dragging an object to a folder is "move the object into this folder;" the default operation when dragging an object to the Shredder is "delete the object;" and the default operation when dragging an object to a Printer object is "print the object." The operation to be performed can be specified by holding one or more keyboard keys while dragging the object. The Shift key indicates "move," the Ctrl key indicates "copy," and the combination of the Shift and Ctrl keys indicates "create shadow." See Chapter 3 for a complete explanation of how to use the drag and drop facility.

- ✗ Open A View — You can open a view of the object by selecting the **Open** item from the object's pop-up menu. This displays another menu that lists the possible views supported by this object. You can select the view you want to open from this cascade menu. You can open the default view for the object by double-clicking mouse button 1 on the object's icon. You can open the Settings Notebook for the object by holding the Alt key on the keyboard while double-clicking mouse button 1 on the object's icon or by selecting the **Settings** item from the object's pop-up menu.

- ✗ Save And Restore Instance Data — The WPObject class defines the structure for saving and restoring the data that make up the personality of the individual object, but doesn't implement any such mechanism. This is one of the reasons you will never see an object of type WPObject – it has no way of saving its persistent form. The actual mechanism for saving and restoring the instance data is defined in the base classes described later in this chapter.

WPObject — pop-up menu items

The WPObject class defines the following items in the pop-up menu for the object. Note that not all options are valid for all objects. The only items that are actually displayed in the pop-up menu are the items that are valid for the object. Pop-up menu items may be removed from the pop-up menu because the object's style prohibits their use or because the object class suppresses the items.

- ✗ Open — The **Open** pop-up menu item always points to a cascade menu containing a list of views supported by the object. You can open a particular view by selecting the arrow to the right of the **Open** item, then selecting the name of the view that you want to open. You can open the default view for the object (which is defined by the actual object itself) by selecting the **Open** item, or by double-clicking mouse button one on the object's icon.

- ✗ Settings — You can open the Settings Notebook for an object by selecting the **Settings** item in the pop-up menu.

- ✗ Help — You can request help for the object by selecting the **Help** item in the pop-up menu. This displays the default help panel for the object, from which you can request other help information. You can get a list of special help topics by selecting the arrow to the right of the **Help** pop-up menu item. For most objects, the special topics include

- **Help index** — Displays an index of help topics available for the object
- **General help** — Displays the default help panel for the object
- **Using help** — Displays information explaining how to use the help system in OS/2
- **Keys help** — Displays information about how to use the keyboard to navigate your way through OS/2 objects
- **Product information** — Displays information about the product that created the object class

✘ **Create another...** — You can create another object similar to the current object by selecting the **Create another...** pop-up menu item. This displays the Create Another dialog box, as shown in Figure 5-2. If you select the arrow to the right of the **Create another...** pop-up menu item, the system displays a list of object classes that are derived from the same class as the current object. You can select which object class you want to use as a template for the new object.

✘ **Copy...** — You can create a copy of the object by selecting the **Copy...** pop-up menu item. This presents the Copy dialog box, as shown in Figure 5-3.

✘ **Move...** — You can move the object to another folder by selecting the **Move...** pop-up menu item. This displays the Move dialog box, as shown in Figure 5-4.

✘ **Create shadow...** — You can create a shadow of the current object by selecting the **Create Shadow...** pop-up menu item. This displays the Create Shadow dialog box, as shown in Figure 5-5.

✘ **Delete...** — You can delete the current object by selecting the **Delete...** pop-up menu item. This displays the Delete Objects dialog box, as shown in Figure 5-6.

✘ **Pickup** — You can start a pickup/drop operation by selecting the **Pickup** pop-up menu item. The object is "picked up," as if you had pointed to the object, pressed and held mouse button two. The object can now be "dragged" to another object and "dropped" on that object by selecting the **Drop** pop-up menu item, which is equivalent to releasing mouse button two. This lets you perform a drag and drop operation on an object without actually holding mouse button two throughout the entire operation. Since the mouse buttons are free during the

Pickup/Drop operation, you can open other folders while the operation is pending, allowing you to drop the object in a folder you wouldn't have been able to see when you originally picked up the object.

✗ **Cancel drag** – You can cancel a Pickup/Drop operation by selecting the **Cancel drag** pop-up menu item.

✗ **Drop** – You can complete a Pickup/Drop operation by selecting the **Drop** pop-up menu item of the object you want to drop the "picked up" object on. You can select the actual operation to be performed when dropping the object by selecting the arrow to the right of the **Drop** pop-up menu item. This presents a list of possible operations, which includes

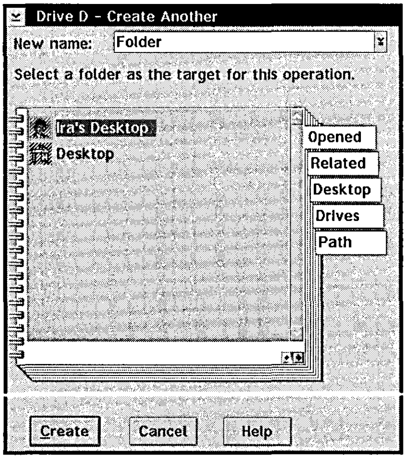
- **Copy** — to make a copy of the object in the target folder
- **Move** — to move the object from its current location to the target folder
- **Create shadow** — to create a shadow of the object in the target folder
- **Cancel drag** — to cancel the Pickup/Drop operation

If you just select the **Drop** pop-up menu item, the default operation is performed.

Create Another dialog box

The Create Another dialog box is used to create another object of the same or similar class as the current object (see Figure 5-2). This creates the same object as if you opened the Templates Folder and dragged a copy of the specified object class's template to a folder.

Figure 5-2. Create Another Dialog Box


Drive D - Create Another

New name: Folder

Select a folder as the target for this operation.

Specify the title for the new object in the New name entry field. This field is initially set to the default title for the object class. For example, if you are trying to create a new folder, the default title is "Folder."

You must specify the destination folder that is to contain the new object. The tabbed pages allow you to specify this folder in a variety of ways. You select the tab that represents the set of folders you want to choose from, then select the destination folder from that page. The sets of folder you can choose from are

- **Opened** — This shows all of the folders that you currently have open.

  Tip

  If you know you are going to create, move, or copy one or more objects into a folder, you can open the folder before trying to create, move, or copy the objects. The selected folder will then appear on the Opened Folders page, making it easier to choose it as the target.

- **Related** — This shows a Tree view of the current drive. In order to select a folder, just click on its icon in the Tree view. If you don't see the folder you want, you can expand any branch of the tree by clicking on the plus sign in front of any icon.
- **Desktop** — This shows a Tree view starting at the current Desktop Folder. You can click on the plus sign in front of any icon to expand its branch of the tree until you see the folder you want, then click on the folder to select it.
- **Drives** — This shows the set of drives (disk, diskette, CD-ROM, and so on) that are available. You can click on the plus sign in front of any of the drive icons to expand the tree for that drive. Keep expanding the tree until you find the folder you want, then click on it to select it.
- **Path** — If you want to specify the name of the folder directly, without going through any of the sets of folders, select the Path page. You can directly enter the fully qualified path name (including the drive letter) of the target folder in the entry field.

After you have filled in the name of the new object and selected the destination folder, click on the **Create** push button to create the new object. If you decide you really don't want to create a new object, click on the **Cancel** push button.

Copy dialog box

The Copy dialog box is used to make a copy of an object (see Figure 5-3). If the selected object is a folder, the new folder will be an exact copy of the original folder, including its contents. All objects contained within the original folder will be copied into the new folder. After the copy operation is complete, the original object still exists in its original folder and a new object, which has the same behavior, properties and contents of the original object, now resides in the destination folder. However, two properties are different: The Object Identifier stays with the original object (the new object has no Object Identifier property) and the Container of the new object is the destination folder.

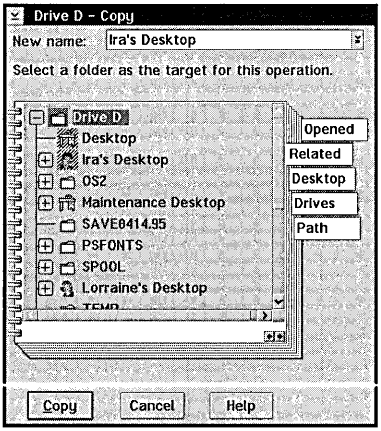
Warning

There is one object supplied as part of OS/2 that performs the copy operation incorrectly: the LaunchPad. If you copy an existing LaunchPad object, the copy looks exactly like the original. The problem is that the Shadow objects contained within the LaunchPad (the objects on the LaunchPad buttons and in the drawers) are not copies of the original Shadows — they are the original Shadow objects. If you delete one of these objects from the original LaunchPad, the corresponding button or drawer location is replaced with a question mark icon in the copy. The same is true in reverse — if you delete an object from the copy it is no longer available on the original.

Tip

There are two simple solutions to this problem. If you want to create a new LaunchPad object, use a REXX script to create it. Alternatively, you can copy a LaunchPad object, then delete all objects from both LaunchPads. When you have deleted all objects from the LaunchPad, it automatically repopulates itself with the default set of objects. Once you have done this, the two LaunchPad objects are independent and you can customize both objects any way you want.

Figure 5-3.
Copy Dialog Box



Drive D - Copy

New name: Ira's Desktop

Select a folder as the target for this operation.

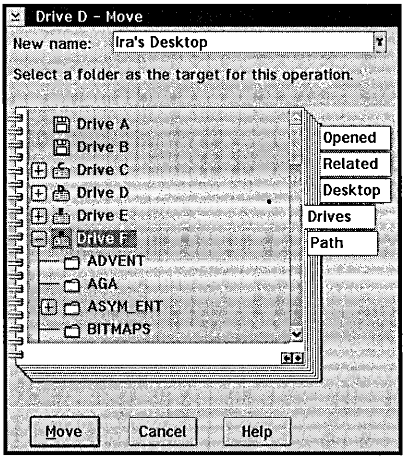
Specify the title for the copy of the object in the New name entry field. This field is initially set to the title of the object being copied. You may leave the name alone if you want the new object to have the same title as the original object or you may specify a new title. If the object being copied is a file system object, the title also specifies the name of the file or directory that corresponds to the object.

You must specify the destination folder that is to contain the new object. You can specify the destination folder in the same manner as for the Create Another dialog box from the Opened, Related, Desktop, Drives, or Path page.

After you have filled in the name of the new object and selected the destination folder, click on the **Copy** push button to copy the object. If you decide you really don't want to copy the object, click on the **Cancel** push button.

Move dialog box

The Move dialog box is used to move an object from its current location to a new folder (see Figure 5-4). If the selected object is a folder, all objects contained within the folder are moved also. After the copy operation is complete, the original object no longer exists in its original folder. The object you now see in the destination folder is the original object, not a new object. It has the same behavior, properties, and contents that it had before you moved it. However, one property is different: The container of the object is changed to the destination folder.



Drive D - Move

New name: Ira's Desktop

Select a folder as the target for this operation.

Figure 5-4.
Move Dialog Box

You can specify a new title for the object in the New name entry field. This field is initially set to the title of the object being moved. You may leave the name alone if you don't want to change the name of the object or you may specify a new title. If the object being moved is a file system object, the title also specifies the name of the file or directory that corresponds to the object.

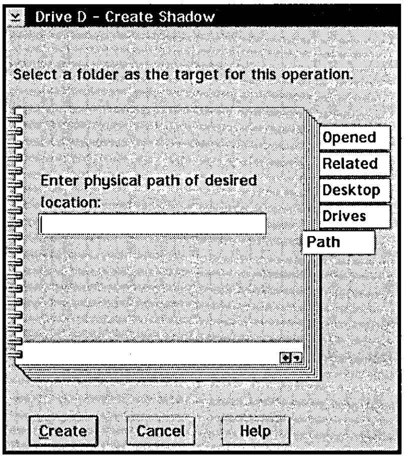
You must specify the new folder that is to contain the object. You can specify the destination folder in the same manner as for the Create Another dialog box from the Opened, Related, Desktop, Drives, or Path page.

After you have filled in the new name of the object (if any) and selected the destination folder, click on the **Move** push button to move the object to the new folder. If you decide you really don't want to move the object, click on the **Cancel** push button.

Create Shadow dialog box

The Create Shadow dialog box is used to create a shadow of an object (see Figure 5-5). After the create shadow operation is complete, the original object still exists in its original folder and a new object (which is linked to the original object) now resides in the destination folder.

Figure 5-5. Create Shadow Dialog Box



Drive D - Create Shadow

Select a folder as the target for this operation.

Enter physical path of desired location:

A shadow of an object must have the same title as the original object, so there is no New name entry field on the Create Shadow dialog box.

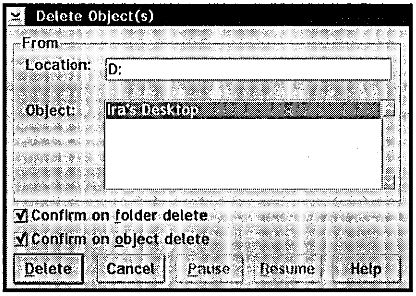
You must specify the folder that is to contain the shadow object. You can specify the destination folder in the same manner as for the Create Another dialog box from the Opened, Related, Desktop, Drives, or Path page.

After you have selected the destination folder, click on the **Create** push button to create the shadow of the original object. If you decide you really don't want to create a shadow object, click on the **Cancel** push button.

Delete Objects dialog box

The Delete Objects dialog box (see Figure 5-6) is displayed if you specify to delete an object and you previously requested to have the Workplace Shell confirm deletion requests (see the discussion on the System object in Chapter 7).

Figure 5-6. Delete Objects Dialog Box



Delete Object(s)

From

Location: D:

Object: Ira's Desktop

☑ Confirm on folder delete

☑ Confirm on object delete

The Location field specifies the name of the folder that contains the object being deleted.

The Object list box specifies the names of the objects that are being deleted.

You can request confirmation of the deletion of each object or folder by checking the appropriate check boxes below the Object list box.

If you want to delete the selected objects, click on the **Delete** push button. If you change your mind, click on the **Cancel** push button to terminate the deletion operation. Once you start the deletion operation, you may click on the **Pause** push button at any time to temporarily stop the operation after the current object is deleted. If there are more objects to be deleted, you can click on the **Resume** button to continue the operation, or click on the **Cancel** button to stop the Workplace Shell from deleting any more objects.

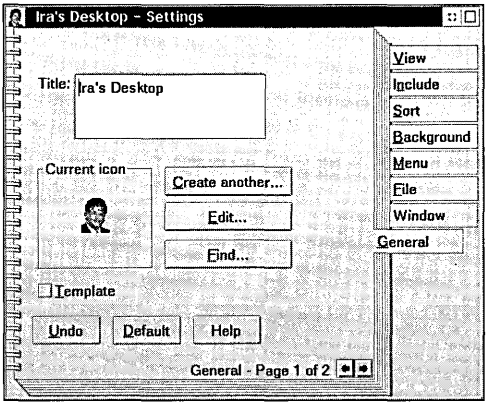
WPObject — Settings Notebook pages

The Settings Notebook pages defined by the WPObject class are available to all objects.

General pages

The first General page displays the title, icon, and template property for an object (see Figure 5-7). If the object is a folder, there is a second General page that displays the animation icon for the folder.

Figure 5-7. WPObject - General Page 1



Ira's Desktop - Settings

Title: Ira's Desktop

Current icon

□ Template

General - Page 1 of 2

You can change the title of the object by changing the data in the Title: field. If you want the title to display as multiple lines in the folder that contains the object, just press the Enter key at the appropriate place within the title.

You can modify the object's icon in four ways.

1 The object's icon is displayed in the Current icon box. You can change the icon directly by dragging any other icon that you see into the current icon box. The icon you drag will replace the object's icon.

2 You can select the **Create another...** push button. This invokes the Icon Editor with a blank icon. You can draw the icon that you want, then save the icon. The icon that you save will replace the object's icon. If you exit the Icon Editor without saving an icon, the object's icon is not changed.

3 You can select the **Edit...** push button. This invokes the Icon Editor with the object's current icon. You can modify the icon anyway you want, then save the icon. The modified icon will replace the object's icon. If you exit the Icon Editor without changing or saving the changed icon, the object's icon is not changed.

If you edit an object's icon, the Workplace Shell will always use the modified icon when displaying that object, not the icon associated with the object. This can be particularly useful for a Program Reference object that points to a program that resides on a removable media (such as a diskette, CD-ROM, or network-attached directory). If you let the system use the default icon, it must access the program in order to display its icon. If the program is not available (for example, you have the wrong CD-ROM inserted in the CD-ROM drive), the Workplace Shell won't be able to display the correct icon. However, if you edit the icon and save it (even with no changes), the icon is stored as part of the Program Reference object and this icon is always available.

4 You can select the **Find...** push button. This invokes the Find Objects dialog box to search for icon files (that is, files whose file extensions are .ICO). You can specify the folder where to start the search, then press the **Find...** push button. If you select an icon from the Find Results Container and press the **OK** push button, the selected icon will replace the object's icon. If you close the Find Results Container without selecting an icon, the object's icon is not changed.

You might have noticed that a Folder object may have a second icon that is displayed when the folder is opened. This is called the *animation icon.* You can change the animation icon on the second General page of the folder's Settings Notebook. If there is no animation icon assigned to the folder (the same icon is displayed when the folder is open as when the folder is closed), the Animation icon box is empty. You can still drag an icon into the box — this will assign an animation icon to the folder. (This is a rather strange case. Normally, an object class has no knowledge of its children classes — only its parent class. However, since the WPObject class provides the first General page for all objects, it also provides the second General page for folder objects, even though this functionality should really be in the WPFolder class.)

If you drag an icon representing a folder that has an animation icon assigned to the Current icon box on the first General page of a folder's Settings Notebook, both the object's icon and the folder's animation icon are copied to the target folder. If you drag any other icon to the Current icon box on the first General page of a folder's Settings Notebook, the object's icon is copied and the folder's animation icon is set to be a copy of the object's icon. This allows you to edit the icon to create an animation icon based on the object's icon. If you drag an icon to the Animation icon box on the second General page of a folder's Settings Notebook, only the animation icon is changed — the object's icon is not affected.

The Template check box, if present, can be used to specify that this object is a template. You can create any object you want, with any contents, then convert the object into a template. For example, you might create a folder, customize it for a particular view, populate it with some objects, then convert it into a template. When you create an object based on the template, either by dragging a copy from the template or by selecting the **Create another...** item from the template's pop-up menu, the new object will have the same settings and a copy of the contents of the template object. If you want to change the template, you can remove the check from the Template check box, modify the template any way you want, then turn the Template option back on.

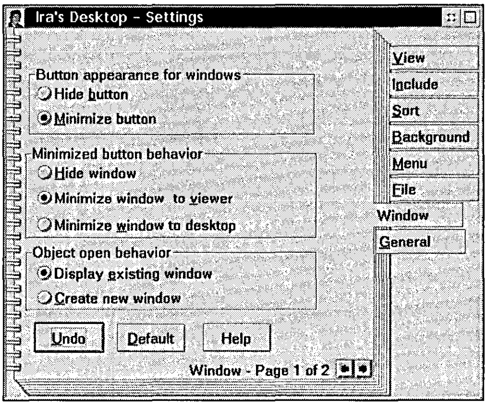
The **Undo** push button can be used to change the properties of the object back to the values that the object had when the General page was first opened. If you press the **Undo** push button on the first General page, the object's icon is restored. If you had previously dragged another folder's icon into the Current icon box for a folder object, pressing the **Undo** push button on the first General page will also restore the object's animation icon. If you press the **Undo** push button on the second General page of a folder object's Settings Notebook, only the folder's animation object is restored.

CD-ROM

The **Default** push button can be used to reset the properties of the object to the defaults for the object class. For some objects, like the OS/2 System Folder, the default icon may not be the icon that was originally displayed. The OS/2 System Folder is a normal folder object with a special icon assigned to it. If you press the **Default** push button for the OS/2 System Folder, the object's icon is replaced by the Folder Class icon, which is the generic folder icon, not the OS/2 System Folder icon. The same is true for the Information, System Setup, Command Prompts, and Multimedia folders. If you inadvertently lose the special icons for these folders, you can restore them by running a REXX file, which is contained on the CD-ROM included with this book.

Window page

The Window page lets you specify the values for the behavior of open views of the object (see Figure 5-8).



Ira's Desktop - Settings

Button appearance for windows

○ Hide button

● Minimize button

Minimized button behavior

○ Hide window

● Minimize window to viewer

○ Minimize window to desktop

Object open behavior

● Display existing window

○ Create new window

Figure 5-8.
WPObject - Window Page

The Button appearance for windows option specifies whether windows for open views of this object are to contain hide push buttons or minimize push buttons. You may select one of the following radio push buttons:

- ✘ **Hide push button** — Windows will contain a **Hide** push button on the title bar, and they will be removed from the screen (hidden) whenever you press the **Hide** push button
- ✘ **Minimize push button** — Windows will contain a **Minimize** push button on the title bar, and they will be minimized to the Desktop, minimized to the Minimized Window Viewer, or hidden whenever you press the **Minimize** push button on the window's title bar, based on the Minimize button behavior option

If you don't select either of these options, the system default value, which is specified on the first Window page of the System object, specifies which push button appears in all windows for the object.

The Minimize push button behavior option specifies the behavior of Minimize buttons on this object's windows. You may select one of the following radio buttons:

- ✘ **Hide window** — Windows will be hidden when you click on their **Minimize** push buttons
- ✘ **Minimize window to viewer** — Windows will be minimized to the Minimized Window Viewer when you click on their **Minimize** push buttons
- ✘ **Minimize window to desktop** — Windows will be minimized to the Desktop when you click on their **Minimize** push buttons

If you don't select one of these options, the system default value, which is specified on the first Window page of the System object, specifies the behavior of the minimize push button.

The Object open behavior option is used to specify what happens when you specify to open a view of an object when a view of that type is already open. You may select one of the following radio push buttons:

- **Display existing window** — When you specify to open a view of an object that is already open, the system brings the view's window to the front of the screen and gives focus to the window
- **Create new window** — When you specify to open a view of an object that is already open, the system opens another view of the object

The system default value, which is specified on the first Window page of the System object, is used for any object for which you haven't specified one of the Object open behavior options on the first Window page for the object.

The **Undo** push button can be used to change the window properties of the object back to the values that the object had when the first Window page was first opened.

The **Default** push button can be used to reset the window properties of the object to the system default values, which are specified on the first Window page of the System object.

The Base Classes — WPAbstract, WPFileSystem, and WPTransient

As I mentioned in the previous section, the WPObject class defines the structure for saving and restoring the data associated with an object, but doesn't implement any mechanism to do so. The three base classes provide that mechanism. (Actually, only two of the base classes store their persistent data. The third base class is provided for symmetry.)

Note

You will probably never see an object that is derived directly from the WPAbstract, WPFileSystem, and WPTransient classes. These classes are similar to the WPObject class because they are building blocks on which useful object classes are built.

WPAbstract — the base class that lives in the OS2.INI file

The main characteristic of the WPAbstract class and all of its derivatives is that these objects must remember some data about themselves, but have no intrinsic place to store that data. The Workplace Shell saves the data for WPAbstract objects in the OS2.INI file.

Uncensored

This provides some advantages and some drawbacks. The access to the OS2.INI file is fast, since it is cached by the system. In fact, when the Workplace Shell thinks it is reading from the OS2.INI file (or writing to it), it is really getting the data from a memory-resident copy of the file. The system automatically synchronizes the real file with the memory-resident copy periodically, when the system is not busy. And, since the data is really in memory, the access is made even faster by using special techniques to access the data randomly rather than serially.

Warning

However, there are several drawbacks to using the OS2.INI file to save the data about WPAbstract objects. Since the data resides in the OS2.INI file, it is not portable. That means that you can't copy an abstract object, like a Program Reference object, to a diskette and move it to another system. If the Workplace Shell allowed you to copy an abstract object to a diskette, you might think you got something on the diskette. If you then looked at that diskette from another OS/2 system, you wouldn't see anything, since that object is not represented in the OS2.INI file on the second system. We therefore don't allow you to copy an abstract object to any removable medium, such as a diskette, read-write optical disk, or a network-attached drive.

A second possible problem with using the OS2.INI file is that the file might become large, fragmented, and subject to corruption by random errors on the storage medium or by system failures. If your system ever hung and you had to cold boot the system (by pressing the dreaded Reset button or by powering your system unit off and on), you might have had the awful experience of CHKDSK announcing problems with one or more of your files, which it "fixes" for you. If one of the files CHKDSK fixes is the OS2.INI file, you might lose some, or all, of your abstract objects.

Uncensored

Fortunately, this doesn't happen very often, and you can protect yourself by periodically letting OS/2 create an archive for your system (see Chapter 11).

The WPAbstract class doesn't have any new properties of its own. There are no new Settings Notebook pages for abstract objects and they don't have any new setup strings. The only thing the WPAbstract class really provides is the mechanism for saving the object's data in the OS2.INI file.

WPFileSystem — the base class for file system objects

The main characteristic of the WPFileSystem class and all of its derivatives is that these objects represent directories and files that reside in the file system. Since each WPFileSystem object is associated with a real file system entity, the Workplace Shell has a good place to save the data for WPFileSystem objects – in the file system. Of course, the Workplace Shell doesn't want to modify the data you have stored in the real file, so it saves the data about the object in the *extended attributes* for the file or the directory.

Uncensored

The good news is that because the object data for WPFileSystem objects is stored in the extended attributes for the file, the WPFileSystem objects are portable. You can copy a file system object to a diskette, take it to another OS/2 system, and see your original object, with all of its properties intact. The bad news is that it is much slower to update the extended attributes for a file than it is to update the OS2.INI file used by abstract objects. The extended attributes are subject to corruption like any other file system data.

Warning

There is one other problem with the WPFileSystem class. Not all file systems support extended attributes. For example, the CD-ROM File System and the Microsoft Windows NT File System don't support extended attributes. This makes it impossible for the WPFileSystem class to store its properties. Any WPFileSystem object that resides on a file system that doesn't support extended attributes can only have the default properties for that object. This means you can't assign a special icon for the file or use a long file name. Also, if you try to copy a WPFileSystem object from a file system that supports extended attributes (like the FAT or HPFS File Systems) to a file system that doesn't support extended attributes (like the Microsoft Windows NT File System), you will lose all of the properties that you customized.

Uncensored

You might wonder how you can have a file system object stored on a Microsoft Windows NT File System when running under OS/2. It's simple. IBM LAN Server 4.0, which you can install on OS/2 Warp, allows an OS/2 client to attach to a Microsoft Windows NT server. If you do this, some of your LAN-attached drives may not support extended attributes.

File systems

OS/2 Warp has built-in support for three different file systems and allows other file systems to be installed on the system. Each of these file systems has slightly different characteristics.

Uncensored

The FAT file system. The File Allocation Table (FAT) File System originated in the first version of the Disk Operating System (DOS), way back in 1981. When the FAT File System was designed, the personal computer at that time only had diskette drives that could hold 320K of data. (That's right – they didn't even hold 360K. The change from 320K 5$^{1}/_{4}$-inch diskettes to 360K diskettes came about a year after the IBM Personal Computer was introduced.) The first popular fixed disks held a whopping 10MB of data. The FAT File System was designed with these requirements in mind; it wasn't even envisioned in those days that we would someday have 128MB optical disks and 4GB fixed disks in personal computers. But, that is the reality today.

The FAT File System is based on a fixed size table that contains at most 65,536 entries that have to represent all of the space on the disk. Each entry in the table defines one *cluster*, which represents a fixed-size chunk of the disk. Since there are a fixed number of clusters, as the size of the disk increases, the size of each cluster must increase also.

The smallest piece of the disk that the FAT File System can allocate to a file is a cluster. If you have a hard disk that is smaller than 128MB in size, the cluster size will be 2048 bytes. This means that a file that contains 1 byte of data will occupy 2048 bytes of space. This isn't so bad. If you have a 1GMB fixed disk, the cluster size is 16,392 bytes. Things rapidly go from bad to worse!

There are several potential problems with the design of the FAT File System. Since the allocation of space on the disk is controlled by one large table, if the table becomes corrupted, the integrity of the data contained in your files cannot be guaranteed. For example, if you run the CHKDSK utility you may get messages indicating "cross-linked sectors" found. This indicates that two files were assigned to the same entry in the file allocation table. Another problem is that clusters can be allocated from the file allocation table but never assigned to a file. This results in "lost clusters."

The original directory structure for the FAT File System allowed for 8 bits to be used for file attributes. When OS/2 was designed, we needed much more space for attributes, so the concept of the *extended attributes* was born. The extended attributes are logically an extension to the file directory entry. However, since there was no place to store the data in the directory, we implemented extended attributes in a separate file, named "EA DATA. SF," which is a hidden file kept in the root directory of the disk. The file directory contains an index that tells the file system where the extended attributes (EAs) for this file start in the EA file. Again, this is prone to problems. If the index value gets lost, the file will lose its extended attributes. If the EA file gets corrupted, you can lose all of the extended attriubutes for all files on the disk. Since WPFileSystem objects keep all of their object data in the extended attributes, if the EA file gets corrupted, you can lose all of the object data for every object stored on that partition. This is not a likely occurrence, but it has happened. We have seen cases where the extended attribute file was corrupted and fixed by CHKDSK and the results were catastrophic. Since the data stored in the extended attributes included the class name, when the extended attributes were lost all directories became plain folders, including the Desktop.

Uncensored

The only way to protect yourself from this kind of loss is to back up your files onto another volume or to tape (using an OS/2-based backup program that understands extended attributes).

Warning

The extended attribute concept itself causes problems with DOS and Windows programs. DOS and Windows do not support extended attributes, and DOS programs don't know EAs exist. If you boot DOS and run the CHKDSK utility, all of the extended attributes will be lost since DOS will think they are contained within lost clusters. If you run a DOS program that modifies a file containing extended attributes, the extended attributres (and the object data) for that file will probably be lost since the program doesn't know how to preserve them. This is true whether the program is running under DOS itself or under OS/2 Warp.

Developer's Desktop

For example, I use Microsoft Word For Windows, under OS/2 (of course), as my word processor. I have special properties assigned to the directories that I keep my word processing data files in. Since Word For Windows only reads the directories, these properties are safe. I have custom icons for some of my folders and I have long names (see the discussion of WPFileSystem Properties) assigned as the titles for the directories. However, I can't assign any special properties to the data files themselves, since Word For Windows would destroy the extended attributes, and the values I had assigned to the properties, when it updated the files.

With all of these potential problems, why would anyone use the FAT File System? There are several good reasons to continue to use the FAT File System.

- ✗ The FAT File System is the only file system supported for diskettes.
- ✗ The FAT File System is fairly efficient when used with a small fixed disk (128MB or smaller).

- ✗ DOS and Windows programs only understand file names that consist of a 1- to 8-character file name followed by an optional 1- to 3-character file extension. This limitation was imposed by the FAT File System, and has never been changed.
- ✗ The only file system supported by DOS is the FAT File System. If you are going to boot DOS as well as OS/2 Warp on your system, the boot partition and any logical drives that you want DOS to be able to see must be FAT partitions.

The HPFS file system. The High Performance File System (HPFS) was created to answer many of the deficiencies in the FAT File System.

HPFS only supports fixed disks, not diskettes. The allocation unit size in HPFS is 512 bytes, regardless of the size of the disk. The extended attributes are stored along with each file, not in one large separate file. The file name structure was greatly relaxed – each part of a file name can now be up to 255 characters long, with the total path name not exceeding 259 characters.

I know — it sounds too good to be true. And it is — there are problems using HPFS also:

- ✗ DOS cannot reference an HPFS partition.
- ✗ DOS programs running under OS/2 Warp can reference files on an HPFS partition; however, they still cannot reference or even maintain the extended attributes for the file.
- ✗ Since DOS programs only understand short file names, if a file or directory has a file name that exceeds the DOS limitation, the file or directory becomes invisible to the DOS program.

The CD-ROM file system. The CD-ROM File System is only used to access files on a CD-ROM. This is a read-only file system (since you can't write to a CD-ROM), which doesn't support extended attributes.

Other file systems. Network products, such as IBM Lan Server 4.0, supply other file systems to access files that reside on remote servers. The characteristics of these file systems vary, depending on the limitations of the file system in use on the file server.

WPFileSystem — properties

WPFileSystem objects inherit most of their properties from the WPObject class. However, there are some properties that are unique to this class.

- **File System object** — There is always a File System object, either a directory or a file, associated with every object of the WPFileSystem class and all derivative classes. You can find the name of the File System object on the first File page of the Settings Notebook for the File System object. The name of the File System object may be different from the Title property of the corresponding WPFileSystem object. For example, if the File System object is maintained by the FAT File System, the name of the file or directory is limited to a 1- to 8-character file name, a period, and a 1- to 3-character file extension. However, the object's Title property may be any character string up to 259 characters long.

  If the number of characters in the object's title exceeds the capacity of the file system, the title is called a *Long File Name* and is stored in the extended attributes for the File System object. The Workplace Shell automatically constructs a legal file name from the object's title. For example, for a File System object that is stored on the FAT File System, the Workplace Shell uses the first eight characters of the object's title to construct the file name and the first three characters following the last period (if any) in the object's title to construct the file extension. A directory whose title is Business Documents might have a real name of BUSINESS, while a file whose title is Ira's Data might have a real name of IRA'S_DA.

- **Creation Date and Time** — This is the date and time that the File System object (the directory or the file) was created. This information is not maintained by the FAT File System, but it is maintained by other file systems, such as HPFS.
- **Last Changed Date and Time** — This is the date and time that the File System object was changed. This information is maintained by all file systems.
- **Last Accessed Date and Time** — This is the date and time that the File System object was accessed last. This information is not maintained by the FAT File System, but is maintained by other File Systems, such as HPFS.
- **File Size** — This is the number of bytes of data stored in the file or the directory.
- **Extended Attribute Size** — This is the number of bytes of extended attributes stored for the File System object.
- **Flags** — These are the file system attributes for the file or directory. These attributes include *hidden*, which indicates whether or not the file or directory is

visible; *system,* which indicates whether or not the file or directory is owned by the system; *read-only,* which indicates whether or not you can update or delete the file or directory; and *archive,* which indicates whether or not the file was changed since the last time it was processed by a backup program.

- **Subject, Comments, Key Phrases, and History** — These are properties that you can assign to the File System object. These values are kept in the extended attributes for the file or directory, and can be interrogated by an OS/2 program.
- **Data type** — This defines the type of data that is contained within the directory or data file. This Data type may be used to associate a program with a data file. See Chapter 2 for a discussion about using associations.

WPFileSystem — behaviors

There are no new behaviors defined by the WPFileSystem object class.

WPFileSystem — pop-up menu items

There are no new pop-up menu items defined by the WPFileSystem object class.

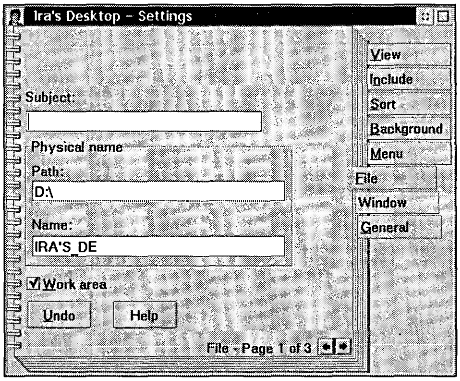
WPFileSystem — Settings Notebook pages

These pages are added to the Settings Notebook for all file system objects.

WPFileSystem — Menu page. The Menu page allows you to specify action items to add to the object's pop-up menu and open cascade menu. It also allows you to specify new cascade menus (and their contents) to add to the pop-up menu. See Chapter 2 for a full description of how to use the Menu page.

WPFileSystem — file pages. The first File page (see Figure 5-9) displays the subject, real file name, and work area property of the file system object.

Figure 5-9. WPFileSystem - File Page 1



Ira's Desktop - Settings

Subject:

Physical name

Path: D:\

Name: IRA'S_DE

☑ Work area

File - Page 1 of 3

You can enter any information you want in the Subject entry field. This information is saved in the .SUBJECT extended attribute and can be accessed by an application program.

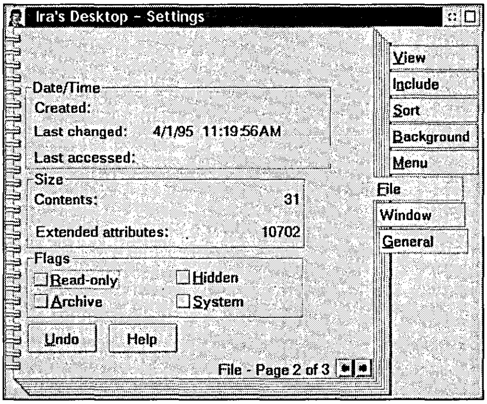
The Path entry field contains the name of the directory that contains the File System object. You cannot change this property directly. The path will be changed automatically if you move the object to a different folder.

The Name entry field contains the real name of the File System object. You cannot change this property directly. You can change the real name of the File System object by changing its title.

The Work area check box is displayed only for folders. See the description of the Work area property for the WPFolder class later in this chapter.

The second File page (see Figure 5-10) displays the File System attributes for the object. The only information on this page that you can directly change are the File System flags.

Figure 5-10. WPFileSystem - File Page 2



Ira's Desktop - Settings

Date/Time
Created:
Last changed: 4/1/95 11:19:56AM
Last accessed:

Size
Contents: 31
Extended attributes: 10702

Flags
□ Read-only □ Hidden
□ Archive □ System

File - Page 2 of 3

The Created field shows the date and time that the file or directory was created. This information is not maintained for the FAT File System.

The Last changed field shows the date and time that the file or directory was changed last. This information is maintained for all file systems.

The Last accessed field shows the date and time that the file or directory was accessed last. This information is not maintained for the FAT File System.

The Size/Contents field shows the number of bytes contained in a file or the number of File System objects contained within a directory.

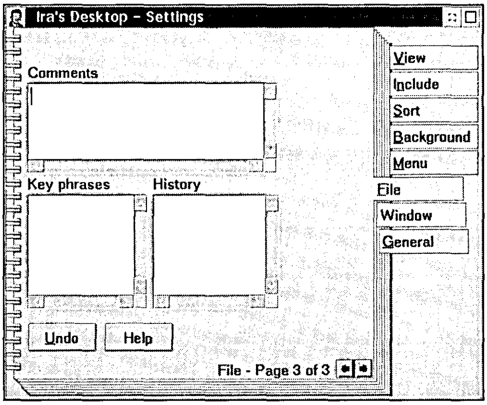
The Size/Extended attributes field shows the number of bytes of extended attributes which are assigned to the file or directory.

The Flags box shows the current values of File System attributes. You may change these values by turning the respective check boxes on or off.

- ✗ The Read-only check box indicates whether the file or directory can be changed.
  - If the check box is off, the file or directory can be changed.
  - If the check box is on, the file or directory can only be read, but it can't be written to and it can't be deleted.
- ✗ The Archive check box indicates whether or not the file or directory has changed since the last time it was backed up either with the BACKUP command, the XCOPY command with the /M option specified, or with a backup program that manipulates the archive attribute.
  - If the check box is off, the file or directory hasn't changed.
  - If the check box is on, the file or directory has changed since the last time it was backed up.
- ✗ The Hidden check box indicates whether the file or directory is hidden.
  - If the check box is off, the file or directory is visible.
  - If the check box is on, the file or directory is not displayed in an open view of the folder containing the file or directory and is not listed using the DIR command.
- ✗ The System check box indicates whether the file or directory belongs to the operating system.
  - If the check box is off, the file or directory belongs to the user.
  - If the check box is on, the file or directory belongs to the operating system, and is hidden.

The third File page (see Figure 5-11) shows optional information that you can save with the file.

Figure 5-11.
WPFileSystem -
File Page 3



You can enter any information you want in the Comments entry field. This information is stored in the .COMMENTS extended attribute and is available to any application program using this file.

You can enter any information you want in the Key phrases list box. This information is stored in the .KEYPHRASES extended attribute and is available to any application program using this file.

The History list box contains information stored about the history of the object by an application program or by the operating system. You cannot change this information. This information is stored in the .HISTORY extended attribute.

WPTransient — the base class for temporary objects

The main characteristic of the WPTransient class and all of its derivatives is that these objects are temporary and do not have any permanent representation. These objects are generally created for the life of some task and are deleted when the task is complete. For example, when you open a printer object you expect to see objects representing the jobs that are in the print queue. These objects only exist while the printer object is open – they are automatically deleted when the print job completes or when you close the printer object. The data that are represented by the object persist independently from the object. Another example is when you minimize an object to the Minimized Window Viewer. The Workplace Shell creates an object that resides in the Minimized Window View folder to represent each minimized object so you can manipulate the objects. These Minimized Window objects are automatically deleted when the original object is closed.

WPTransient — properties

There are no new properties defined by the WPTransient object class.

WPTransient — behaviors

There are no new behaviors defined by the WPTransient object class.

WPTransient — pop-up menu items

There are no new pop-up menu items defined by the WPTransient object class.

WPTransient — Settings Notebook pages

There are no new Settings Notebook pages defined by the WPTransient object class.

I told you this base class is rather boring. It just sits there, temporarily, and does nothing for itself. However, you must keep in mind that the base classes are just building blocks on which other classes are built.

Note

The WPTransient may not be too exciting on its own. But think about an object-oriented database program. You might have data stored in an indexed database file. When you want to see this data, the program might want to construct a Tree view container showing the relationships between the data elements.

Uncensored

One way to do that is to create a folder that it can open in Tree view and populate the folder with transient objects representing the elements of the database. It would create a subclass of WPTransient with properties and behaviors that make the database usable. For example, each object could support a view that shows the detailed data-base record. These objects could also support being printed, singly or in combination, being deleted, being copied to another similar database, and so on. They could have special pop-up menu items allowing the user to perform other actions on the database records. When the user closes the database view, these objects would all be deleted, even though the underlying database remains intact.

Could these database objects be implemented using the other base classes? The WPFileSystem class requires a separate file for each object. This is not feasible for a classical indexed database. It is also not very efficient, since each allocated file requires extra disk storage in the form of a directory entry and the rounding effect of file space allocation.

OK, so the WPFileSystem class doesn't sound usable. How about the WPAbstract class? It is possible to create abstract objects representing each record in the database. But, for a large database this would create a very large overhead in the OS2.INI file. It also makes the database much less portable, since part of the database now resides in the OS2.INI file on the system on which the database was built. What if you want the database to reside on a file server and be able to access it from multiple clients? How do you do that if the Workplace Shell objects representing the database records don't exist on the client systems? It is difficult to envision how to construct such a database.

With the transient object concept, this isn't a problem. When you open a view of the database, the database application simply builds the transient objects that it needs on your system. When you close the database, the objects go away. The worst that can happen is that you leave the database open and shut down your system. When the system is restarted, the objects are gone, since they had no persistent form. The next time you open the database, the objects are recreated.

The point is, you can't overlook the power of a seemingly boring, nonfunctional object class.

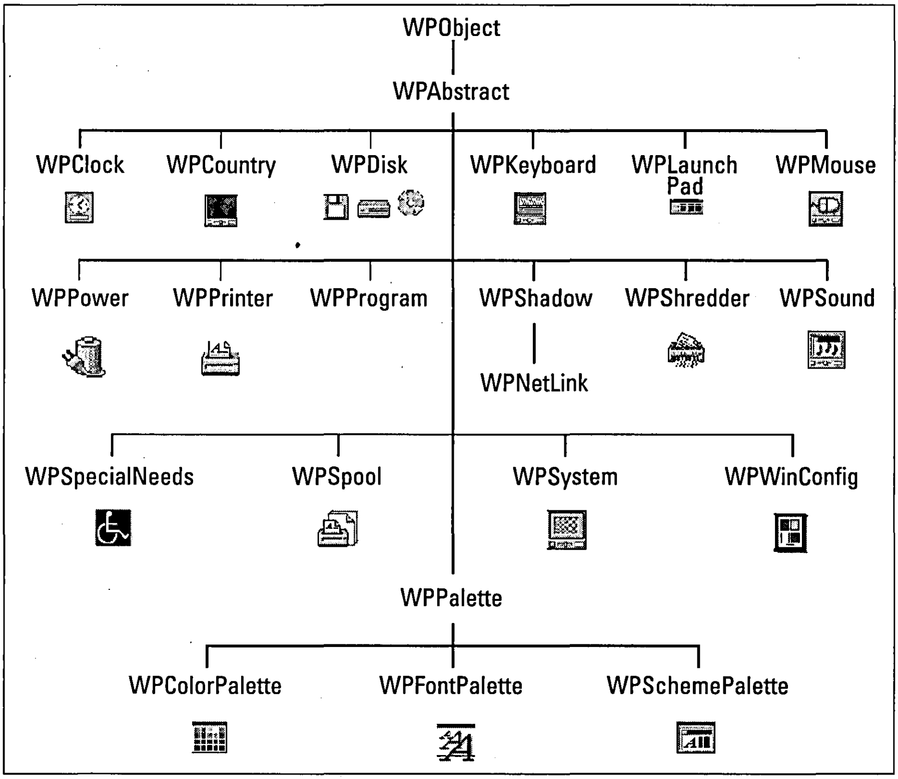
Abstract Objects

Figure 5-12 shows all Workplace Shell classes that are derived from the WPAbstract class. There are oodles of them!

Most of these objects are described in detail in other chapters of this book. Here is a list of all of the abstract objects and where to find the information about them:

- ✗ **WPClock** — The System Clock object (see Chapter 7).
- ✗ **WPColorPalette** — The Color Palette (see Chapter 6).
- ✗ **WPCountry** — The Country object (see Chapter 7).
- ✗ **WPDisk** — The Disk object.
- ✗ **WPKeyboard** — The Keyboard object (see Chapter 7).
- ✗ **WPLaunchPad** — The LaunchPad object (see Chapter 4).
- ✗ **WPMouse** — The Mouse object (see Chapter 7).

Figure 5-12. Workplace Shell Object Class Hierarchy - Abstracts



- ✘ **WPNetLink** — The Network Shadow object (see Chapter 9).
- ✘ **WPPalette** — The Palette object.
- ✘ **WPPower** — The Power object (see Chapter 7).
- ✘ **WPPrinter** — The Local Printer object (see Chapter 14).
- ✘ **WPProgram** — The Program Reference object (see Chapter 2).
- ✘ **WPRPrinter** — The Remote Printer object (see Chapter 14).
- ✘ **WPSchemePalette** — The Scheme Palette (see Chapter 6).
- ✘ **WPShadow** — The Shadow object.
- ✘ **WPShredder** — The Shredder object.
- ✘ **WPSound** — The Sound object (see Chapter 7).
- ✘ **WPSpecialNeeds** — The Special Needs object.

- ✗ WPSpool — The Spooler object (see Chapter 7).
- ✗ WPSystem — The System object (see Chapter 7).
- ✗ WPWinConfig — The WIN-OS/2 Setup object (see Chapter 7).

WPDisk — the Disk object

Actually, this is much more than a "disk" object. This object represents a file system device, such as a hard disk drive, diskette drive, CD-ROM, magnetic tape drive, network attached drive, PCMCIA device, or a RAM disk. The Disk objects reside in the Drives folder. They are automatically created when the Drives folder is opened, so you always have a picture of the current set of file system devices currently attached to your system. For example, if you dynamically access network drives, these drives will show up in the Drives folder only when they are attached.

WPDisk — properties

The property defined by the WPDisk class is

- ✗ **Drive Number** — This is the logical drive number (DRIVENUM= setup string) for the file system device. This number is in the range of 1 to 26, and corresponds to the drive letters A: through Z:.

WPDisk — behaviors

The behaviors defined by the WPDisk class depend on the type of file system device that is represented by the object. These behaviors include

- ✗ **Check Disk** — Verifies the integrity of the data on the diskette or fixed disk.
- ✗ **Copy Disk** — Copies the contents of a diskette to another diskette.
- ✗ **Eject Disk** — Ejects the removable medium from the drive. This behavior is available for certain types of diskette drives and CD-ROM drives.
- ✗ **Format Disk** — Erases all information on the diskette or fixed disk and prepares the diskette or fixed disk to accept new information.
- ✗ **Lock Disk** — Locks the removable medium in the drive so the user can't remove it. This behavior is available for certain types of diskette drives and CD-ROM drives.
- ✗ **Unlock Disk** — Unlocks the removable medium so the user can remove it. This behavior is available for certain types of diskette drives and CD-ROM drives.

WPDisk — pop-up menu items

The pop-up menu items defined by the WPDisk object class vary depending on the type of file system device represented by the WPDisk object. These pop-up menu items include

- **Check disk...** — You can verify the integrity of the data on the diskette or fixed disk by selecting the **Check disk...** pop-up menu item. This displays the Check Disk dialog box, as shown in Figure 5-13. You might choose this action if you suspect there has been damage done to the data on a diskette or fixed disk, or if you are getting repeated error reports when trying to access data or programs on the diskette or fixed disk. It is particularly important to check all fixed disks after a power failure or some other system failure that forces you to restart your system without invoking the Shut Down facility or restarting your system using the Ctrl-Alt-Del sequence.
- **Copy disk...** — You can copy the contents of a diskette to another diskette by selecting the **Copy disk...** pop-up menu item on the Disk object for the source diskette. This displays the Drive Copy Window, as shown in Figure 5-15. The Workplace Shell automatically chooses a target diskette drive for you. If you only have one diskette drive installed in your system that matches the characteristics of the source diskette, the same drive is used for the source and target. If you have two diskette drives installed in your system that match the characteristics of the diskette being copied, the second drive is used for the target. For example, if you have one 3¹/₂-inch diskette drive and one 5¹/₄-inch diskette drive in your system, the source and target diskette will be the same drive. If you have two 3¹/₂-inch diskette drives in your system, the copy operation uses both drives.
- **Eject disk** — You can eject the medium from the diskette or CD-ROM drive by selecting the **Eject disk** pop-up menu item on the Disk object for the diskette or CD-ROM. When you select the **Eject disk** pop-up menu item, the diskette or CD-ROM is automatically ejected from the drive regardless of whether or not you previously locked the drive (see **Lock disk** pop-up menu item). This option only appears in the pop-up menu if the diskette drive or CD-ROM drive supports ejecting the medium through program control.
- **Format disk** — You can erase all information on the diskette or fixed disk and prepare the diskette or fixed disk to accept new information by selecting the **Format disk...** pop-up menu item. This displays the Format Disk dialog box, as

shown in Figure 5-16. You normally select this option if you want to remove all information from a diskette so you can reuse it, if you purchase a new "unformatted" diskette and want to prepare it for use, or if you install a new fixed disk or change the partition sizes on an existing fixed disk and want to prepare the fixed disk for use by OS/2. Be very careful about using this option on a fixed disk. If you accidentally specify to format a fixed disk, all information contained on that fixed disk will be erased and cannot be retrieved.

- **Lock disk** — You can lock the medium in the drive so it can't be removed by pressing the eject button on the drive by selecting the **Lock disk** pop-up menu item. This option only appears in the pop-up menu if the diskette drive or CD-ROM drive supports locking the medium through program control. You might choose this option if you are going to leave a diskette or CD-ROM in your machine while you leave your system unattended. You would normally lock the diskette or CD-ROM drive and lock the keyboard before leaving your machine. This prevents anyone from removing the diskette or CD-ROM while you are gone. When you return, you would normally unlock the keyboard. You can leave the diskette or CD-ROM drive locked or you can unlock it at that time.

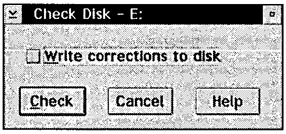
Warning

If you leave a diskette locked in the A: drive, you might have a problem restarting your system, since most systems boot from the A: drive if there is a diskette installed in the drive. If the diskette is locked in the A: drive, you might not be able to restart OS/2 from your fixed disk.

- **Unlock disk** — You can unlock the medium in the drive so it can be removed by pressing the eject button on the drive by selecting the **Unlock disk** pop-up menu item. This option only appears in the pop-up menu if the diskette drive or CD-ROM drive supports locking the medium through program control. See the discussion about the **Lock disk** pop-up menu item for an explanation of when you might use the **Unlock disk** pop-up menu item.

Check Disk dialog box. The Check Disk dialog box (see Figure 5-13) is used to specify the options for the CHKDSK command (see Chapter 15 for a description of the CHKDSK command), which is invoked as a result of selecting the **Check disk...** pop-up menu item.

Figure 5-13. Check Disk Dialog Box



Check Disk - E:

□ Write corrections to disk

The Write corrections to disk check box allows you to specify whether or not to correct any file system problems that are found. If you turn on the check in this check box, the CHKDSK command will automatically fix any problems found and write those corrections to the diskette or fixed disk. If you leave this check box unchecked, the CHKDSK command will report any problems found but will not correct them.

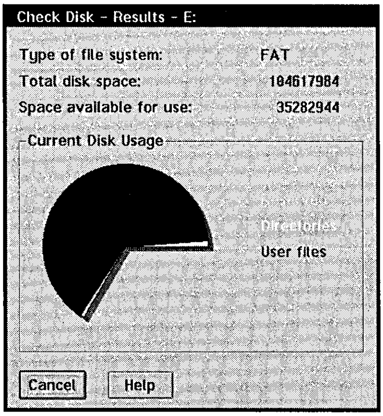
Tip

If you suspect there may be a problem with a diskette or fixed disk, you can invoke CHKDSK to report just the problems to you. If CHKDSK reports cross-linked sectors or file allocation errors, you might want to copy all of the data from the specified partition to another partition before letting CHKDSK fix the problem. You might be able to save some of your data that you would otherwise lose when CHKDSK fixes the problem. (Of course, saving your data might not work if the file system control structures are badly corrupted, but you have nothing to lose in trying.) Once you run CHKDSK again and let it fix the problems it found, you can then compare your saved data to the files that CHKDSK fixed and keep the better files.

After you set the check box the way you want it, you can click on the **Check** push button to start the CHKDSK program. This will display the Check Disk Results Window (see Figure 5-14).

You can click on the **Cancel** push button to terminate the operation without invoking the CHKDSK program or after you have finished looking at the results.

Figure 5-14. Check Disk Results Window



Check Disk - Results - E:

Type of file system: FAT
Total disk space: 104617984
Space available for use: 35282944

The Check Disk Results Window shows you the statistics about the drive. The Type of file system item tells you whether the file system controlling this diskette or fixed disk is the FAT File System or the HPFS File System, or some other file system you might have installed. This information is presented as soon as the CHKDSK operation starts.

The Total disk space line tells you how many bytes of data the diskette or fixed disk can hold. This information is not presented until the CHKDSK operation is completed.

The Space available for use line tells you how many bytes of data are currently not being used on the diskette or fixed disk. This information is not presented until the CHKDSK operation is completed.

Warning

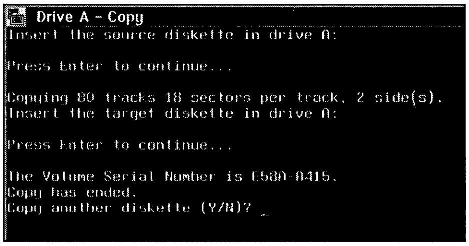
If you are using a disk compression program, like Stacker for OS/2, the number of available bytes may not be accurate. Available bytes are calculated by the disk compression program based on the average compression ratio for that fixed disk. However, the files that you create may not be able to be compressed (for example, files that are already compressed cannot be compressed again) and they might take more space than expected.

The Current Disk Usage area displays a pie chart representing the percentage of the total capacity of the diskette or fixed disk used by directories, user files (files that do not have the System attribute), unusable areas (areas on the diskette or fixed disk that were marked as "bad sectors" when the diskette or fixed disk was formatted), extended attributes, and reserved (files that have the System attribute). This information is not presented until the CHKDSK operation is completed.

You can click on the **Cancel** push button at any time to terminate the CHKDSK operation before it completes or to return to the Check Disk dialog box after you are finished looking at the results.

Drive Copy Window. The Drive Copy Window (see Figure 5-15) is displayed while the system is copying a diskette from one drive to another. This Window contains messages from the DISKCOPY command (see Chapter 15 for a description of the DISKCOPY command) invoked by the Workplace Shell when you select the **Copy disk...** pop-up menu item.

Figure 5-15. Drive Copy Window



```
Insert the source diskette in drive A:

Press Enter to continue...

Copying 80 tracks 18 sectors per track, 2 side(s).
Insert the target diskette in drive A:

Press Enter to continue...

The Volume Serial Number is E58A-A415.
Copy has ended.
Copy another diskette (Y/N)? _
```

The first message prompts you to insert the diskette to be copied into the source drive. This is the drive from which you selected the **Copy disk...** pop-up menu item. It is normally A: or B:. You must insert the diskette to be copied into the specified diskette drive, then press **Enter.**

It is a good idea to write protect any diskette you are copying from, to make sure you don't accidentally use this diskette as the target instead of the source. For a 3¹/₂-inch diskette, you can just slide the write protect slider down before inserting the diskette in the drive. For a 5¹/₄-inch diskette, you have to cover the notch on the diskette with a write protect tab. If you decide not to copy a diskette at this time, press **Ctrl-C** or **Ctrl-Break** to terminate the copy operation.

The next message informs you of the characteristics of the diskette being copied. It tells you the number of tracks, the number of sectors per track, and the number of sides of the diskette that are being copied. For the most part, this is pretty useless information and you can ignore it. The only thing that is important is that the source and target diskettes must be of the same type. For example, if you are copying a 3¹/₂-inch high density diskette (which is normally indicated by the letters *HD* in the upper right corner of the diskette), the target must also be a high density diskette. Similarly, for a 5 ¹/₄-inch diskette, you can only copy a high capacity diskette (normally indicated by the letters *HC* on the label) to another high capacity diskette.

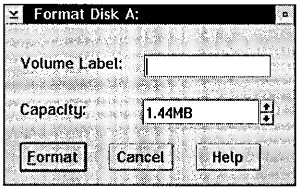
When the COPY command is done reading the information from the source diskette, it prompts you to insert the target diskette and press **Enter.** This diskette must be of the same type as the source diskette and must not be write protected. If the source and target are the same physical drive (for example, A: and B: are the same drive on a single drive system), you must remove the source diskette before inserting the target diskette. (This is why it is a good idea to write protect your source diskette — if you forget to switch diskettes and simply press **Enter,** you might destroy your source diskette.) When you have inserted the target diskette into the specified drive, press **Enter** to continue the operation. If you decide not to copy the diskette after all, you can press **Ctrl-C** or **Ctrl-Break** to terminate the copy operation.

When the copy operation is complete, the COPY command will tell you what Volume Serial Number it has assigned to the new diskette. This information is probably useless to you also.

The COPY command then asks if you want to copy another diskette. If you respond Y, the process will start over, so you can copy multiple diskettes with one invocation of the COPY command. Again, make sure you don't mix up your source and target diskettes. If you respond N, the Drive Copy Window will close.

Format Disk dialog box. The Format Disk dialog box (see Figure 5-16) is used to specify the options used when formatting a diskette or fixed disk.

Figure 5-16.
Format Disk Dialog Box



Volume Label:

Capacity: 1.44MB

Specify the name you want to use to refer to the diskette or fixed disk in the Volume Label entry field. This name can be up to 11 characters long. The characters you can use in the volume label may be restricted by the file system that is used to format the diskette or fixed disk. All diskettes are formatted using the FAT File System. A fixed disk may be formatted using either the FAT File System or the HPFS File System. The name you specify for the volume label can be displayed using the VOL command (see Chapter 15), the DIR command, or the Details page of the Disk object.

✗ **Formatting diskettes** — For a diskette you must specify the number of bytes of data that the diskette can hold in the Capacity spin button. For a $3^{1}/_{2}$-inch diskette, this may be 720KB, 1.44KB, or 2.88KB. You must specify the correct size for the type of diskette you insert in the drive. Most modern $3^{1}/_{2}$-inch diskette drives can detect the type of diskette that is inserted in the drive and will only allow you to format the diskette for its proper capacity.

Note

However, some diskette drives (for example, some of the diskette drives in IBM personal computers such as the PS/1 or PS/2 brands) allow you to format diskettes at the "wrong" capacity. They will allow you to format a high density diskette for a capacity of 720KB. If you do this, you will find that the diskette might not be able to be read in another machine whose diskette drive senses the type of diskette and automatically prepares itself to read the diskette at the "correct" capacity.

Uncensored

Even though you might be able to override the normal capacity, always format $3^{1}/_{2}$-inch double density diskettes (no special indication on the diskette) for 720KB capacity; $3^{1}/_{2}$-inch high density diskettes (indicated by the letters HD on top-right corner of the diskette) for 1.44MB capacity; and $3^{1}/_{2}$-inch DS-ED diskettes (indicated by the letters ED on top-right corner of the diskette) for 2.88MB capacity. For $5^{1}/_{4}$-inch diskettes, the restriction of capacity is inherent in the diskettes themselves — double-sided diskettes can't be formatted as if they are high capacity diskettes and high capacity diskettes are unreliable when formatted as double-sided diskettes, so always format $5^{1}/_{4}$-inch double-sided diskettes (no special indication on the diskette) for 360KB capacity and $5^{1}/_{4}$-inch high capacity diskettes (indicated by *HC* on the diskette label) for 1.2MB capacity.

✗ **Formatting a fixed disk** — For a fixed disk, you must choose the file system that is to control the data on the fixed disk. You can select either the **HPFS** radio push button for the High Performance File System or the **FAT** radio push button for the FAT File System. The choice is not so simple. If you intend to boot DOS sometimes and refer to files on this fixed disk partition, you must use the FAT File System.

Tip

If the partition is only going to be referred to from OS/2, you have to consider two things: the type of programs you are running and the size of the partition. If you are going to run DOS or Windows programs and you want them to be able to reference data on this partition, you might want to use the FAT File System. DOS and Windows programs can read and write data on an HPFS partition (under OS/2, of course) as long as the names of the files and their directories conform to the FAT File System naming convention. If you assign a long name to a file or to one of the directories in the path to the file on an HPFS partition, a DOS or Windows program won't be able to see the file, and therefore can't reference it.

If you assign a long name to a directory on a FAT partition, the directory really has a short name (only its Title property is a long name) so a DOS or Windows program has no problem referencing any file in that directory. If you are not constrained to using the FAT File System, you then have to consider the size of the partition. If this is a relatively small partition (your guess is as good as mine as to what *small* means – maybe 256MB or less) the system performance may be better if you choose the FAT File System. If this is a large partition (again, I am not really sure what *large* means – certainly over 512MB is large) the HPFS File System will perform better.

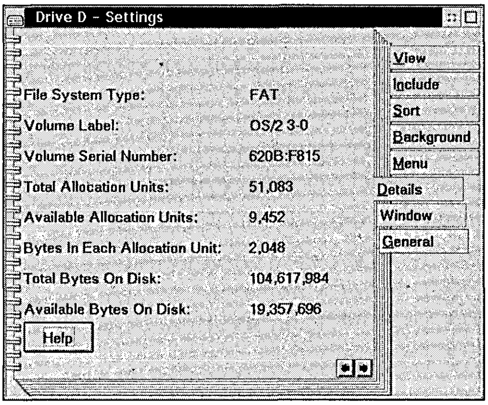
On my personal system at home, I have 3 FAT partitions and 1 HPFS partition (C: is a bootable DOS partition; D: and E: are bootable OS/2 partitions; and F: is a program/data partition that is only used under OS/2). The bootable DOS partition clearly must be FAT. The program/data partition is slightly over 1200MB long (fixed disk prices dropped so much I just couldn't resist buying a 2GB drive!) so it is HPFS. The OS/2 partitions are both small (100MB each). I made them FAT partitions so that in an emergency I could still boot DOS and perform simple maintenance on them (like editing the CONFIG.SYS file or replacing a bad executable module). This may just be paranoia on my part, since with OS/2 Warp I can now boot directly to a command prompt and perform the simple maintenance functions.

After you have set up the parameters, click on the **Format** push button to start formatting the diskette or fixed disk. Remember that once you have started the format operation, all of the data on the diskette or fixed disk are irretrievably lost – you can't change your mind. If you decide that you don't really want to lose this data, click on the **Cancel** push button instead.

WPDisk — Settings Notebook pages

The Settings Notebook for a disk object contains the following page.

Details page. The Details page (see Figure 5-17) shows information about the drive.



File System Type: FAT
Volume Label: OS/2 3-0
Volume Serial Number: 620B:F815
Total Allocation Units: 51,083
Available Allocation Units: 9,452
Bytes In Each Allocation Unit: 2,048
Total Bytes On Disk: 104,617,984
Available Bytes On Disk: 19,357,696

Figure 5-17.
WPDisk Object – Details Page

- ✘ The File System Type field shows what file system controls this volume. It may be FAT for a diskette or fixed disk, HPFS for a fixed disk, CDFS for a CD-ROM, or some other file system for another type of device.
- ✘ The Volume Label field displays the label assigned to the volume. This is the name you specified when you formatted the diskette or fixed disk or the label assigned to the volume (like a CD-ROM) when it was manufactured. If no volume label was assigned, the field specifies "No Label."
- ✘ The Volume Serial Number field displays the serial number that was assigned to the volume when it was formatted. This is a unique two-part hexadecimal number. You normally don't have to worry about this value – it is used only to insure that you don't change the medium in a removable media device (like a diskette or CD-ROM drive) while the system is using files on that device.
- ✘ The Total Allocation Units field displays the size of the volume in terms of "allocation units." See the description of the Bytes In Each Allocation Unit field.
- ✘ The Available Allocation Units field displays the amount of storage available for use on the volume in terms of allocation units. For a device that you can't write on (like a CD-ROM), the number of Available Allocation Units may be zero even though the volume isn't full.
- ✘ The Bytes In Each Allocation Unit field displays the size of each allocation unit, in bytes. The allocation unit is assigned when the volume is formatted. For the FAT File System, the allocation unit is proportional to the size of the volume. For

example, a small volume, such as a diskette or a small fixed disk, might have an allocation unit size of 2,048 bytes. As the volume size increases, the size of the allocation unit increases. This is because the FAT File System only supports 65,536 allocation units on any one volume, no matter how large the volume is.

For example, if you have a volume that is 512MB in size, the allocation unit must be 8,192 bytes. If the volume is between 512MB and 1,024MB, the allocation unit must be 16,384 bytes. The significance of the allocation unit size is that it is the *smallest* unit of space that can be allocated on the volume. For example, if the allocation unit is 8,192 bytes, every file on that volume takes a minimum of 8,192 bytes of space — even a file that contains only 1 byte of data! On the other hand, the allocation unit size for HPFS is 512 bytes, no matter how large the volume is. This makes for a much more efficient use of large volumes.

- ✘ The Total Bytes On Disk field shows the total size of the volume in bytes. This number is the same as the Total Allocation Units field multiplied by the Bytes In Each Allocation Unit field. Note that this is the actual number of bytes. If you are used to thinking about the size of your fixed disk or diskettes in terms of MB (megabytes), you have to divide this number by 10,485,76 (because 1 megabyte is 1,024 x 1,024 bytes).
- ✘ The Bytes Available On Disk field shows the number of free bytes on the volume. Again, this is the Available Allocation Units multiplied by Bytes In Each Allocation Unit. When you create a new file on the volume, the Bytes Available On Disk is decreased by an even multiple of the allocation unit size — not by the number of bytes of data in the file.

WPPalette — the Palette object class

You will never see an object whose class is WPPalette. WPPalette is an object class whose sole purpose is to perform the common functions needed by the WPColorPalette, WPFontPalette, and WPSchemePalette objects.

A palette consists of a rectangular arrangement of cells. Each cell contains the representation of one of the members of the palette. For the palettes that are part of OS/2 the cells contain colors (for the Color Palette), the names of fonts (for the Font Palette), or icons representing schemes (for the Scheme Palette). The WPPalette class provides the structure for the palette, while the child classes provides the mechanisms to implement their own type of palettes.

WPPalette — properties

The properties defined by the WPPalette class include

- ✗ **Number Of Columns Of Cells** — This is number of columns of cells in the palette (XCELLCOUNT= setup string).
- ✗ **Number Of Rows Of Cells** — This is the number of rows of cells in the palette (YCELLCOUNT= setup string).
- ✗ **Width Of Each Cell** — This is the width of each cell, in pixels (XCELLWIDTH= setup string).
- ✗ **Height Of Each Cell** — This is the height of each cell, in pixels (YCELLHEIGHT= setup string).
- ✗ **Horizontal Separation Of Each Cell** — This is the horizontal gap between columns of cells (XCELLGAP= setup string).
- ✗ **Vertical Separation Of Each Cell** — This is the vertical gap between rows of cells (YCELLGAP= setup string).

WPPalette — behaviors

The behaviors defined by the WPPalette class include

- ✗ **Palette View** — Displays the window containing the cells in the palette. The WPPalette class positions each cell in the window, resizes the window if you drag the sizing borders, and scrolls the cells within the window. The child classes are responsible for drawing the items within the cells themselves.
- ✗ **Display The Proper Mouse Pointer** — The WPPalette class determines when to display a special mouse pointer when you drag the contents of a cell. The child classes are responsible for drawing the mouse pointer itself.
- ✗ **Control Drag And Drop Operations** — The WPPalette class tracks the mouse during a drag operation and informs the child class when you drop the cell on a window. The child class is responsible for performing the actual drop operation.

WPPalette — pop-up menu items

There are no new pop-up menu items defined by the WPPalette object class. However, the Open cascade menu contains an entry for "Palette."

WPPalette — Settings Notebook pages

There are no new Settings Notebook pages defined by the WPPalette object class.

WPShadow — the Shadow object

The Shadow object is a representation of another Workplace Shell object. A Shadow object lets you see an object in a folder of your choice, without moving the original object. For example, if you have a directory on the F: drive and would like to see a folder representing that directory on the Desktop, you could create a Shadow object of the directory on the Desktop. The Shadow object resides in the Desktop folder, but the directory it represents still resides on the F: drive.

WPShadow — properties

A Shadow object has very few properties of its own – it mostly reflects the properties of the original object. For example, the title and icon for a Shadow object are the same as the title and icon of the original object. The properties defined by the WPShadow class include

- ✘ **Original object** — This is the object that is linked to the Shadow object (SHADOWID= setup string).
- ✘ **Style** — This is a set of characteristics for the Shadow object that indicate the behaviors that are allowed for the Shadow object. These characteristics are independent of the style of the original object. These characteristics include:
  - Whether or not the Shadow object can be moved to another folder (NOMOVE= setup string)
  - Whether or not the Shadow object can be copied to another folder (NOCOPY= setup string)
  - Whether or not the Shadow object be deleted or dragged to the Shredder (NODELETE= setup string)
  - Whether or not you can create a shadow of the Shadow object (NOLINK= or NOSHADOW= setup strings).

Most of these affect what items appear in the pop-up menu for the shadow object. For example, the original object may be not deleteable (that is, its style contains the NODELETE=YES flag) but the Shadow object may be deleteable.

Uncensored

The implementation of the Style property is not correct, and may cause you some problems. When you are looking at the properties of a shadow, you correctly see the NOMOVE, NOCOPY, NODELETE, and NOLINK values from the shadow and the other style flags from the linked object. However, if you try to set the properties of a shadow, all of the properties are set in the shadow object. For example, if you try to set the visiblity style of a shadow, the style flag is set into the Shadow object. However, if you try to look at the visibility style flag, you see the style flag from the linked object. This means that you can't mark a Shadow object as invisible because it never reports its own visibility state. The same is true for all of the other style flags.

WPShadow — behaviors

The Shadow object doesn't define any new behaviors. It mainly passes all requests on to the original object, and lets the original object define the behaviors of the Shadow object. However, the Shadow object actually processes all Delete, Move, Copy, Link, and Pickup requests by itself. This allows you to manipulate the Shadow object independently from the linked object.

When a view of a Shadow object is opened, both the Shadow object and the original object are marked as open in their respective folders. When a view of an object is opened, all shadows of that object are also marked open in their folders.

When a characteristic of an object is changed (for example, its title or icon), that change is automatically reflected in all shadows of that object.

WPShadow — pop-up menu items

The WPShadow class adds the following item to the pop-up menu for the object.

✗ **Original** — The Original pop-up menu item points to a list of actions that can be performed on the original object. These actions include

- **Locate** — Opens the folder containing the original object
- **Copy...** — Opens the folder containing the original object, selects the original object, and initiates a Copy operation for the original object
- **Delete...** — Deletes the original object

WPShadow — Settings Notebook pages

There are no new Settings Notebook pages defined by the WPShadow object class.

WPShredder — the Shredder object

The Shredder object is an object that acts like a paper shredder – it immediately destroys anything put in it. In the case of a paper shredder, you carry a stack of papers to the shredder, put them in the shredder, and get a bunch of useless strips of paper out of the bottom. (OK, they aren't actually useless – you can recycle them.) The Shredder object works similarly – you drag one or more objects to the Shredder object, drop them, and they vanish in a puff of smoke. (Use your imagination for this one!)

WPShredder — properties

There are no new properties defined by the WPShredder object class.

WPShredder — behaviors

The Shredder object has one behavior – any object that is dragged to it is destroyed, if possible. The effect is exactly the same as if you select the **Delete...** item from the object's pop-up menu. The only difference is that you can drag multiple objects to the Shredder and delete them all together, and you can have a sound effect played when the object is deleted through the Shredder object.

WPShredder — pop-up menu items

There are no new pop-up menu items defined by the WPShredder object class.

WPShredder — Settings Notebook pages

There are no new Settings Notebook pages defined by the WPShredder object class.

WPSpecial Needs — the Special Needs object

The Special Needs object is automatically created on your Desktop when you activate the special needs facilities in the Keyboard object (see Chapter 7). This object is just a reminder that the special needs facilities have been activated.

WPSpecialNeeds — properties

There are no new properties defined by the WPSpecialNeeds object class.

WPSpecialNeeds — behaviors

There are no new behaviors defined by the WPSpecialNeeds object class. There are no views defined for the WPSpecialNeeds object class, so this object can't be opened.

WPSpecialNeeds — pop-up menu items

There are no new pop-up menu items defined by the WPSpecialNeeds object class.

WPSpecialNeeds — Settings Notebook Pages

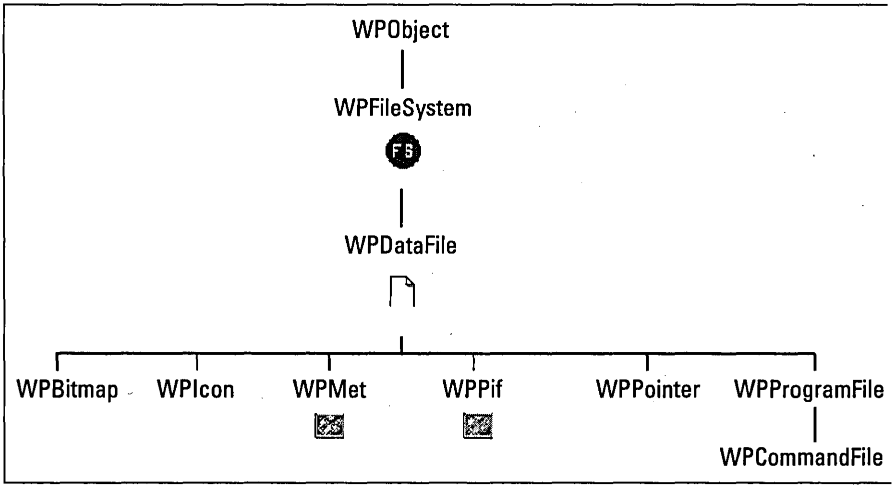
There are no new Settings Notebook pages defined by the WPSpecialNeeds object class.

Data File Objects

You might notice in Figure 5-1 that there are two groups of object classes defined under WPFileSystem. This is because there are two types of file system objects — data files and directories. They both share some common characteristics, which are defined in the WPFileSystem object class, and they have some differences. The WPDataFile object class defines the common characteristics of all data file objects, while the WPFolder object class defines the characteristics of all directory objects.

Figure 5-18 shows all data file Workplace Shell classes that are derived from the WPFileSystem class.

Figure 5-18. Workplace Shell Object Class Hierarchy - Data Files



WPDataFile — the Data File object

The Data File object represents a data file in the file system. The Data File object always resides in the Folder object that corresponds to the directory that contains the data file. You can use the Shadow object to see a representation of a Data File object in a folder other than where the data file exists. For example, you could create a Shadow of a data file on the Desktop. You could then manipulate the Shadow object just as if it were the original data file, including opening it, dragging it to another object, printing it, and so on. The only thing you can't do directly with the Shadow object is delete the original object by dragging the Shadow object to the Shredder — that just destroys the Shadow object, not the original object..

WPDataFile — properties

There are no new properties defined by the WPDataFile class.

WPDataFile — behaviors

There are no new behaviors defined by the WPDataFile object class.

WPDataFile — pop-up menu items

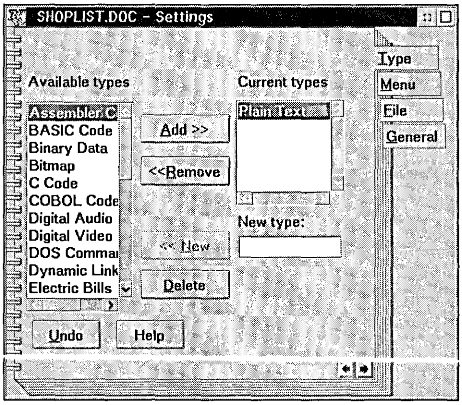
There are no pop-up menu items defined by the WPDataFile object class.

WPDataFile — Settings Notebook pages

The Settings Notebook for a data file object contains the following pages.

Type page. The Type page (see Figure 5-19) displays the data types currently assigned to the data file and allows you to change the list of assigned data types. This information is stored in the .TYPE extended attribute and is available to any application program.

Figure 5-19. WPDataFile - Type Page



The Current types list box contains the list of data types currently assigned to this data file. Normally, you would assign a single data type to a data file, since it can only contain one type of data at a time. However, since different programs that can process this data may be associated to different data type names, you might include multiple names to allow these programs to access the data properly.

You can add a new type to the data types list by selecting a data type from the Available types list box, then clicking on the **Add>>** push button. This moves the selected data type from the Available types list to the Current types list. If the data type you want to assign isn't listed in the Available types list box, you can create the new data type by typing its name in the New type entry field, then click on the **<<New** push button. After you add the new type, you can then select it and add it to the data type list.

You can remove a type from the data types list by selecting the data type from the Current types list box, then clicking on the **<<Remove** push button. This moves the selected data type from the Current types list back to the Available types list.

If you want to delete a data type from the Available types list, select the data type and click on the **Delete...** push button.

Other data file classes

There are a bunch of classes built on top of the WPDataFile class. These object classes define data files with slightly different characteristics. Each class is distinguished by the data type or by the name of the data file. These classes have special meaning to the Workplace Shell and to certain programs that are a part of OS/2.

The child classes of WPDataFile include

- **WPBitmap** — This data file contains a bitmap. A data file is a member of the WPBitmap object class if its data type is Bitmap or if its file name ends in .BMP. A WPBitmap object can be used as the background bitmap for a folder or for the system lockup background bitmap.
- **WPIcon** — This data file contains an icon. A data file is a member of the WPIcon object class if its data type is Icon or if its file name ends in .ICO. A WPIcon object can be used as the icon for any other object. An icon object can be edited using the Icon Editor.

- **WPMet** — This data file contains a Presentation Manager meta file. This is the description of how to draw something in a window. A data file is a member of the WPMet object class if its data type is Metafile or if its file name ends in .MET. A WPMet object can be displayed using the Picture Viewer.
- **WPPif** — This data file contains a Picture Interchange Format File. A data file is a member of the WPPif object class if its data type is PIF file or if its file name ends in .PIF.
- **WPPointer** — This data file contains a pointer. A data file is a member of the WPPointer object class if its data type is Pointer or if its file name ends in .PTR. A WPPointer object can be assigned to one of the mouse pointers in the Mouse object (see Chapter 7).
- **WPProgramFile** — This data file contains an executable program. A data file is a member of the WPProgramFile object class if its data type is Executable or if its file name ends in .COM or .EXE. A WPProgramFile can be invoked by double-clicking mouse button one on the object's icon. See Chapter 2 for a compelete description of the WPProgramFile object.
- **WPCommandFile** — This data file contains an OS/2 or DOS command file. A data file is a member of the WPCommandFile object class if its data type is DOS Command File or OS/2 Command File or if its file name ends in .BAT or .CMD. A WPCommandFile can be invoked by double-clicking mouse button 1 on the object's icon.
- **WPPrinterDriver** — This data file contains an OS/2 Printer Driver. A data file is a member of the WPPrinterDriver object class if its file name ends in .DRV.

Warning

Any file whose name ends in .DRV will be taken to be a Printer Driver File. The WPPrinterDriver class has a class style that makes all files of its class nondeleteable. Therefore, you will not be able to delete any file ending in .DRV from the Workplace Shell. The only way to delete a .DRV file from the Workplace Shell is to delete the folder containing the data file.

Tip

You can go to an OS/2 Command Prompt and delete the .DRV file from the command line. This will get rid of the data file and the corresponding WPPrinterDriver object.

Folder Objects

Data File objects are only half the File System picture. Folder objects are the other half—Workplace Shell objects that represent directories in the file system.

Figure 5-20 shows all folder Workplace Shell classes that are derived from the WPFileSystem class.

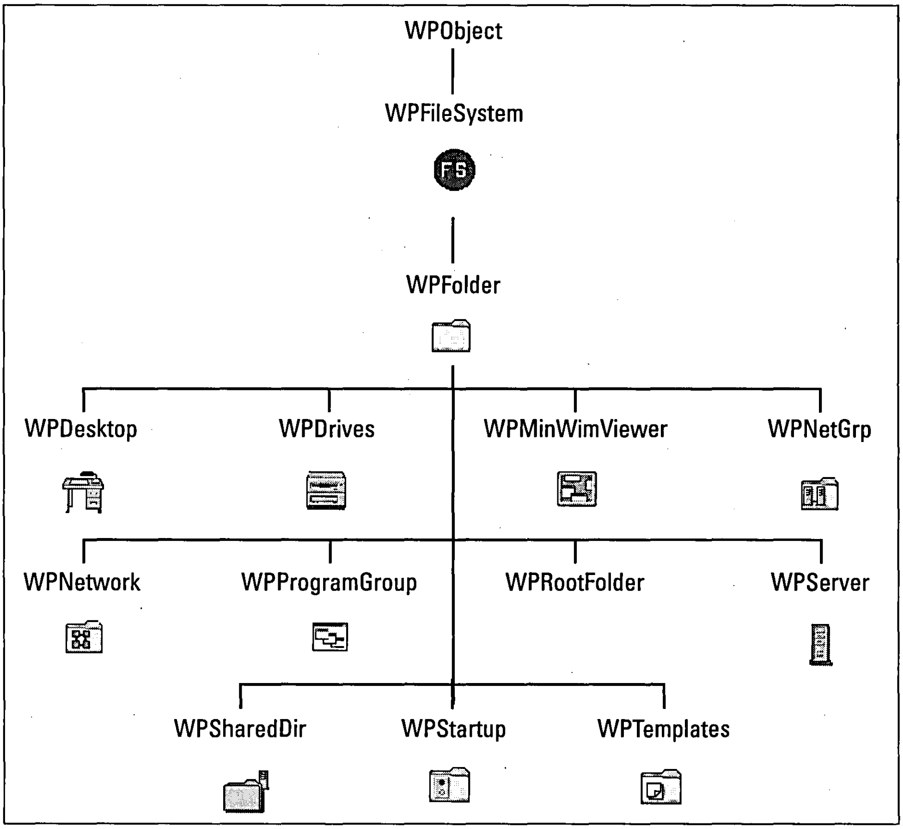


Figure 5-20. Workplace Shell Object Class Hierarchy - Folders

WPFolder — the Folder object

The Folder object represents a directory in the File System. You can use the Shadow object to see a representation of a Folder object anywhere you like. For example, you could create a Shadow of a directory on the Desktop. You could then manipulate the Shadow object just as if it were the original directory, including opening it, dragging it to another object, and so on. The only thing you can't do directly with the Shadow object is delete the original object by dragging the Shadow object to the Shredder — that just destroys the Shadow object, not the original object.

WPFolder — properties

The properties defined by the WPFolder class include:

- ✘ **Default View** — This defines the default view for the folder. This is the view you get if you just double-click on the folder's icon. You can set the default view using the DEFAULTVIEW= setup string or the Menu page of the Settings Notebook for the folder.

- ✘ **Icon View Characteristics** — This defines the characteristics of an open Icon View of the folder. These characteristics can be set using the ICONVIEW=, ICONFONT=, and ICONVIEWPOS= setup strings or on the first View page of the Settings Notebook for the folder.

- ✘ **Tree View Characteristics** — This defines the characteristics of an open Tree view of the folder. These characteristics can be set using the TREEVIEW= and the ICONFONT= setup strings or on the second View page of the Settings Notebook for the folder.

- ✘ **Details View Characteristics** — This defines the characteristics of an open Details View of the folder. These characteristics can be set using the DETAILSVIEW=, DETAILSFONT=, DETAILSCLASS=, and DETAILSTODISPLAY= setup strings or on the third View page of the Settings Notebook for the folder.

- ✘ **View Sorting Characteristics** — These characteristics indicate how the objects seen within an open view of the folder are to be sorted. You can set these characteristics using the SORTCLASS=, SORTBYATTR=, DEFAULTSORT=, and ALWAYSSORT= setup strings or on the Sort page of the folder's Settings Notebook.

- ✘ **Icon Positions** — These are the positions of all icons that you see when you open the Icon View of the folder. This information is saved so you see the icons in the same position every time you open the folder.

- ✘ **Background** — This is the background color or bitmap that you see when you open any view of the folder. You can set the background using the BACKGROUND= setup string or the Background page of the folder's Settings Notebook.

- ✘ **Work Area** — This property indicates whether or not this is a Work Area folder (see Chapter 2 for a description of the characteristics of a Work Area folder). This property can be set using the WORKAREA= setup string or on the File page of the folder's Settings Notebook.

- **Animation Icon** — This is the icon that you see when a view of the folder is open. This icon is optional – if it isn't defined, you see the same icon whether the folder is open or closed. You can change this icon using the ICONNFILE= or ICONNRESOURCE= setup strings or on the second General page of the folder's Settings Notebook.
- **Self Close Option** — This property defines the behavior of the folder when you open an object from within the folder.

WPFolder — behaviors

The behaviors that are defined by the WPFolder class include

- **Icon View, Tree View,** and **Details View** — These are the three views defined for all folders. See Chapter 2 for a description of these Views Of A Folder.
- **Work Area** — A folder may take on the characteristics of a Work Area as described in Chapter 2.
- **Self Close** — A folder may automatically close itself when you open an object inside the folder.

WPFolder – pop-up menu items

The WPFolder class adds the following items to the pop-up menu for the object.

- **Open** (cascade menu) — The Open cascade menu for a folder contains three items: Icon view, Tree view, and Details view. You can open the corresponding folder view by clicking on one of these cascade menu items. If you just click on the **Open** pop-up menu item, you will open the default view for the folder.
- **Open parent** — You can open the parent folder (that is, the folder containing this Folder object) by selecting the **Open parent** pop-up menu item. If you hold the Shift key while clicking on the **Open parent** pop-up menu item, the parent folder is opened and the Folder object is closed.
- **Refresh now** — You can request the system to refresh the open view of the folder by clicking on the **Refresh now** pop-up menu item.

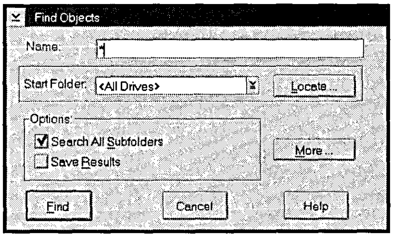
Uncensored

When the Workplace Shell refreshes the folder, it updates all information about the objects inside the folder. This includes adding or removing objects from the folder view if data files were copied to or removed from the directory since the last refresh, updating the object's titles if data files were renamed, and updating the object's icons if they were changed. For example, if a folder contains icon data files and you edit the icon data using the Icon Editor, you won't see the changed icon in the open view of the folder unless you refresh the folder.

- ✗ **Find...** — You can search for objects by selecting the **Find...** pop-up menu item. This displays the Find Objects dialog box, as shown in Figure 5-21.
- ✗ **Select** — You can select or deselect all objects in the folder by clicking on the **Select** pop-up menu item. This displays a list of two items: **Select all** and **Deselect all**. Click on the **Select all** item to select all objects contained within the folder or click on the **Deselect all** item to deselect all objects contained within the folder. The selections are normally used in data transfer operations. See Chapter 3 for information about data transfer.
- ✗ **Sort** — You can sort the objects contained in the folder by clicking on the **Sort** pop-up menu item. This sorts the objects by the default sort criterion (in ascending order). If you want to sort the objects based on a particular characteristic, click on the arrow to the right of the **Sort** pop-up menu item. This displays a list of properties that you can sort on. Click on the property you want and the objects in the folder will be sorted based on that property. See the section on the Sort Page for information on how to specify what properties appear in the Sort cascade menu.
- ✗ **Arrange** — You can arrange the objects in an Icon view of a folder in neat rows by selecting the **Arrange** pop-up menu item. See Chapter 2 for more information about Icon view and what the Arrange action does.

Find Objects dialog box

The Find Objects dialog box (see Figure 5-21) is used to find objects on your system. This dialog box is displayed when you ask the system to search for something. If the Find Objects dialog box is displayed because you selected the **Find...** pop-up menu item, it allows you to specify options that limit the search. If the dialog box is displayed because you clicked on a **Find...** push button on one of the Workplace Shell Settings Notebook pages, the search options are normally established by the Settings Notebook page and you might not be able to change them.



Name: *

Start Folder: <All Drives>

Options:

☑ Search All Subfolders

☐ Save Results

Figure 5-21.
Find Objects
Dialog Box

Specify the title of the object to be found in the Name entry field. You can use the * and ? wildcard characters within the name, where ? specifies any single character and * specifies any character or characters. For example, you could specify a name of *.DOC, which specifies to find all objects whose title end in the characters .DOC. The default name is *, which means find all objects which meet the other search criteria, regardless of the object's title.

Specify where to start searching in the Start Folder combination box. You can choose one of the entries in the list box, which contains the name of the desktop folder, <All Drives> (which specifies to search all drives for the specified objects), and the names of each drive, or you can specify the fully qualified name of the starting folder (including the drive letter). You can click on the **Locate...** push button to get a set of tabbed pages, like the Create Another dialog box (see Figure 5-2), from which you can select the starting folder.

You can specify that only the starting folder is to be searched by turning off the check in the Search All Subfolders check box. Turn on this check if you want to search the whole directory tree starting at the specified folder.

The results of the find operation are normally put into a temporary container. If you want the results saved in a folder on the Desktop, check the Save Results check box. If this option is selected, the folder will contain shadows of each of the objects that satisfies the search criteria.

You can specify various criteria to limit the objects that satisfy the search by clicking on the **More...** push button. This presents you with the Find Criteria dialog box, as shown in Figure 5-22. Once you have all the search criteria set up the way you want them, click on the **OK** push button to return to the Find Objects dialog box.

When you have defined all of the parameters for the search, click on the **Find...** push button to start searching the specified folders for all objects that meet the search criteria you set up.

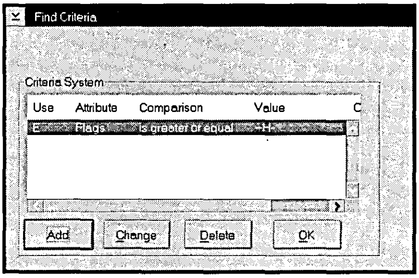
If you request to save the find results, the Workplace Shell creates a new Find Results folder on the Desktop that contains shadows of all objects meeting the search criteria and opens the folder. You can then manipulate these objects like any other shadows – you can open them by double-clicking on them, opening their Settings Notebooks, dragging them to other folders, deleting them (that is, deleting the shadows – not the original objects), and so on.

If you didn't request to save the find results, the Workplace Shell opens a window that contains all objects meeting the search criteria. You can manipulate these objects exactly the same as if you had opened a folder containing the objects. These are the actual objects, not shadows, so if you delete one of them, you are deleting the actual object.

Warning

Let me reiterate one important point. If you request to save the find results, the window that opens is a Find Results Folder that contains shadows of the found objects. If you delete one of these objects, you are only deleting the shadow. If you do not request to save the find results, the window that opens is a Container Window that contains the *real* objects. If you delete one of these objects, you are deleting the actual object.

Figure 5-22.
Find Criteria
Dialog Box



Use	Attribute	Comparison	Value	C
E	Flags	is greater or equal	-H-	

The Find Criteria dialog box lets you specify one or more criteria that an object has to meet in order to be included in the Find results.

See the section on Search Criteria later in this chapter for a description of the criteria that you can use to narrow the search. If you want to add a new search criterion, click on the **Add** push button. This displays the Add Criteria dialog box, as shown in Figure 5-23, which allows you to specify a new search criterion.

You can select one of the search criterion on the Find Criteria dialog box and click on the **Change** push button. This displays the Change Criteria dialog box, as shown in Figure 5-24, to change the search criterion.

You can delete one of the search criterion by selecting it on the Find Criteria dialog box and clicking on the **Delete** push button.

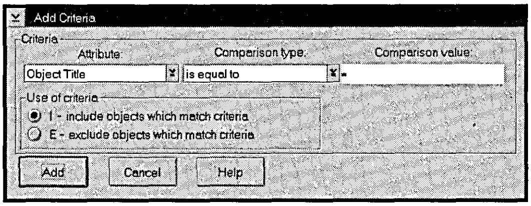
Once you have all the search criteria set up the way you want them, click on the **OK** push button to return to the Find Objects dialog box.

Search criteria

Both the Find Objects dialog box and the Include page of the Folder Settings Notebook allow you to specify one or more criteria to narrow the scope of the operation. In the case of the Find Objects dialog box, the criteria are used to limit the number of objects that are found. For the Include page, the criteria are mainly used to hide some objects so they don't appear in an open view of the folder. In both cases, the search criteria are used to build a set of objects.

These criteria can be specified on the Add Criteria (see Figure 5-23) and Change Criteria (see Figure 5-24) dialog boxes. (I'm sorry — it's not my fault! I know that "criterion" is singular and "criteria" is plural, but the person who named the dialog boxes obviously didn't think about that. You can only add or change one criterion at a time using these dialog boxes.)

Figure 5-23. Add Criteria Dialog Box



Attribute: Object Title

Comparison type: is equal to

Comparison value: *

● I - include objects which match criteria

○ E - exclude objects which match criteria

The first thing you can specify (even though it is last on the dialog box) is whether the new criterion is to allow an object that meets the criterion to be included in the final set of objects or if it is to be excluded. Select the **I** radio push button if an object is to be added to the final set of objects if it meets the criterion or select the **E** radio push button if an object is to be removed from the final set of objects if it meets the criterion.

You now have to specify the criterion used to include or exclude an object. In order to specify the criterion you have to specify three items: the property of the object to be tested, how it is to be tested, and what value of the property will satisfy this criterion. You specify the property to be tested from the Attribute list box, how to test this property from the Comparison type list box, and what value to test for under Comparison value.

Here are the properties you can specify and what you can do with them:

- **Object Title** — This is the title of the object. There is only one way to test the Object Title and that is to test whether or not it is the same as the comparison value. You type the title you want to test for in the Comparison value entry field. You may use wildcard characters in the comparison value, where "?" specifies any single character and "*" specifies none or any number of characters. For example, if you don't want to see the System Setup object in the OS/2 System Folder, you could specify an Exclude criterion of "Object title" "is equal to" "System Setup" on the Include page for the OS/2 System Folder. As another example, to find all documents whose titles start with the word "Chapt" you might specify an Include criterion of "Object title" "is equal to" "Chapt*.doc" for the Find Objects dialog box.

- **Object Style** — This is the style of the object. Remember, the style consists of a set of individual flags, each one indicating a different attribute of the object. The Comparison type contains three choices to allow to you isolate the attributes you want to test:

  - **Is equal to** — The object's style must contain only the attributes you select and no others

  - **Contains** — The object's style must contain all of the attributes that you select

  - **Does not contain** — The object's style must not contain any of the attributes that you select.

  You then must select at least one attribute from the Comparison value list box. This is a multiple selection list box, so you can select one or more attributes. For example, if you want to find all objects that are moveable and copyable, you could specify an Include criterion of "Object Style" "does not contain" "Not Moveable and Not Copyable" for the Find Objects dialog box.

- **Object Class** — This is the class of the object. You have two choices here for the Comparison type:

  - **Is equal to** — To test for a particular class of object, select the class of the object from the Comparison value list box. For example, if you want to hide the LaunchPad icon from the Desktop, define an Exclude criterion that specifies "Object Class" "is equal to" "LaunchPad" on the Include page for the Desktop Settings Notebook.

- **Is descended from** — To test for any object that is in some part of the object class hierarchy (the object class hierarchy is useful for something!), select the class that is at the top of the hierarchy from the Comparison value list box. For example, if you want to find all Abstract objects, define an Include criterion that specifies "Object Class" "is descended from" "Abstract" on the Find Objects dialog box.

✘ **Real name** —This is the real file name for a file system object. You can specify a variety of Comparison types: **is equal to, is greater than, is less than, is greater or equal, is less or equal, is not equal.** You must type the value you want to compare the file's name to in the Comparison value entry field. You can use wildcards here also. If you want to find all files whose real names are of the form "FIGnn-nn.PCX" (where n indicates a digit), you can specify an Include criterion of "Real name" "is equal to" "FIG??-??.PCX" on the Find Objects dialog box.

✘ **Size** and **Extended Attribute Size** — These are the size of a file system object or the size of its extended attributes. You can specify a Comparison types of **is equal to, is greater than, is less than, is greater or equal, is less or equal, is not equal.** You must then specify the number of bytes to compare the file size or extended attribute size to in the Comparison type spin button. For example, to find all files that are larger than 1,024 bytes you would specify an Include criterion of "Size" "is greater than" "1024" on the Find Objects dialog box. To find all files that are between 1,024 and 65,535 bytes long, you must specify two Include criteria: "Size" "is greater or equal to" "1024" and "Size" "is less or equal" "65535." The only objects that will meet both criteria are files whose sizes are between 1,024 and 65,535 bytes.

✘ **Last write date** — This is the last date that the file system object was updated. The Comparison type may be **is on, is after, is before, is on or after, is on or before,** or **is not.** You specify the date you want to compare the file's last write date to in the month, day, and year spin buttons under the Comparison value heading. For example, if you want to find all files that were written after July 4, 1994, you can specify an Include criterion of "Last write date" "is after" "7 4 1994."

✘ **Last write time** — This is the last time that the file system object was updated. The Comparison type may be **is on, is after, is before, is on or after, is on or before,** or **is not.** You specify the time you want to compare the file's last write time to in the hour, minute, and AM/PM spin buttons under the Comparison value heading. For example, if you want to find all files that were written after 6 p.m. you can specify an Include criterion of "Last write time" "is after" "6 0 PM." You might want to find all files that were written after 2 p.m. on September 8, 1994, but there is no way to specify this. If you specify two Include criteria of "Last write date" "is on or after" "9 8 1994" and "Last write time" "is after" "2 0 PM", you will find all files that were last updated after 2 p.m. on any date from September 8, 1994 to the present. In other words, the "Last write time" "is after" "2 0 PM" applies to all days, not just September 8, 1994.

- **Last access date** and **Last access time** — These properties are the last date or time the file system object was read. Since this information is only maintained by the HPFS File System, these properties are meaningless for files stored on the FAT File System. You would specify these properties the same way as the Last write date and time.

- **Creation date** and **Creation time** — These properties are the date and time the file system object was created. Since this information is only maintained by the HPFS File System, these properties are meaningless for files stored on the FAT File System. You would specify these properties the same way as the Last write date and time.

- **Flags** — This is the combination of the file system, Read Only, Archive, Hidden, and System attributes. You can specify a Comparison type of **is equal to, is greater than, is less than, is greater or equal, is less or equal,** or **is not equal.** You can select the values of the four file system attributes to compare to in the spin buttons under the Comparison value heading. If the spin button contains a "-", the corresponding attribute is not specified. If the spin button contains a letter, the attribute is specified (*R* stands for the Read Only attribute, *A* is for Archive, *H* is for Hidden, and *S* is for the System attribute).

Note

In order to understand how the comparison works, think of the four attributes as a 4-bit value, where 0 indicates the attribute is not specified and 1 indicates the attribute is specified. Then take the four attributes from the Comparison value spin buttons as a 4-bit value also. Compare the two values, ignoring all bit positions where the Comparison value spin button contains a "-". The result of this comparison indicates whether or not the include the file in the final set. Are you sufficiently confused? So am I!

Note

Let's try a simple example. If you specify "Flags" "is less than" "—H-", this indicates that all files that are not hidden are to be included in the final set of objects. If this is too confusing for you (like it is for me), use the individual flags for the search criterion (see next bulleted item).

- **Read Only Flag, Hidden Flag, System Flag, Directory Flag,** and **Archived Flag** — These properties are the individual file system attributes. Each attribute can be tested to be **ON** or **OFF**. For example, to hide any object that has the hidden attribute, specify an Exclude criterion of "Hidden Flag" "is equal to" "ON" for the Include page of the folder's Settings Notebook.

- **Subject, Comment, Key Phrases,** and **History** — These properties are the Subject that you can specify on the first File page of the Settings Notebook and the Comments, Key Phrases, and History that you can specify on the third File page. The Comparison types you can specify are **is equal to, contains,** and **does not contain.** You must type the text to compare the Subject to in the Comparison value edit field. You can use the "?" and "*" wildcard characters. For example, to find all files that contain the word "invoice" in their subject, define an Include criterion that specifies "Subject" "contains" "invoice."

After you have specified the criterion, click on the **Add** push button to add the criterion. Remember, an object has to meet all defined criteria in order to be put into the final set of objects. If you decide you don't want to add a new criterion, just click on the **Cancel** push button.

The Change Criteria dialog box (see Figure 5-24) is used to change an existing search criterion. It has the exact same options as the Add Criteria dialog box. You can change the criterion to anything you want – it does not have to be even remotely similar to its original value. For example, you can change a criterion that currently reads Exclude "Object Class" "is equal to" "LaunchPad" to Include "Subject" "contains" "invoice."

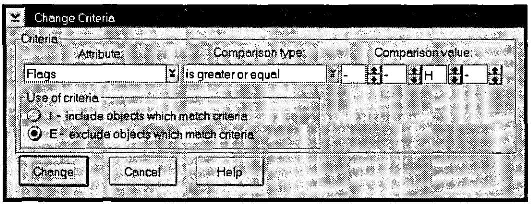


Figure 5-24. Change Criteria Dialog Box

Change Criteria

Criteria

Attribute: Flags

Comparison type: is greater or equal

Comparison value: - - H -

Use of criteria

○ I - include objects which match criteria

● E - exclude objects which match criteria

After you have changed the criterion the way you want, click on the **Change** push button to change the criterion. Remember, an object has to meet all defined criteria in order to be put into the final set of objects.

If you decide you don't want to change the criterion, just click on the **Cancel** push button.

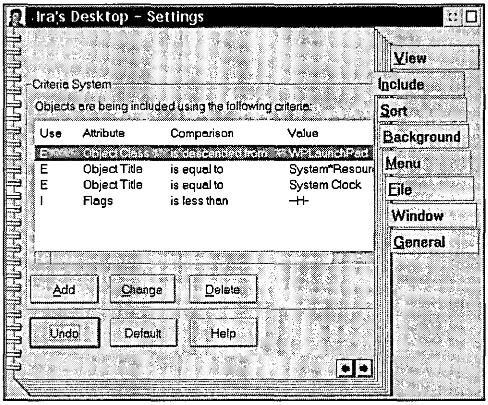
WPFolder — Settings Notebook pages

The Settings Notebook for all folders contain the following pages.

View pages. The three View pages allow you to change the characteristics of the Icon view, Tree view, and Details view of the folder. These pages are described in detail in Chapter 6.

Include page. The Include page (see Figure 5-25) allows you to specify what objects are to be shown in an open view of the folder. You can specify criteria that include certain types of objects, as well as criteria that exclude certain objects. Using these options you can completely control which objects you see in an open view of the folder.

Figure 5-25. WPFolder – Include Page



Ira's Desktop - Settings

Criteria System

Objects are being included using the following criteria:

Use	Attribute	Comparison	Value
E	Object Class	is descended from	WPLaunchPad
E	Object Title	is equal to	System*Resour
E	Object Title	is equal to	System Clock
I	Flags	is less than	—H—

The Criteria System group shows the criteria that are currently being used to select the objects that are shown in an open view of the folder. See the section on Search Criteria in this chapter for a description of the criteria that you can specify here.

If you want to add a new criterion to further limit the objects that are displayed, click on the **Add** push button. This displays the Add Criteria dialog box, as shown in Figure 5-23, which allows you to add a new search criterion.

You can change one of the search criteria by selecting the line containing the criterion that you want to change and clicking on the **Change** push button. This displays the Change Criteria dialog box, as shown in Figure 5-24.

You can remove one of the search criteria by selecting the line containing the criterion that you want to remove and clicking on the **Delete** push button.

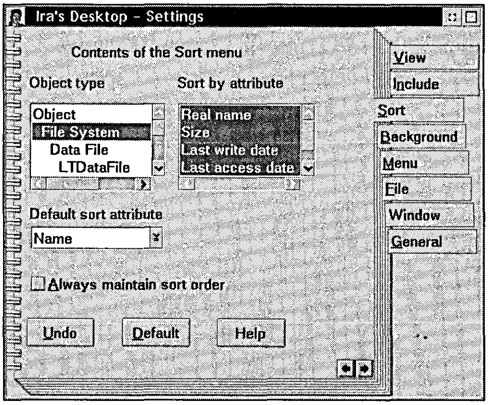
These changes take place immediately, so if you have a view of the folder open when you are changing the search criteria you can see the results as you make the changes.

Developer's Desktop

Here is an example of how I use the Include page. I have a Work Area Folder that I use as my main folder. I keep a LaunchPad object open in my work area, as well as a program that monitors system performance and disk free space. Whenever I open my Work Area Folder, these objects automatically open. But, I don't want to see the icons for these objects – they just clutter my folder. I added the two following criteria to the Include page for my Work Area Folder to make these icons invisible: Exclude "Object class" "is equal to" "LaunchPad" and Exclude "Object title" "is equal to" "System*Resources." The second criterion has an "*" in the middle of the name since the object has a two-line title, and the "*" matches the new line character in the title. On the rare occasion when one of these objects is accidentally closed, I just select the **Find...** option from the Work Area Folder's pop-up menu to find all objects in the folder. I then double-click on the object in the Find Results Window to reopen the object.

Sort page. The Sort page (see Figure 5-26) allows you to specify the items that appear on the cascade menu when you select the **Sort** pop-up menu item.

Figure 5-26. WPFolder - Sort Page



Ira's Desktop - Settings

Contents of the Sort menu

Object type: Object, File System, Data File, LTDataFile

Sort by attribute: Real name, Size, Last write date, Last access date

Default sort attribute: Name

□ Always maintain sort order

You can select the items that appear in the Sort cascade menu from the list of properties for one object class. The rationale behind this logic is that most folders contain objects of a similar type. For example, a folder might represent a directory that contains a selection of data files. In this case, most (if not all) of the objects contained within the folder are part of the File System object class, and you would select the properties to sort by from the File System class. On the other hand, if you have a folder that represents an object-oriented data base, most of the objects contained within the folder might be data base entries. In this case you would want to sort the folder contents by some property of the data base entries.

Uncensored

You can select the predominant object class from the Object type list box. If you see several entries in this list box with the same name, the problem is caused by an improperly written object class that was installed as part of some product. For example, if you see two entries in a row that read "Folder," the second entry represents a child class of WPFolder that neglected to specify a unique class name. If you know what product is causing this problem, you should contact the supplier of the product and inform it about the problem.

Once you have selected an object class, all of the properties of that class that you can sort by are displayed in the Sort by attribute list box. Initially, all properties of the specified object class are selected. You can select or deselect a property by clicking on it. All of the properties that are selected will appear in the Sort cascade menu.

Note

You can choose which property is the default sort property by selecting it in the Default sort attribute combination box. You might notice that this combination box has two entries that don't show up in the Sort by attribute list box (Name and Type). These two properties are always available, so they are not displayed in the Sort by attribute list box. If you change the default sort property while the Always maintain sort order check box is checked, the folder is automatically re-sorted.

Finally, there is one check box near the bottom of the Sort page. If you check the Always maintain sort order check box, the contents of the folder are automatically sorted by the default sort property every time the folder is opened. Also, when you add a new item to the folder (by copying or moving an object into the folder or by creating a new object in the folder) the contents of the folder are automatically re-sorted. When you check this check box, the folder is automatically sorted by the default sort attribute. If you have an Icon view or Details view of the folder open while you are changing the options, you will see the results immediately.

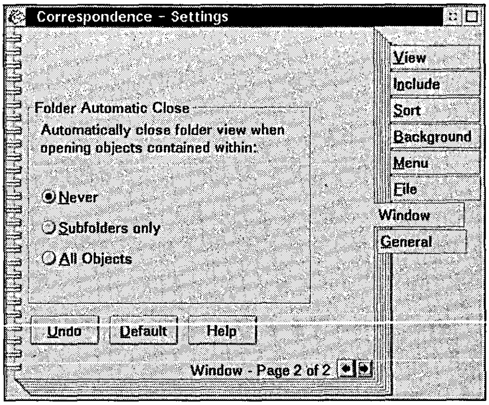
Note

Sorting the contents of a folder (either automatically or manually) only applies to Icon view and Details view, not Tree view.

Background page. The Background page allows you to specify what kind of background you want displayed in an open view of the folder. The Background page is described in Chapter 6.

Window page 2. The second Window page (see Figure 5-27) allows you to specify the action the folder is to take after opening an object within the folder.

Figure 5-27. WPFolder - Window Page 2



Correspondence - Settings

Folder Automatic Close

Automatically close folder view when opening objects contained within:

● Never

○ Subfolders only

○ All Objects

Window - Page 2 of 2

The Folder Automatic Close option allows you to specify the action the system takes when it opens an object from within the folder. You can select one of the following radio push buttons:

- ✘ Never — The system leaves the folder open when you open an object within the folder
- ✘ **Subfolders only** — The system automatically closes the folder when you double-click on a folder within it
- ✘ **All objects** — The system automatically closes the folder when you double-click on any object within it

If you don't select any of the radio push buttons on this page, the system default value, which appears on the second Window page of the Settings Notebook for the System object, applies to this folder.

The Folder Automatic Close option doesn't apply to any folder that is marked as a Work Area Folder, including the Desktop. If the Folder Automatic Close option doesn't specify that the folder is to be closed automatically, you can dynamically override this option by holding the Shift key when you double-click on an object within a folder. In this case, the system will automatically close the folder (unless it is a Work Area Folder), regardless of the type of object and the setting of the Folder Automatic Close option. On the other hand, if the Folder Automatic Close option specifies to close the folder when opening an object inside the folder, there is no way to dynamically override the option and keep the folder open.

The Undo push button can be used to change the Folder Automatic Close property of the folder back to the value that the folder had when the second Window page was first opened.

The Default push button can be used to reset the Folder Automatic Close property of the folder to the system default value, which is specified on the second Window page of the System object.

WPDesktop — the Desktop Folder object

The Desktop Folder object is a special folder that covers the screen and acts as the base for all windows. The open view of the Desktop is special because its title bar is above the screen and the sizing border is just outside the visible area of the screen. The only part of the Desktop window that you can see is the client area. You can't minimize or hide this window, and no other window can ever be behind it, so the Desktop will never cover any other window.

The Desktop is also a Work Area Folder. When you start the system, any objects that were open on the Desktop Folder are automatically reopened.

WPDesktop — properties

The properties defined by the WPDesktop class include

- Lockup Characteristics — This set of properties includes the timing and other characteristics for automatic lockup of the keyboard and mouse, as well as the background color or bitmap that you see when the system is locked up. These properties can be set only from the Lockup pages of the Desktop's Settings Notebook.
- Archive Characteristics — This set of properties includes information about whether or not to create an archive of the key system files when the system is started, where to put the archive files, and characteristics of the recovery dialog box. See Chapter 11 for information about the archive and recovery facilities.

- **Save Settings Option** — This property indicates whether or not the system is to save the state of the Desktop when the system is shut down. You can specify the value of this property on the Desktop page of the Settings Notebook.

WPDesktop — behaviors

The behaviors that are defined by the WPDesktop class include

- **Lockup** — This is a facility to lock the keyboard and mouse automatically after a certain period of inactivity. You can also request that the system lock the system at any time. After the system is locked, you have to specify the correct password to unlock the system. The lockup facility is described below, under Lockup pages.
- **Archive** — This is a facility to automatically copy critical system files so you can restore the state of the system after a catastrophic system failure. This facility is described in Chapter 11.
- **Save Desktop Settings** — The Workplace Shell automatically saves the position of all open views of a folder and the positions of all icons within the folder when you close the open view of the folder. The next time you open that folder, the window containing the folder view is repositioned to the same location on the screen, and all icons within the folder are restored to their previous positions. This allows you to shut down the system and see the same objects in their same positions when you restart the system. If the folder is a Work Area Folder, the system also remembers which objects within the folder are open, so they can be reopened when the Work Area Folder is reopened.

WPDesktop — pop-up menu items

The WPDesktop class adds the following items to the pop-up menu for the Desktop:

- **Lockup now** — You can request that the system lock the keyboard and mouse at any time by selecting the **Lockup now** pop-up menu item of the Desktop Folder.
- **Shut down...** — You can shut the system down by selecting the **Shut down...** pop-up menu item.

Warning

Because the Workplace Shell and the file systems cache information, you should shut down the system before turning off your computer so the system has a chance to write all information from the cache onto the fixed disk. If you don't shut down the system before turning off your computer, you risk losing information that is currently stored in the cache. If critical system files are not properly written out before turning off the computer, you might not be able to restart the system.

Uncensored

Occasionally, the shut down procedure fails to complete. There are several situations that you can recognize. First, if you watch the messages being displayed by the shut down procedure, you might see the screen blanked out and the "Please wait, shutdown still in progress..." message box displayed and stay on the screen for a while, then get a blue border (actually, this is the focus window border color) and the system hangs. If nothing happens after this and the disk activity light on your system unit stays off, the shut down procedure has actually completed but the Workplace Shell can't write the final message box on the screen. (This is usually seen in a memory-constrained system.) It is safe to turn off your computer. Sometimes, the "Please wait, shutdown still in progress..." message box is erased from the screen and the "Shutdown has completed. It is now safe to turn off your computer..." message box is only partially written to the screen (sometimes you see just the box with no text in it, sometimes you see the box with one or two lines of text instead of all three lines). Again, everything is OK, and you can turn off your computer safely.

In other cases, the shut down procedure simply fails to complete. This is sometimes caused by a particular application program refusing to shut down or some other problem within the system. If you request that the system shut down and there are no messages indicating what is going on, look at the disk activity light on your system unit. If the light is flashing, the system is trying to shut down. It may simply be taking a long time. If there is no disk activity for a long time (30 seconds or so), the shut down procedure has probably failed. Try to shut the system down by rebooting the system (that is, press **Ctrl-Alt-Del**). If nothing happens within 30 seconds, try pressing **Ctrl-Alt-Del** again. If this doesn't force a shut down, try pressing **Ctrl-Alt-NumLock** twice. If the screen blanks out and the message "Insert a blank formatted diskette to begin dumping memory." is displayed, press **Ctrl-Alt-Del** again. This sometimes gets the system to shutdown. If nothing works, either press the Reset button on your system unit (if it has one) or turn the system unit off and then on. Let the system restart (including the automatic CHKDSK function caused by the improper shut down). When the Workplace Shell has started, shut down the system again. Hopefully, you will not lose any valuable information because of the forced reset of the system.

- **System setup** — You can open the System Setup folder by selecting the **System setup** pop-up menu item. This gives you a quick way of customizing the system.

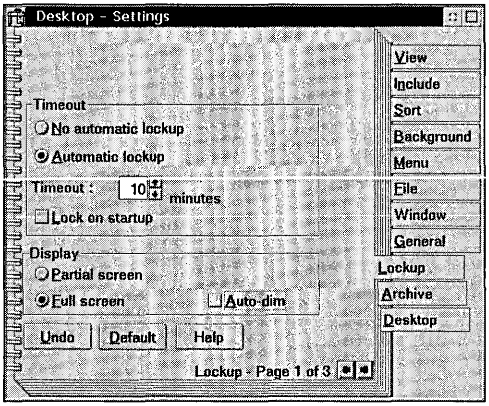
  If you select the **System setup** pop-up menu item and nothing happens, the System Setup folder may have lost its Object Identifier (<WP_CONFIG>). In this case, you can run the REXX script on the CD-ROM included with this book that recreates the System Setup folder to correct the problem.

WPDesktop – Settings Notebook pages

These pages are added to the Settings Notebook for the Desktop folder.

Lockup pages. The three Lockup pages allow you to specify the options for the lockup facility.

Figure 5-28. WPDesktop - Lockup Page 1



Desktop - Settings

Timeout
○ No automatic lockup
● Automatic lockup
Timeout : 10 minutes
□ Lock on startup

Display
○ Partial screen
● Full screen
□ Auto-dim

Lockup - Page 1 of 3

The first Lockup page (see Figure 5-28) allows you to specify the timing characteristics for the lockup facility.

Select the **No automatic lockup** radio push button if you don't want the system to automatically lock the keyboard and mouse.

Select the **Automatic lockup** radio push button if you want the system to automatically lock the keyboard and mouse after a period of inactivity. You can specify the amount of time to wait before the system locks up in the **Timeout** spin push button. After each keystroke or mouse button press, the system automatically starts a timer. If there is no further keyboard or mouse button activity for at least the amount of time (in minutes) specified in the Timeout spin push button, the system automatically locks up the system. If you press a key or press a mouse button, the timer is restarted.

You can also specify that the system automatically locks the keyboard and display whenever the system is started by checking the Lock on startup check box. If you select this option, you will have to type in your password before doing any work on the system.

You can use the options in the Display group to indicate what the system is to display when it locks the system, either due to the automatic timeout or due to you manually locking the system (from the Desktop's pop-up menu or from the LaunchPad). You can select one of the following radio push buttons:

- ✗ **Partial screen** — The system displays a small window that prompts you to enter the password
- ✗ **Full screen** — The system clears the entire screen and displays the background specified on the second Lockup page

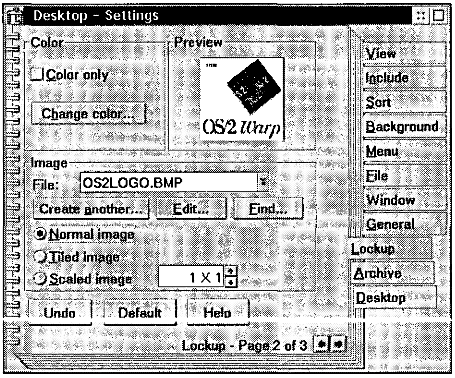
You can also request a primitive screen saver function by checking the Auto-dim check box. If you select this option, the system clears the screen approximately two minutes after the system is locked up and moves a lock icon around the screen to remind you that the system is locked up. When you start typing in the password, the system restores the screen to the state it was in before the Auto-dim function started.

Uncensored

The two-minute timeout for the Auto-dim function cannot be changed. The Timeout value that you can specify is for the keyboard/mouse inactivity only.

You can use the second Lockup page (see Figure 5-29) to specify the appearance of the screen when the system is locked and you select the Full screen display option. You may notice that this Settings Notebook page looks identical to the Background page in the Settings Notebook. See the description of the Folder Background page in Chapter 6 for a full explanation of how to specify the lockup background.

Figure 5-29. WPDesktop - Lockup Page 2



Desktop - Settings

Color

□ Color only

Preview

Image

File: OS2LOGO.BMP

● Normal image

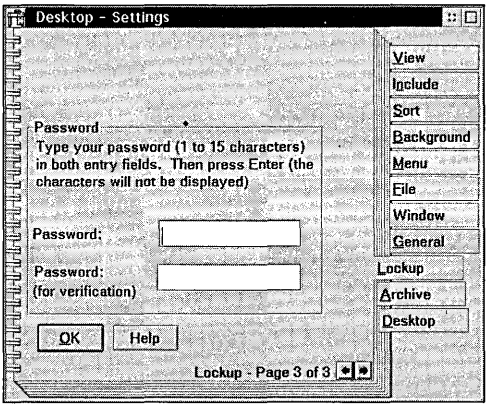
○ Tiled image

○ Scaled image 1 X 1

Lockup - Page 2 of 3

You can use the third Lockup page (see Figure 5-30) to specify the password that you need to type to unlock the system. You must specify the same password in both of the Password entry fields. If you don't specify the password before the system locks up, the system will automatically prompt you for the password before locking the system.

Figure 5-30.
WPDesktop - Lockup Page 3



Desktop - Settings

Password

Type your password (1 to 15 characters) in both entry fields. Then press Enter (the characters will not be displayed)

Password:

Password: (for verification)

Lockup - Page 3 of 3

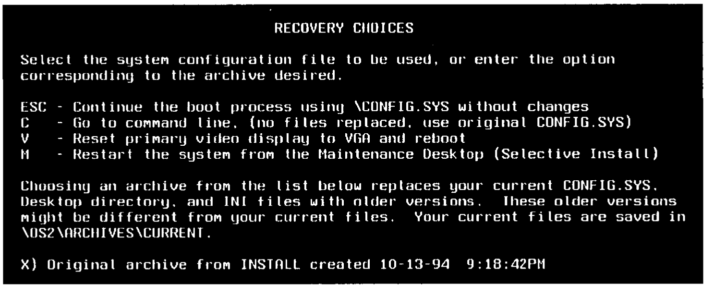
Warning

If you forget your lockup password, you won't be able to unlock the system. (You also won't be able to shut down the system, either through the Desktop pop-up menu, the LaunchPad, or the Ctrl-Alt-Del key sequence.) This can be particularly troublesome if you specify the Lock on startup option. If this option is set, the system will automatically lock up when the Workplace Shell is started. This makes it rather difficult to change your password to something you can remember.

Uncensored

If you find yourself in this situation, there is a way out. Restart the system (if the system is locked up, the only way to restart the system is by pressing the Reset button on your system unit or by turning the system unit power off then on), press Alt-F1 when you see the word OS/2 in the upper left corner of the screen, and then specify C on the Recovery Choices screen (see Figure 5-31) to get a command prompt. When the command prompt appears, you can switch to the OS2 directory (by issuing the command CD \OS2) and issue the command MAKEINI OS2.INI LOCK.RC. This removes the lockup password from the system. You can then restart the system by pressing **Ctrl-Alt-Del**. Because the system now doesn't have a password, it prompts you for the password before locking the system.

Figure 5-31.
Recovery Choices Screen

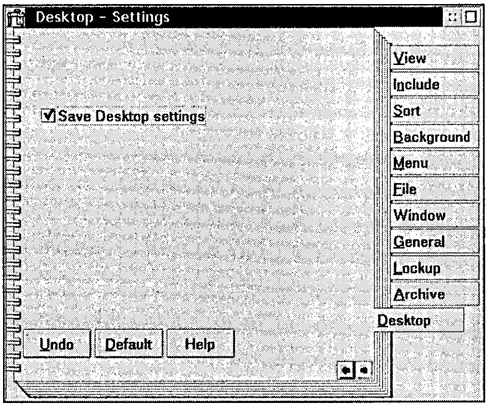


```
                         RECOVERY CHOICES

Select the system configuration file to be used, or enter the option
corresponding to the archive desired.

ESC - Continue the boot process using \CONFIG.SYS without changes
C   - Go to command line. (no files replaced, use original CONFIG.SYS)
V   - Reset primary video display to VGA and reboot
M   - Restart the system from the Maintenance Desktop (Selective Install)

Choosing an archive from the list below replaces your current CONFIG.SYS,
Desktop directory, and INI files with older versions.  These older versions
might be different from your current files.  Your current files are saved in
\OS2\ARCHIVES\CURRENT.

X) Original archive from INSTALL created 10-13-94  9:18:42PM
```

Archive page. The Archive page is described in Chapter 11.

Desktop page. The Desktop page (see Figure 5-32) allows you to specify whether or not to save the state of the Desktop when the system is shut down.

Figure 5-32.
WPDesktop - Desktop Page



Desktop - Settings

☑ Save Desktop settings

If you check the Save desktop settings check box, the system automatically saves the following information when you close a folder:

- ✗ The location of the folder view's window on the screen
- ✗ The location of all icons within the folder view
- ✗ The list of all objects that are open if the folder is marked as a Work Area

The system also automatically saves this information for all folders when you check this box.

If you check this box, the system can restore the Desktop to the state it was in when you shut down. If you remove the check from this box, the state of the Desktop that is restored when the system starts is the state that was saved last. This allows you to arrange the Desktop the way you want it, save its state, and have this state restored from this time on, no matter what the Desktop looks like when you shut the system down.

WPDrives — the Drives Folder object

The Drives Folder object is a special folder that contains WPDisk Objects for all diskette drives, fixed disks, CD-ROMs, PCMCIA devices, and network-attached drives. The Drives Folder object automatically checks the status of the system when it is opened and makes sure it has icons for each file system device currently attached to the system.

WPDrives — properties

There are no new properties defined by the WPDrives object class.

WPDrives — behaviors

There are no new behaviors defined by the WPDrives object class.

WPDrives — pop-up menu items

The WPDrives class adds the following item to the pop-up menu for the drives folder:

- Create partition... — You can invoke the FDISKPM program by selecting the Create partition... pop-up menu item.

WPDrives — Settings Notebook pages

There are no new Settings Notebook pages defined by the WPDrives object class.

WPMinWinViewer — the Minimized Window Viewer Folder object

The Minimized Window Viewer folder object is a special folder that contains WPMinWindow objects for all objects that have been minimized to the Minimized Window Viewer.

WPMinWinViewer — properties

There are no new properties defined by the WPMinWinViewer object class.

WPMinWinViewer — behaviors

There are no new behaviors defined by the WPMinWinViewer object class.

WPMinWinViewer — pop-up menu items

There are no new pop-up menu items defined by the WPMinWinViewer object class.

WPMinWinViewer — Settings Notebook pages

There are no new Settings Notebook pages defined by the WPMinWinViewer object class.

WPNetGrp — the Network Group Folder object

The Network Group Folder object is a special folder that contains information about a network-attached program group. See Chapter 9 for information about the Network Group Folder object.

WPNetwork — the Network Folder object

The Network Folder object is a special folder that contains information about all network attachments. See Chapter 9 for information about the Network Folder object.

WPProgramGroup — the Program Group Folder object

The Program Group Folder object is a special folder that behaves like a program group did in OS/2 1.x.

WPRootFolder — the Root Folder object

The Root Folder object is a special folder that represents the root directory of a Disk object.

WPRootFolder — properties

There are no new properties defined by the WPRootFolder object class.

WPRootFolder — behaviors

There are no new behaviors defined by the WPRootFolder object class.

WPRootFolder — pop-up menu items

There are no new pop-up menu items defined by the WPRootFolder object class.

WPRootFolder — Settings Notebook pages

There are no new Settings Notebook pages defined by the WPRootFolder object class.

WPServer — the File Server Folder object

The File Server Folder object is a special folder that contains information about a network-attached File Server. See Chapter 9 for information about the File Server Folder object.

WPSharedDir — the Shared Directory Folder object

The Shared Directory Folder object is a special folder that contains information about a network-attached directory. See Chapter 9 for information about the Shared Directory Folder object.

WPStartup — the Startup Folder object

The Startup Folder object is a special folder that contains objects that are automatically opened when the system is started.

There are several ways of starting programs automatically when the system is started:

- ✗ The STARTUP.CMD command file is executed when the Workplace Shell is started. This command file can be a plain command file or a REXX script. It normally contains statements that start programs.
- ✗ The Startup Folder contains objects or shadows of objects that are opened automatically when the Workplace Shell is started, unless the CONFIG.SYS file contains the statement SET RESTARTOBJECTS=NONE. See Chapter 10 for a description of the SET RESTARTOBJECTS= statement.
- ✗ All objects that were open (including programs that were running) when the system was shut down are automatically reopened when the Workplace Shell is started, unless the CONFIG.SYS file contains the statement SET RESTARTOBJECTS=NONE or SET RESTARTOBJECTS=STARTUPFOLDERSONLY.

There can be one or more Startup Folders defined in the system. When the Workplace Shell is started, all Startup Folders are located and all objects contained within them are opened.

Uncensored

The order that objects within the Startup Folder are opened is not easily controlled. If you open the Startup Folder in Flowed Icon view, you see the objects in the order that they will be opened.

WPStartup — properties

There are no new properties defined by the WPStartup object class.

WPStartup — behaviors

There are no new behaviors defined by the WPStartup object class.

WPStartup — pop-up menu items

There are no new pop-up menu items defined by the WPStartup object class.

WPStartup — Settings Notebook pages

There are no new Settings Notebook pages defined by the WPStartup object class.

WPTemplates — the Templates Folder object

The Templates Folder object is a special folder that contains template objects for all registered object classes. The Templates Folder object automatically checks all registered object classes when it is opened and makes sure it contains a template for each class that doesn't have a class style containing the CLSSTYLE_NEVERTEMPLATE option. See Chapter 2 for a description of templates.

WPTemplates — properties

There are no new properties defined by the WPTemplates object class.

WPTemplates — behaviors

There are no new behaviors defined by the WPTemplates object class.

WPTemplates — pop-up menu items

There are no new pop-up menu items defined by the WPTemplates object class.

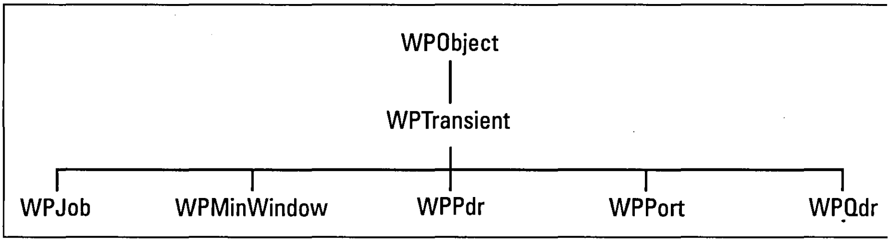
WPTemplates — Settings Notebook pages

There are no new Settings Notebook pages defined by the WPTemplates object class.

Transient Objects

Figure 5-33 shows all Workplace Shell classes that are derived from the WPTransient class.

Figure 5-33. Workplace Shell Object Class Hierarchy - Transients



WPJob — the Job object

The Job object represents a printer job. The Job objects are created when you open a view of a Printer object. Each Job object represents one file that is being printed or is waiting to be printed. See Chapter 14 for a complete description of the WPJob Object.

WPMinWindow — the Minimized Window object

The Minimized Window object represents a window that has been minimized to the Minimized Window Viewer. The Minimized Window objects are always found within the Minimized Window Viewer Folder. These objects are created if you minimized a window, and the Settings Notebook for the object specifies the Minimize window to viewer option. Once it is created, a Minimized Window object is destroyed only when the window it represents is closed. If you restore or maximize the minimized window, the Minimized Window object is hidden until it is needed again.

WPMinWindow — properties

The properties defined by the WPMinWindow class include

- **Original Window** — This is the handle of the window represented by the Minimized Window object.
- **Icon** —This is the normal icon of the object represented by the Minimized Window object. If the original object is a folder, this is always the closed folder icon.

WPMinWindow — behaviors

The behaviors that are defined by the WPMinWindow class include

- **Manipulate Original Window** — The Minimized Window object can restore, maximize, or close the original window.

WPMinWindow — pop-up menu items

The WPMinWindow class has the following items in its pop-up menu:

- Help — You can request information about the original object by selecting the **Help** pop-up menu item.
- **Pickup** — You can initiate a Pickup/Drop operation for the original object by selecting the **Pickup** pop-up menu item. See Chapter 3 for information about Pickup and Drop.
- **Restore** — You can restore the minimized window to its previous size and location on the screen by selecting the **Restore** pop-up menu item.
- **Maximize** — You can maximize the minimized window by selecting the **Maximize** pop-up menu item.
- Close — You can close the minimized window by selecting the **Close** pop-up menu item.

Uncensored

If the minimized window is a view of a Workplace Shell object, the next time you try to open the object it will open in the minimized state. For example, if you open an Icon view of the OS/2 System Folder, minimize it to the Minimized Window Viewer, and close the window from the pop-up menu of the Minimized Window object, the next time you try to open the OS/2 System Folder you will not see it open. The folder's icon will become cross-hatched and the icon will change to the open folder icon, but you won't see the folder open. If you look inside the Minimized Window Viewer you will see the OS/2 System Folder. You can restore the window by double-clicking on the object's icon again, by double-clicking on the Minimized Window's icon in the Minimized Window Viewer or by selecting **Restore** from the Minimized Window's pop-up menu.

- **Window list** — You can display the Window List by selecting the **Window list** pop-up menu item.

WPMinWindow — Settings Notebook pages

The Minimized Window object doesn't have a Settings Notebook.

WPPdr — the Printer Driver object

The Printer Driver object represents a printer driver. The Printer Driver objects are created when you open the Settings Notebook of a Printer object and turn to the Printer driver page. The Printer Driver objects are deleted when you close the Settings Notebook. See Chapter 14 for a description of the Printer object's Settings Notebook.

WPPort — the Printer Port object

The Printer Port object represents a printer port. The Printer Port objects are created when you open the Settings Notebook of a Printer object and turn to the Output page. The Printer Port objects are deleted when you close the Settings Notebook. See Chapter 14 for a description of the Printer object's Settings Notebook.

WPQdr — the Print Queue Driver object

The Print Queue Driver object represents a print queue driver. The Print Queue Driver objects are created when you open the Settings Notebook of a Printer object and turn to the Queue options page. The Print Queue Driver objects are deleted when you close the Settings Notebook. See Chapter 14 for a description of the Printer object's Settings Notebook.

Summary

This chapter could have been titled, "Everything you always wanted to know about Workplace Shell objects but were afraid to ask." I suspect that before you read this chapter you didn't even know what to ask. Hopefully, I have shed enough light on the object class structure used within the Workplace Shell to make it more usable.

For the most part, the object hierarchy is important to you only when you are setting up Find, Sort, or Include criteria. The rest of the time, you don't really care what object classes you are dealing with. However, you should now understand why some objects behave differently than other objects and why Settings Notebook pages and pop-up menu items come and go. Once you understand the options that are available within the Workplace Shell, you can use them to your advantage.

For example, I got tired of seeing the LaunchPad icon on the Desktop, since the LaunchPad itself was always open. My first thought was to hide the LaunchPad icon behind the LaunchPad window. This didn't really work, since it didn't entirely fit behind the window. My second idea, which worked much better, was to hide the LaunchPad by using the Desktop's Include page to exclude all objects of class WPLaunchPad. Once I saw how the Include page works, I was also able to exclude several other objects from view — objects that I have open all the time. Again, I want to see the open view of the object, not the object's icon.

Ira H. Schneider is an Advisory Programmer on the Workplace Shell Development Team. He has been instrumental in developing OS/2 Warp, including performance improvements and functional enhancements to the Workplace Shell. He has also worked on the OS/2 Swapper and on OS/2 System Initialization, and he was Lead Architect for OS/2 1.2. Ira has been with IBM since 1969.

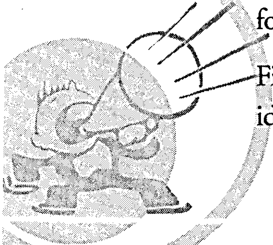
Chapter 6

Customizing the Appearance of Windows

Colors and Fonts and Schemes, Oh My!

There are many ways you can customize the appearance of windows within OS/2 Warp. In this chapter, I will explain what window elements you can customize and how to use the Workplace Shell to make your windows look the way you want.

What You Can Customize



The window elements you can customize fall into three categories: colors, fonts, and folder background bitmaps.

Figure 6-1 shows a typical window. It is an open icon view of a folder. Do you have any idea how many different elements of this window can be customized?

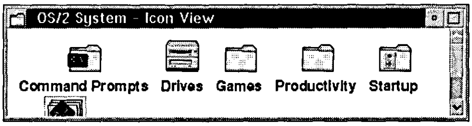


Figure 6-1.
Typical Window

You will probably be surprised to learn that 29 elements can be customized on this one window (see Figure 6-2)! These window elements can be specified:

1 Active title bar color

2 Active title bar text color

3 Active title bar text background color

4 Inactive title bar color

5 Inactive title bar text color

6 Inactive title bar text background color

7 Title bar bottom color

8 Title bar text font

9 Active window border color

10 Inactive window border color

11 Scroll bar background color

12 Scroll bar slider color (push button color)

13 Folder background color or bitmap

14 Frame color

15 Icon text color

16 Icon text font

17 Icon text background color

18 Icon text visibility

19 Shadow text color

20 Shadow text highlight color

21 Shadow text highlight background color

22 Menu background color

23 Menu text color

24 Menu text font

25 Menu disabled text color

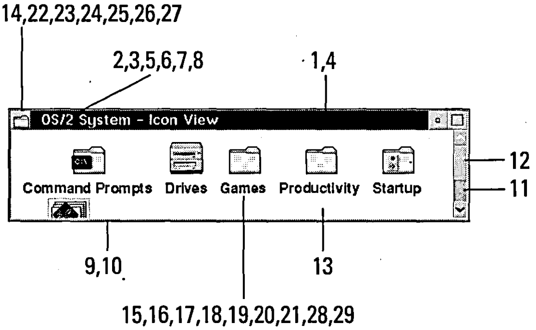
26 Menu highlight text color

27 Menu highlight background color

28 Highlight text color

29 Highlight background color

Figure 6-2. Customizable Elements in the Typical Window



You can specify the colors to be used for most elements of a window. This may be a window within the Workplace Shell, such as an icon view of a folder, a dialog box, or a settings notebook, or it may be a window displayed by an application program. When you specify the colors to be used for windows within the Workplace Shell, we save the colors so your preferences will be honored the next time you display the same window. However, within an application program, the program is responsible for saving the window's characteristics. For a well-behaved application program, the colors and fonts you specify will take effect immediately and will be saved. Other programs may not respond to the customization at all, or may not save the customized colors or fonts. You will have to experiment with your application programs to determine whether they allow you to change their colors or fonts.

You can change the fonts used to display text within windows. This includes the text in the title bar, menu bar, context menus, text below icons, and other text within the windows.

You can also specify a bitmap (a picture file) to be used as the background for the open views of a folder within the Workplace Shell, including the Desktop.

Colors

Before I describe what window elements you can customize, I want to give you a short explanation about the use of colors on your display.

On most modern displays, colors are composed of a mixture of red, green, and blue light, which are the primary colors on a display. The proportion of the primary colors determines the color that you see. You can specify an amount between 0 and 255 for each of the primary colors. A pure bright red area on the screen is composed of red, green, and blue light mixed in the proportion of 255, 0, and 0. A bright white area on the screen has color values of 255, 255, and 255. A black area on the screen has color values of 0, 0, and 0 (that is, no light at all).

The way the color information is presented to the display (or monitor) is controlled by the display adapter in your computer. Three main modes are commonly in use by today's display adapters:

✘ **16-color mode** — The display adapter can display only 16 unique colors on the screen at any time. Generally, this mode allows you to choose any 16 colors out of a total of 64 or 256 colors. For example, the IBM Video Graphics Array (VGA) supports only 16-color mode.

✘ **256-color mode** — The display adapter can display any 256 unique colors on the screen at any time. Generally, this mode allows you to choose any 256 colors. Each color can have any color value. For example, the IBM Extended Graphics Array (XGA) and most super VGA adapters support 256-colors at some resolution.

✘ **Direct-color mode** — The display adapter can display any color at any location on the screen. In this mode, there is no limit to the number of unique colors that can be displayed at any one time. For example, many super VGA adapters support direct-color mode at lower resolutions.

In 16-color mode, all programs are limited to the 16 colors currently selected by the display adapter. You can display text and windows in limited color. You can't display good bitmaps in 16-color mode, since most good bitmaps need more colors.

In 256-color mode, the combination of colors displayed by all currently visible windows is limited to 256 unique colors. You can display text and windows in a wide range of colors. Most bitmaps look good in 256-color mode. However, programs must cooperate to manage the 256 colors, which can be displayed at any time. The OS/2 Palette Manager coordinates these programs.

In direct-color mode, since the number of colors on the screen is unlimited, there is no need for coordination between programs — each program can display any colors it needs.

The color mode used by your display adapter is generally limited by the hardware installed in your computer and the screen resolution you want to use. For example, a super VGA display adapter may support 256-color mode when running with a resolution of 1024 x 768 pixels (that is, 1024 columns and 768 rows of color dots on the screen) or direct-color mode when running with a resolution of 640 x 480 or 800 x 600 pixels. In this case, you can choose whether you want higher resolution or a larger number of colors on the screen. The limiting factor is often the amount of memory installed on the display adapter. With some display adapters, you can add more memory to the display adapter to be able to use direct-color mode at a higher resolution.

Palette Manager

If your display adapter is set up for 256-color mode, you will be affected by the Palette Manager. If your display adapter is set up for 16-color mode or direct-color mode, you can skip this section, because it doesn't apply to you.

Colors in some windows, including open views of folders, may not look the way you expect.

Uncensored

It is important to understand how colors are managed when selecting the colors to be used for the various window elements. For example, if you choose a beautiful shade of cyan for the active title bar and you invoke a palette-managed program, the active title bar may change to an ugly shade of blue. Even worse, if you select a pale yellow for the Desktop background, you may find your Desktop changes to white after certain programs are invoked. This section explains why this "color corruption" happens and what you can do to prevent it.

Palette Manager

Note

I am going to describe the Palette Manager as if it really exists in the system. In reality, no such component exists within OS/2. The Palette Manager function is spread out among the components that comprise the graphics subsystem. However, the concept of palette management is easier to talk about by referring to the Palette Manager.

Suppose you have two windows open on the screen. In one of these windows, you are running a program that is trying to display a pattern containing 256 shades of blue. In the other window, you are running the same program, but there the program is trying to display a pattern containing 256 shades of red. Because your display adapter is capable of displaying only 256 unique colors, something must decide what set of colors to display. That something is the Palette Manager.

✗ When the system is started, the Palette Manager initializes your display adapter to display 256 colors known to the system. These 256 colors are the default palette. If a program tries to paint an area of a window with a color, the Palette Manager sees if the specified color is in the default palette. If it is, it paints the window with the specified color.

If that color is not in the default palette, the Palette Manager tries to construct the requested color by combining two colors from the default palette. The window is painted with an alternating series of dots of the two selected colors. This process of combining two colors to simulate another color is called *dithering.*

✗ Until a palette-managed application program is invoked, the Palette Manager has the full 256 colors that it can use to dither, and it will probably be able to construct a close approximation of the requested color. If a program tries to display a bitmap, the Palette Manager can't use dithering to simulate colors, because each dot on the screen is already specified as a color in the bitmap. As the bitmap is painted on the screen, the Palette Manager tries to match each dot in the bitmap to the closest color in the default palette. A bitmap that is constructed from colors in the default palette will look correct. A bitmap that is constructed from a set of custom colors may not look very good. In fact, in the extreme case, the bitmap will be totally unrecognizable!

- ✘ The Palette Manager provides a facility for programs to display bitmaps correctly in this situation. Whenever a window is given the focus (that is, when you click on the window), the program has the opportunity to tell the Palette Manager that it has a requirement for a specific set of colors. The Palette Manager will save up to 236 of those colors and will start constructing a color set to be sent to the display adapter. The remaining 20 colors are reserved for use by the system. (Actually, a program can request the use of all 256 colors. This feature is rarely used, so the system usually has 20 colors that it can count on.)

- ✘ The Palette Manager then asks all other open windows if they have any color requirements. If they do, these colors are added to the color set, until the color set is full. At this point, the Palette Manager reinitializes the display adapter with the new color set. When this happens, you will see the screen flash, and all windows will probably look terrible.

- ✘ The Palette Manager then instructs all open windows to repaint themselves. When the windows are repainted, the windows that are painted by programs that responded to the Palette Manager will look as good as possible, because the Palette Manager can match their color requirements to the color set. However, the other windows may not look as good. For each color a nonpalette-managed program tries to use, the Palette Manager again looks to see if that color is in the current color set. If it is, that color is used, and it looks correct. If the color is not in the current color set and must be constructed by dithering, the Palette Manager can't use the entire 256 color set to dither from — it can use only the 20 reserved colors.

  Because only 190 colors can be constructed from the 20 reserved colors (actually, there are slightly fewer than 400 combinations of the 20 reserved colors), the chance of exactly matching the requested color is much smaller. The window area is painted with the color that is closest to the requested color, but it may not match exactly. In fact, it may not match at all. For example, if a program tries to paint an area of its window with a pale yellow, the closest color the Palette Manager may be able to construct is white.

✘ Every time a different window is given the focus, this color selection process is repeated. In my example of the two windows, one trying to display 256 shades of blue and one trying to display 256 shades of red, when the blue window is given the focus the color set constructed by the Palette Manager will contain mostly different shades of blue, plus the 20 reserved colors.

The blue scale window will look pretty much correct. However, because the 256 shades of red must be constructed from a very limited set of reds, the red scale window will probably contain only a few shades of red and will not look right.

If you now click on the red scale window, the color set will be changed to a predominately red color set, the red window will look good, and the blue window will have only a few shades of blue.

All other windows will probably look the same in either case, because most of their colors are coming from the same reserved 20 colors.

✘ When all programs that understand palette management close their windows, the system has a chance to revert redundent to its initial state. If you click on three windows that do not participate in palette management (or click back and forth between two such windows), the Palette Manager reloads the default color set and understands that all 256 colors are again available for dithering. The screen will flash and all windows will be repainted.

In OS/2 Warp, we added limited palette management support into the Workplace Shell. If you open the System Object in the System Setup folder and click on on the **Workplace shell palette aware** check box on the first Screen page, the behavior of the system is changed in two ways.

1 When you give focus to an open view of a folder (or the Desktop) that contains a background bitmap, the Workplace Shell informs the Palette Manager that it needs a set of colors to paint the bitmap. This starts the color set selection process discussed earlier in this section. On the other hand, if you give focus to an open view of a folder that has only a background color defined, the Workplace Shell does not communicate with the Palette Manager. Its window must be painted with the closest color from the current color set. (This was done so the system doesn't flash and repaint all windows every time you open a folder or switch between folders.)

2 The Palette Manager *never* restores the system to its initial state. You can click back and forth between nonpalette-managed windows until you are blue in the face, but the Palette Manager will not reload the default color set.

You will simply have to experiment with having the **Workplace shell palette** aware option on or off and choose the setting that is most pleasing to you.

Tip

I would strongly suggest that you display a 256-color bitmap using a palette-managed program while you are trying to select colors to be used on your system. In this way, you will see the worst possible scenario, and can select colors that will still look good when a palette-managed program is running. The simplest way to display a 256-color bitmap is to turn on the **Workplace shell palette aware** option and assign the bitmap as the background of a folder, then open the folder. You can then select colors that look good in the worst of situations.

Window elements that you can color

Figures 6-3 through 6-8 show typical windows. Each figure illustrates a particular type of window and shows the window elements for which you can specify the color. Following each figure is a table that explains what the window elements are and how to color them.

The columns in Tables 6-1 through 6-6 contain the following information:

- **Window Element** — The key number from the figure and the name of the window element
- C — You can set this window color using the Color Palette
- N — You can set this window color using the Settings Notebook for the object
- **Default Color** — The default system color value for this window element
- **Notes** — Notes about the window element

All of these window elements can be set using the Scheme Palette.

Uncensored

The system default color values are the values that are set when OS/2 Warp is installed. If you installed OS/2 Warp over a previous version of OS/2, you may not see these colors because we preserve the colors that you previously customized. If you want to reset the system colors to their new default values, open the Scheme Palette, hold the **Alt** key and drop the **Default** scheme on the Desktop. (See the Scheme Palette section later in this chapter.)

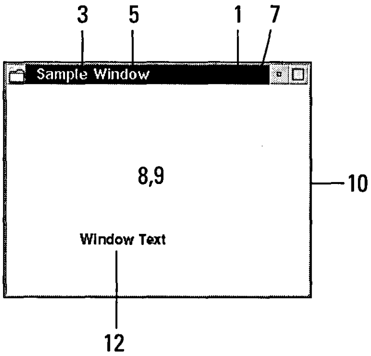


Figure 6-3. Window Elements in the Window That Has the Focus

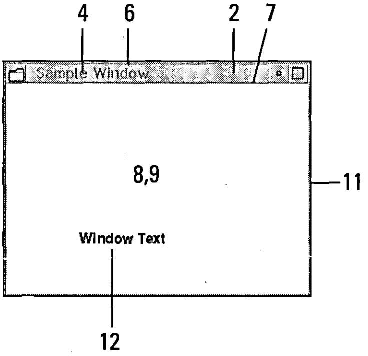


Figure 6-4. Window Elements in a Window That Doesn't Have the Focus

Table 6-1. What You Can Color In A Window

Window Element	C	N	Default Color	Notes
1. Active Title Bar	✓		dark purple	Choose a color that contrasts with the Inactive Title Bar color so you can easily identify the window that has the focus.
2. Inactive Title Bar	✓		light gray	
3. Active Title Bar Text	✓		white	
4. Inactive Title Bar Text	✓		dark gray	
5. Active Title Bar Text Background	✓		dark purple	Normally you would specify this color to be the same as the Active Title Bar color.

Window Element	C	N	Default Color	Notes
6. Inactive Title Bar Text Background	✓		light gray	Normally you would specify this color to be the the Inactive Title Bar color.
7. Title Bar Bottom			dark gray	This is the color of the line separating the title bar from the client area of the window.
8. Window Background	✓		white	This is the color for the background of a window. This is the same color as the Folder Background color.
9. Application Workspace			light gray	This is the color used for the background of application program main windows.
10. Active Window Border	✓		light gray	You may want to choose a color that contrasts with the Inactive Window Border color so you can easily identify the window that has the focus.
11. Inactive Window Border	✓		light gray	
12. Window Text	✓		black	This is the color of text in an application window. It is also the color of the text in the Details View of a folder.

Figure 6-5.
Window Elements in a Menu

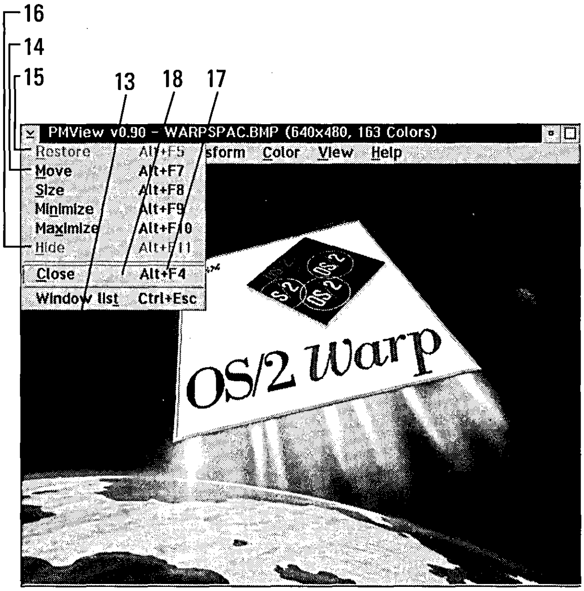
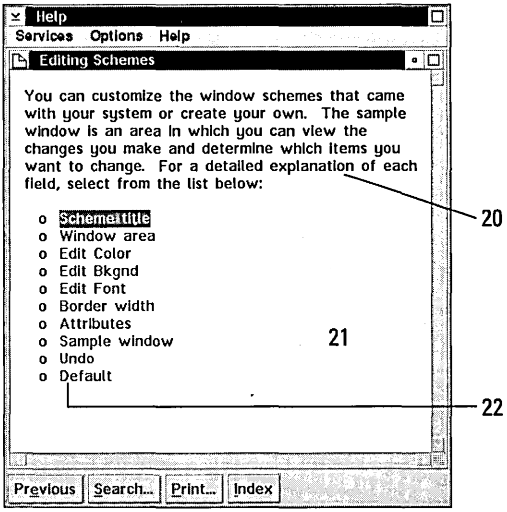


Table 6-2. What You Can Color In A Menu

Window Element	C	N	Default Color	Notes
13. Window Frame			dark gray	This is the color for the frame around a window or a menu.
14. Menu Text		✓	black	This is the color for the text of a nonselected, enabled menu item.
15. Menu Background		✓	light gray	This is the color for the back ground for all nonselected menu items.
16. Menu Disabled Text			dark gray	This is the color for the text of a disabled menu item.
17. Menu Highlight Text			black	This is the color for the text of a selected menu item.
18. Menu Highlight Background	✓		light gray	This is the color for the background for a selected menu item.
19. Shadow			dark gray	This is the color for the drop shadow for menus and dialog boxes (not shown in figures).



You can customize the window schemes that came with your system or create your own. The sample window is an area in which you can view the changes you make and determine which items you want to change. For a detailed explanation of each field, select from the list below:

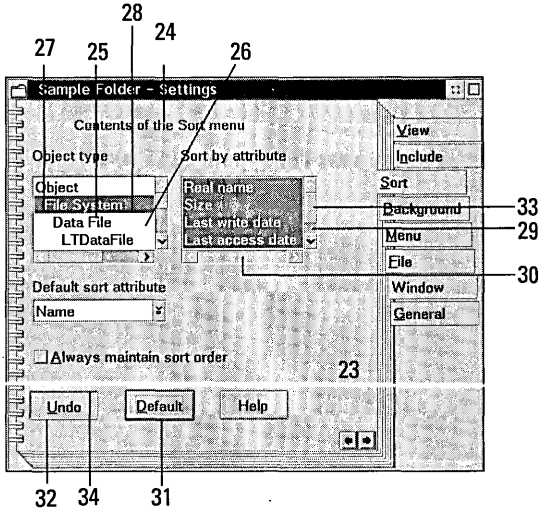
- Scheme title
- Window area
- Edit Color
- Edit Bkgnd
- Edit Font
- Border width
- Attributes
- Sample window
- Undo
- Default

Figure 6-6.
Window Elements in a Help Window

Table 6-3. What You Can Color In A Help Panel

Window Element	C	N	Default Color	Notes
20. Help Text	✓		black	This is the color for normal text on a help panel.
21. Help Background	✓		white	This is the color for the background of a help panel.
22. Help Highlight			blue	This is the color for highlighted text on a help panel.

Figure 6-7.
Window Elements in a Dialog Window



Contents of the Sort menu

Object type: Object, File System, Data File, LTDataFile

Sort by attribute: Real name, Size, Last write date, Last access date

Default sort attribute: Name

□ Always maintain sort order

Table 6-4. What You Can Color In A Dialog Box

Window Element	C	N	Default Color	Notes
23. Dialog Background		✓	light gray	This is the color for the background of a dialog box.
24. Window Static Text			blue	This is the color for static text within a window.
25. Output Text			black	This is the color for nonselected text in an output field, entry field, or list box.
26. Entry Field and List Box Background	✓		yellow	This is the color for the back ground of an entry field and for a list box.
27. Selection Highlight Foreground			white	This is the color used for selected text in an entry field or list box. This is also the color used for the title of a selected object in a folder.
28. Selection Highlight Background			dark gray	This is the color used for the background for selected text in an entry field or list box.This is also the color used for the background for the title of a selected object in a folder and the color around the selected object's icon.
29. Scroll Bar Background	✓		medium gray	This is the color for the background area of an enabled scroll bar.
30. Field Background	✓		light gray	This is the color for the background area of a disabled scroll bar and the default background color for all other controls.
31. Button Default			black	This is the color for the border around the default push button.
32. Button Light			white	This is the light color used when creating the 3D effects for a push button and other dialog elements.

Window Element	C	N	Default Color	Notes
33. Button Middle	✓		light gray	This is the middle color used when creating the 3D effects for a push button and other dialog elements. This is also the color used for the moveable portion of a scroll bar (the slider).
34. Button Dark			dark gray	This the dark color used when creating the 3D effects for a push button and other dialog elements.

Figure 6-8. Window Elements in the Icon View of a Folder

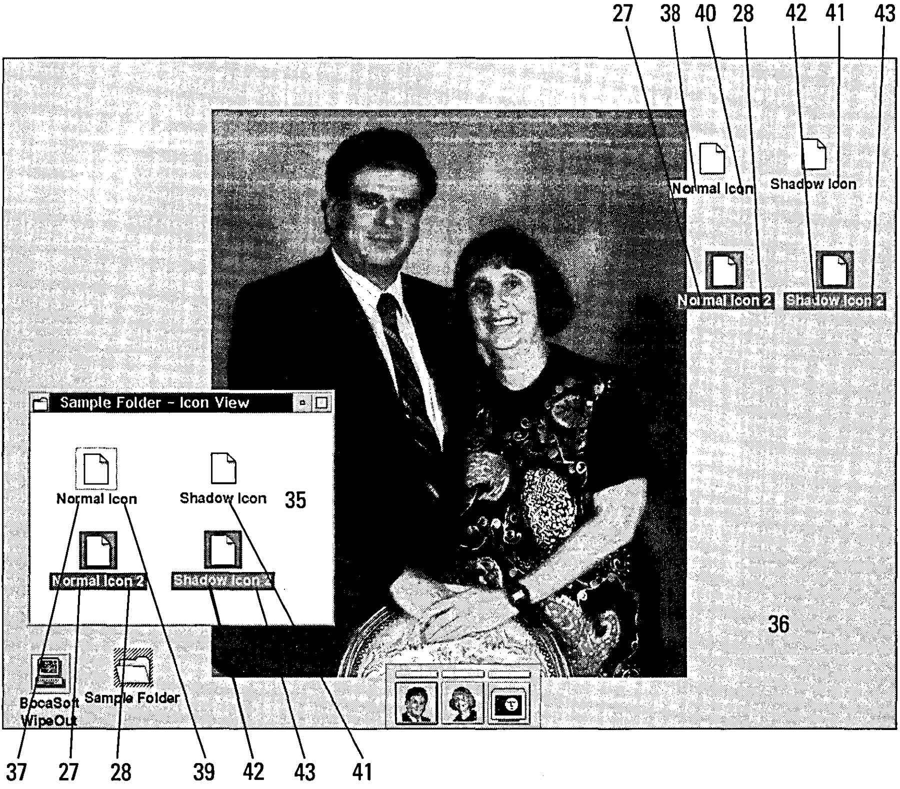


Table 6-5. What You Can Color In A Folder

Window Element	C	N	Default Color	Notes
35. Folder Background	✓	✓	white	This color may not be visible if the folder background also has a bitmap defined.
36. Desktop Background	✓	✓	teal	This color may not be visible if the Desktop folder background also has a bitmap defined.
37. Folder Icon Text	✓	✓	black	This is the color used for the title of a nonhighlighted object in a folder. If you select invisible text, this color is ignored.
38. Desktop Icon Text	✓	✓	black	This is the color used for the title of a nonhighlighted object in any view of the Desktop folder. If you select invisible text, this color is ignored.
39. Folder Icon Text Background		✓	transparent	This is the color of the background for the title of a nonhighlighted object in an icon or tree view of a folder. It is also the color of the background for all nonhighlighted lines of data in a details view of a folder. If this color is specified as transparent, the folder's background color or bitmap shows through around the text.
40. Desktop Icon Text Background		✓	transparent	This is the color of the background for the title of a nonhighlighted object in an icon or tree view of the Desktop folder. It is also the color of the background for all nonhighlighted lines of data in a details view of the Desktop. If this color is specified as transparent, the Desktop's background color or bitmap shows through around the text.

Window Element	C	N	Default Color	Notes
41. Shadow Text			blue	This is the color for the title of a nonhighlighted shadow object in any folder, including the Desktop.
42. Shadow Foreground Highlight			white	This is the color for the title of a selected shadow object in any folder, including the Desktop.
43. Shadow Background Highlight			dark gray	This is the color for the background of the title of a selected shadow object in any folder, including the Desktop. It is also the color that surrounds the selected shadow object's icon.

Table 6-6. Miscellaneous Window Elements You Can Color

Window Element	C	N	Default Color	Notes
44. Page Background			white	This is the color for the background of a notebook page (not shown in figures).
45. Presentation Manager Desktop Background	✓		cyan	This is the color for the background of the Presentation Manager Desktop (not shown in figures). This is the color that covers the screen before the Workplace Shell is started and is the color of the screen after the Workplace Shell terminates. It is also the color for the background for all icons that are minimized to the Desktop, as well as the background color for the text below the minimized icons.

Tip

It may be easier to read the text if you select nondithered colors for the text and the text background window elements, such as the Title Bar Text and Title Bar Text Background.

Tip

If you specify a background bitmap for a folder (or the Desktop), you may want to select a color (that is, not transparent) for the Icon Text Background to make it easier to read the icon text. If the Icon Text color you choose contrasts well with most of the background bitmap, you can select an Icon Text Background color that blends in with the background bitmap to make the text less obtrusive. However, you may want to select an Icon Text Background color that contrasts with the background bitmap if you want the icon text to stand out.

Fonts

Before I tell you what window elements contain fonts that you can change, let me give you a brief explanation of the characteristics of fonts on the display. A font is a particular style of letters, numbers, and special symbols. OS/2 can use an unlimited number of fonts.

There are two different kinds of fonts — the bitmap font and the scaleable font. The bitmap font consists of a set of bitmaps (or pictures) for each letter, digit, and special symbol that can be displayed. Each variation of the font (for example, different letter sizes, normal, bold, italics, and so on) requires a separate set of bitmaps to define the characters. The bitmap fonts, such as Helv, usually have a rather limited set of sizes and other variable characteristics available.

Note

The font named *Helv* is a bitmap font, while the font named *Helvetica* is a scaleable font. Both fonts are part of the Helvetica type face family; the main difference is that the *Helv* font can be displayed in only certain point sizes while the *Helvetica* font can be scaled to any point size.

The scaleable font consists of a set of directions used to draw each character. These directions can be used for a wide range of characteristics. For example, they can be used to draw the font no matter what size the characters are — the system simply has to scale the length of each line segment to fit the specified size. Also, an italic font can be constructed from these directions by simply skewing the lines. You will find that scaleable fonts, such as Helvetica, have a wider range of sizes and other variable characteristics available.

The size of a font is normally specified in a measurement called *points.* When printing text on paper, one point is always 1/72 inches high, so a 12pt font is used to print with a line spacing of 6 lines per inch. However, things are not so simple on the display. Even though OS/2 knows how many pixels high and wide the screen is, OS/2 doesn't necessarily know how many inches high and wide the display is. It is rather difficult for OS/2 to display a 12pt font so that it takes up exactly 1/6 inches if it doesn't know how many pixels make up an inch. So, the system makes its best guess. Sometimes it is right, and sometimes it is wrong.

The best I can say is that the larger the font size you specify, the larger the font will be on the display screen. However, a 12pt Helvetica font may not be the same size as a 12pt Times Roman font. You will have to experiment with the font size until you find the font that you want. Also, with the System VIO font, the size bears little relation to the actual size of the characters — it seems to simply indicate which of the VIO fonts to use. Again, you will have to experiment to see what each size means.

You should probably understand one more thing about fonts before trying to use them. A set of fonts, defined by the International Standards Organization (ISO), is recognized internationally. These fonts look the same worldwide. The non-ISO fonts, like Helvetica, may look slightly different in different countries. Certain countries specify that ISO fonts must be available, so OS/2 provides a set of ISO fonts for everyone to use. Some people like the looks of the non-ISO fonts better. If you select a font such as System Proportional, you may find Non-ISO listed as one of the style choices. In this case, the normal styles are the ISO standard fonts, while the Non-ISO style is the non-ISO font. Again, you will have to decide for yourself what fonts to use.

Window elements that support fonts

Table 6-7 lists the window elements for which you can specify fonts.

The columns in the table contain the following information:

- **Window Element** — The name of the window element
- N — You can set this window font using the Settings Notebook for the folder.
- **Default Font** — The default system font for this window element
- **Notes** — Notes about the window element

All of these window elements can be set using the Font Palette and the Scheme Palette.

Uncensored

The system default fonts are the values that are set when OS/2 Warp is installed. If you installed OS/2 Warp over a previous version of OS/2, you may not see these fonts because we preserve the fonts that you previously customized. If you want to reset the system fonts to their new default values, open the Scheme Palette, hold the **Alt** key and drop the **Default** scheme on the Desktop. (See the Scheme Palette section later in this chapter.)

Table 6-7. What Fonts You Can Assign

Window Element	N	Default Font	Notes
Title Bar Text		10.System Proportional	This font is used for the title bar for all windows.
Menu Text		10.System Proportional	This font is used for all items in a menu.
Window Text		10.System Proportional	This font is used for the text within an application window. It is also the font used for the text within a flowed or nonflowed icon view, a tree view, or a details view of a folder, including the Desktop.
Folder Icon Text	✓	8.Helv or 9.Helv ISO	This font is used for the titles for all objects in a non-grid icon view of a folder. This includes shadow objects.
Desktop Icon Text	✓	8.Helv or 9.Helv ISO	This font is used for the titles for all objects in a non-grid icon view of the Desktop. This includes shadow objects.

If ISO fonts are installed, the default font for the Folder Icon Text and the Desktop Icon Text is either "9.Helv ISO" or "10.Helv ISO".

Folder backgrounds

You can specify a background color or bitmap for the Desktop or any folder within the Workplace Shell. You can specify the background either through the Scheme Palette or the Settings Notebook for the folder or the Desktop.

How to Customize Colors, Fonts, and Folder Backgrounds

Four main facilities of the Workplace Shell can be used to customize the appearance of windows: the Color Palette, the Font Palette, the Scheme Palette, and the Settings Notebook for each folder.

The Color Palette

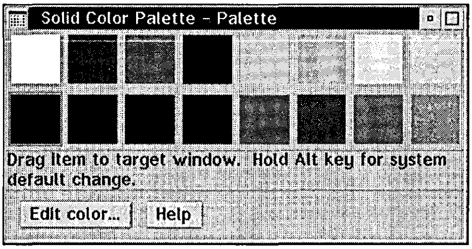
The Color Palette is used to specify colors for window elements.

There are two versions of the Color Palette in the System Setup folder. The Solid Color Palette (see Figure 6-9) contains 16 color squares representing the 16 colors available on all displays, from the simplest EGA display up to the most complex 16 million color capable Super VGA display. The Mixed Color Palette (see Figure 6-10) contains 256 color squares consisting of a wide range of colors. Some of these colors may be solid colors on some displays, while other colors are dithered. If you open the Mixed Color Palette while a palette-managed application program is running, you will notice that some of the colors are missing. This is the problem I described in the Palette Manager section. Some of the colors in the Mixed Color Palette are constructed from colors that are not available in the 20 reserved colors, so they are displayed as the wrong color. However, these are the same colors you will see if you select them for your folder background.

Uncensored

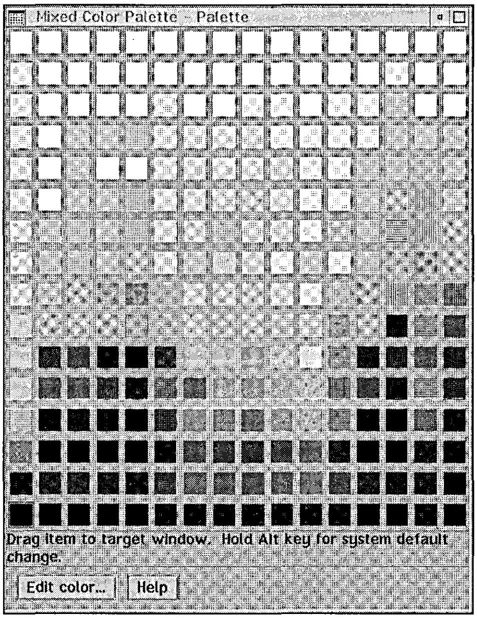
If you installed OS/2 Warp over a previous version of OS/2, a third Color Palette object exists in the System Setup folder. This is the Color Palette from the previous version of OS/2. We don't remove it from the System Setup folder because you may have customized some of the colors previously, and we don't want you to have to recustomize the color palette.

Figure 6-9.
Solid Color Palette



Drag item to target window. Hold Alt key for system default change.

Figure 6-10.
Mixed Color Palette



Drag item to target window. Hold Alt key for system default change.

You can set either foreground or background colors using the Color Palette, and the scope of the color selection can be local or global.

To use the Color Palette, open the System Setup folder and double-click on the **Solid Color Palette** (if you have a 16-color display adapter) or on the **Mixed Color Palette** (if you have a 256 or more color display adapter). You then select the color you want, move the mouse pointer over the desired color square, press and hold mouse button two, drag the mouse pointer (which is now a Paint Can icon with the selected color dripping from it) to the window element you want to color, position the end of the drip inside the window element, and release mouse button two. This will set the background color of the window element to the selected color. See Table 6-8 for the effects of dropping a color on various windows.

You can change the foreground color for many window elements by holding **Ctrl** while dragging the paint can to the window element and dropping the color. See Table 6-8 for the effects of dropping a color on various windows while holding **Ctrl**.

You can change the system default color for many window elements by holding **Alt** while dragging the paint can to the window element and dropping the color. The system default color is used by a window element only if no color has been explicitly assigned to that window element. For example, a folder view will inherit the system default Folder Background color unless a color has been assigned to that folder view. Once a color has been assigned to a window element, it will not inherit the corresponding system default value until the color is removed from the window element, normally by dropping a new color on the window element while holding **Alt**. This changes the corresponding system default color and removes the local color value from the window element.

Table 6-8 shows the colors you can change using the Color Palette. The table contains the following information:

- ✗ **Window** — This is the type of window you drop the color on. This is the window that is under the color drip when you release mouse button two.
- ✗ **Background Color Changed** — This is the window element that is changed if you release mouse button two while not holding **Ctrl**. These window elements are listed in Tables 6-1 through 6-6. If you are holding **Alt** when you release mouse button two, the corresponding system default color is also changed.
- ✗ **Foreground Color Changed** — This is the window element that is changed if you release mouse button two while holding **Ctrl**. These window elements are listed in Tables 6-1 through 6-6. If you are holding **Alt** when you release mouse button two, the corresponding system default color is also changed.
- ✗ **Notes** — These are notes about the effects of dropping a color on particular windows.

Table 6-8. Colors You Can Change Using The Color Palette

Window	Background Color Changed	Foreground Color Changed	Notes
Button	Button Middle	Window Text	If you hold **Ctrl** and **Alt** while dropping a color on a button, the system default Menu Text color is set.
Non-grid icon view of the Desktop	Desktop Background	Desktop Icon Text	If **Alt** is pressed view of while setting the Desktop Background color, the Presentation Manager Desktop color is set to the same color. This color is used as the background of any icon minimized to the Desktop, as well as the icon text background color for these icons. The system default Desktop Icon text color is the color of the text below any icon minimized to the Desktop.
Other view the Desktop	Desktop Background	Window Text	If **Alt** is pressed of while setting the Desktop Background color, the Presentation Manager Desktop color is set to the same color.
Dialog Box	Dialog Background	Window Text	
Entry Field	Entry Field and List Box Background	Window Text	
Non-grid icon view of a Folder	Folder Background	Folder Icon Text	
Other view of a Folder	Folder Background	Window Text	
Frame Window Border of the window that has the focus	Active Window Border	Window Text	

Window	Background Color Changed	Foreground Color Changed	Notes
Frame Window Border of a window that does not have the focus	Inactive Window Border	Window Text	
Help Panel	Help Background	Help Text	
List Box	Entry Field and List Box Background	Window Text	
Menu	Menu Background, Menu Highlight Background, and Menu Disabled Background	Menu Text	If you drop a color on a menu while holding **Alt**, only the system default Menu Background and Menu Highlight Background colors are set.
Notebook Page	Window Background	Window Text	No, this is not a typographical error — dropping a color on a notebook really sets the Window Background color instead of the Page Background color.
Enabled Scroll Bar	Scroll Bar Background	Button Middle	
Disabled Scroll Bar	Field Background	Button Middle	The change to the Button Middle color, which is the scroll bar's slider color, will not be visible until the scroll bar becomes enabled. However, if you change the system default Button Middle color, it will affect all other enabled scroll bars, as well as other push buttons.
Title Bar of the window that has the focus	Active Title Bar and Active Title Bar Text Background	Active Title Bar Text	

continued

Table 6-8. Colors You Can Change Using The Color Palette, continued

Window	Background Color Changed	Foreground Color Changed	Notes
Title Bar of a window that does not have the focus	Inactive Title Bar and Inactive Title Bar Text Background	Inactive Title Bar Text	
Any other window	Window Background	Window Text	

Use nondithered colors for the foreground and background colors for any window element that contains text, such as the title bar. If you select dithered colors for either the text or the background, you may find that the text is unreadable. Remember that with a dithered color, the system tries to combine colors by alternating two different colored dots to create a specified color. What do you think happens if three or four different color dots are close together? They dither also, blurring the distinction between the text and the background. Even worse, the dithered pattern can use the same color for some of the background dots as the text color, making the text blend into the background even more.

It isn't always obvious when you are using dithered colors. For the text color, you are safe choosing black, white, red, green, or blue (the primary colors). For the background, you can usually tell if it is dithered by looking at the colored area closely. If it looks like all one color, it is probably nondithered. If it looks like it is made up of different colored dots, it is dithered. If your text looks blurry or parts of the text look like they are missing, you have probably selected a dithered background. Just change the color and see if it helps. Also, certain fonts are more prone to blending into the background. You may have to experiment with the text color, background color, and font until you get a combination that looks good to you.

Creating a custom color

If you don't see the color you want on the Color Palette, you can construct a custom color by following this procedure:

1 Open the Mixed Color Palette.

2 Select a color square to change. All the color squares in the top row are white, so you can change all but one of them without losing anything. Double-click on the color square you want to change. This will open the Edit Color dialog box (see Figure 6-11). Alternatively, you can select the color square you want to change by clicking on the color square, then select the **Edit color...** push button.

3 Use the color wheel (the big colored circle near the top of the Edit Color dialog box) to choose the basic color and use the slider bar (the vertical colored bar to the right of the color wheel) to choose the brightness. The actual color you select is in the sliding rectangle on the slider bar:

- Move the mouse pointer to the center of the crosshair within the color wheel, press and hold mouse button two, move the crosshair around on the color wheel until the color you want is visible in the slider bar, then release mouse button two.
- Move the mouse pointer to the sliding rectangle on the slider bar, press and hold mouse button two, move the slider up or down until the color you want is within the rectangle, then release mouse button two.
- Close the Edit Color dialog box and the color square you have chosen is changed to your custom color.

A second way to select the color is to use the **Red**, **Green**, and **Blue** or **Hue**, **Saturation**, and **Brightness** spin buttons at the bottom of the Edit Color dialog box. If you don't see the spin buttons, click on the **Values>>** push button.

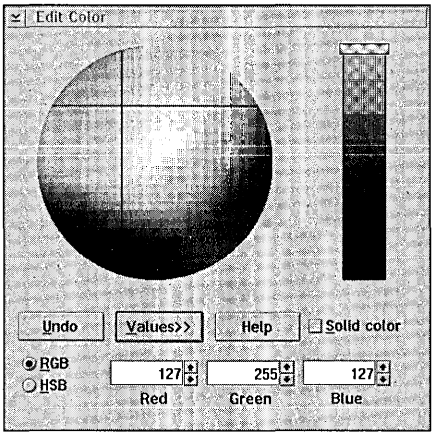
- If you select the **RGB** radio button, the spin buttons represent the proportion of Red, Green, and Blue colors that comprise the selected color. The amount of each color can be specified as a value between 0 and 255. For example, white is specified as Red = 255, Green = 255, and Blue = 255, pure red is Red = 255, Green = 0, Blue = 0, pure green is Red = 0, Green = 255, Blue = 0, and so on.
- If you select the **HSB** radio button, the spin buttons represent the Hue (a value between 0 and 360 representing the angle around the color circle, in degrees), Saturation (a value between 0 and 100 representing the distance from the center of the color circle, where 0 is the center of the circle and 100 is the outer edge of the circle), and the Brightness (a value between 0

and 100 representing the distance on the slider bar, where 0 is the bottom of the slider bar and 100 is the top of the slider bar). For example, white is specified as Hue = 0 (actually, any value will do), Saturation = 0, and Brightness = 100, pure red is Hue = 0, Saturation = 100, and Brightness = 100, pure green is Hue = 120, Saturation = 100, and Brightness = 100, and so on.

- If you select the **Solid color** check box, the Edit Color dialog box will let you select only solid (nondithered) colors. If you select this option, you will notice that fewer colors exist in the slider bar (because your color selection is more restricted) than if you turn off this option.

4 Close the Edit Color dialog box by double-clicking on the system menu (at the upper left corner of the dialog window).

Figure 6-11.
Edit Color Dialog Box



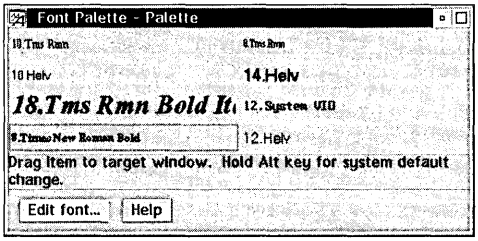
127	255	127
Red	Green	Blue

The Font Palette

The Font Palette, as shown in Figure 6-12, is used to specify fonts for window elements or to add or remove fonts from the system.

The Font Palette is also found in the System Setup folder.

Figure 6-12.
Font Palette



Drag item to target window. Hold Alt key for system default change.

To use the Font Palette, open the System Setup folder and double-click on the **Font Palette.** You then select the font you want, move the mouse pointer over the desired font name, press and hold mouse button two, drag the mouse pointer (which is now a pencil) to the window element for which you want to set the font, and release mouse button two. This will set the font of the window element.

Table 6-9 shows the fonts that you can change using the Font Palette. The table contains the following information:

- ✗ **Window** — This is the type of window you drop the font on (the window that is under the point of the pencil when you release mouse button two).
- ✗ **Font Changed** — This is the window element that is changed when you release the mouse button. These window elements are listed in Table 6-7. If you are holding **Alt** when you release mouse button two, the corresponding system default font is also changed.
- ✗ **Notes** — These are notes about the effects of dropping a font on particular windows.

Table 6-9. Fonts You Can Change Using The Font Palette

Window	Font Changed	Notes
Non-grid icon view of the Desktop	Desktop Icon Text	The system default Desktop Icon Text font is the font used for the text below any icon desktop.
Other view of the Desktop	Window Text	

continued

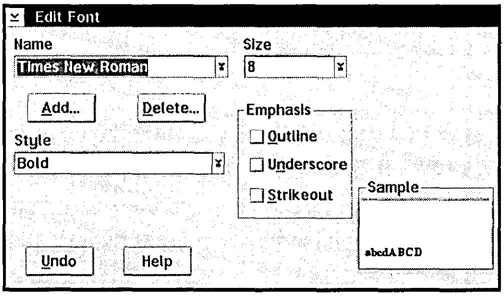
Table 6-9. Fonts You Can Change Using The Font Palette, continued

Window	Font Changed	Notes
Non-grid icon view of a Folder	Folder Icon Text	
Other view of a Folder	Window Text	
Menu	Menu Text	
Title Bar	Title Bar Text	Only one Title Bar Text font exists, so it doesn't matter whether the Title Bar you drop the font on has the focus or not.
Other Window	Window Text	

Creating a custom font

If you don't see the font you want on the Font Palette, you can select another font by following this procedure:

1. Open the Font Palette.
2. Select a font that you want to change by double-clicking on the name of the font. This will open the Edit Font dialog box (see Figure 6-13). Alternatively, you can select the name of the font by clicking on the name of the font, then selecting the **Edit font...** push button.
3. Select the name of the font you want from the Name list box. The Sample window shows a sample of the font. You then select the size of the font from the Size list box. You can choose the style of the font (for example, normal, bold, italics, bold-italics, or non-ISO) from the Style list box. You can also choose special characteristics (outline, underscore, or strikeout). You may notice that not all styles and characteristics are available for all fonts.
4. Close the Edit Font dialog box by double-clicking on the system menu. The font you chose is changed to your selected font.



Name: Times New Roman

Size: 8

Style: Bold

Figure 6-13. Edit Font Dialog Box

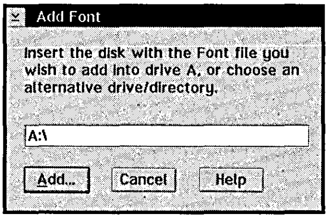
Adding new fonts

The Font Palette can be used to add new fonts to the system.

OS/2 Warp supports two different formats of fonts: bitmap fonts and scaleable fonts. The bitmap fonts supported by OS/2 are supplied in .FON files. The scaleable fonts supported by OS/2 must follow the standard for Adobe Type 1 fonts. These fonts require two files per font — a .PFB file (which contains the definition of the font itself) and an .AFM file (which contains the font metrics). Some vendors supply Adobe Type 1 fonts in Windows format, which requires the .PFB file and a .PFM file. If you have an Adobe Type 1 font that doesn't include the .AFM file, contact the vendor to get the .AFM file before you try to install the font.

When OS/2 installs an Adobe Type 1 font, it builds an .OFM file that contains the font metrics. You can also install an Adobe Type 1 font if the font definition diskette contains both the .PFB and the .OFM files for the font.

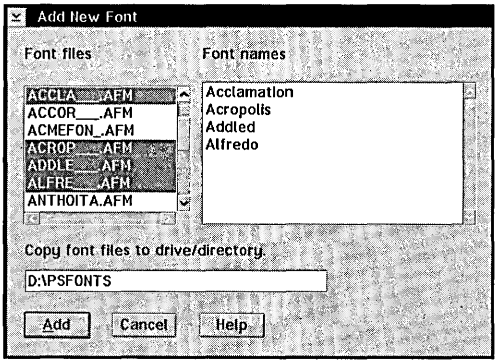
To add one or more new fonts to the system, open the Font Palette and open the Edit Font dialog box by clicking on the **Edit font...** push button. It doesn't matter which font you have selected, because that font won't be affected. Select the **Add...** push button. This displays the Add Font dialog box, as shown in Figure 6-14. Put the diskette containing the font definition into the A: drive or enter the path where the font definition files reside, then press the **Add...** push button. This presents the Add New Font dialog box (see Figure 6-15). Select the names of the fonts to be added in the Font files list box, then select the **Add** push button.



Insert the disk with the Font file you wish to add into drive A, or choose an alternative drive/directory.

A:\

Figure 6-14. Add Font Dialog Box



Font files

ACCLA___.AFM
ACCOR___.AFM
ACMEFON_.AFM
ACROP___.AFM
ADDLE___.AFM
ALFRE___.AFM
ANTHOITA.AFM

Font names

Acclamation
Acropolis
Addled
Alfredo

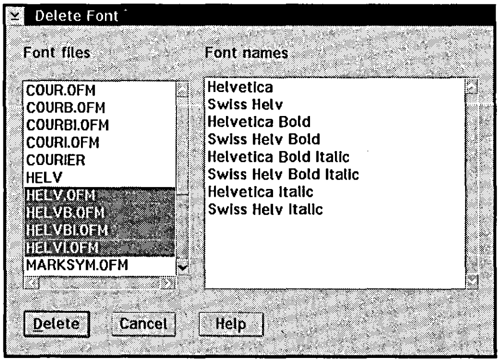
Copy font files to drive/directory.

D:\PSFONTS

Figure 6-15. Add New Font Dialog Box

Removing old fonts

The Font Palette can also be used to remove fonts that you no longer need. To remove fonts from the system, open the Font Palette and open the Edit Font dialog box by clicking on the **Edit font...** push button. It doesn't make any difference which font is selected, because that font won't be affected. Select the **Delete...** push button. This displays the Delete Font dialog box, as shown in Figure 6-16. Select the names of the fonts to be removed in the Font files list box, then press the **Delete** push button.



Font files

COUR.OFM
COURB.OFM
COURBI.OFM
COURI.OFM
COURIER
HELV
HELV.OFM
HELVB.OFM
HELVBI.OFM
HELVI.OFM
MARKSYM.OFM

Font names

Helvetica
Swiss Helv
Helvetica Bold
Swiss Helv Bold
Helvetica Bold Italic
Swiss Helv Bold Italic
Helvetica Italic
Swiss Helv Italic

Figure 6-16. Delete Font Dialog Box

The Folder Settings Notebook

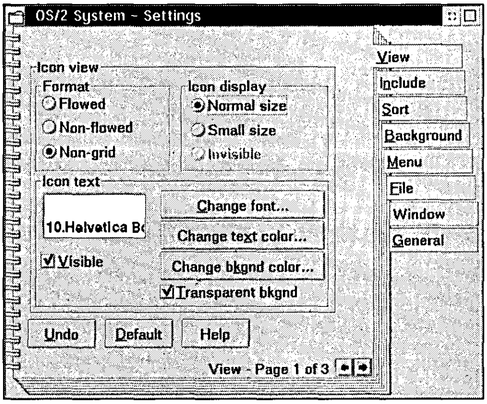
Look in the Folder Settings Notebook (including the Desktop) for the pages that affect the folder views' colors, fonts, and background.

The first three pages in the Settings Notebook for a folder are the View pages 1 through 3.

Icon view page

The first View page, as shown in Figure 6-17, is for Icon view. The radio buttons in the Format and Icon display groups allow you to select the way the information is presented when an Icon view of the folder is open.

Figure 6-17.
Folder Object — Icon View Page



OS/2 System - Settings

Icon view

Format
○ Flowed
○ Non-flowed
● Non-grid

Icon display
● Normal size
○ Small size
○ Invisible

Icon text
10.Helvetica B
☑ Visible
☑ Transparent bkgnd

View - Page 1 of 3

- ✗ **Flowed** shows the contents of the folder in a series of columns, with the icon for the object followed by the object's title. In Flowed Icon view, you will always see a horizontal scroll bar at the bottom of the folder. You can choose to display the icons for each object in **Normal** size (that is, use the 32 x 32 or 40 x 40 icon), **Small** size (that is, use the 16 x 16 or 20 x 20 icon), or **Invisible** (that is, don't display the icon, only display the object titles). If you choose invisible icons, you see just columns of names of objects.

- ✗ **Non-flowed** shows the contents of the folder in a single column, with the icon for the object followed by the object's title. In Non-flowed Icon view, you will always see a vertical scroll bar at the right side of the folder. You can choose to display the icons for each object in **Normal** size (that is, use the 32 x 32 or 40 x 40 icon), **Small** size (that is, use the 16 x 16 or 20 x 20 icon), or **Invisible** (that is, don't display the icon, only display the object titles). If you choose invisible icons, you see just a single column of names of objects.

- **Non-grid** shows the contents of the folder with the icons initially arranged in horizontal rows, with the object's title centered below the icon. In Non-grid Icon view, you can move the icons around to any arrangement you want. You can ask the Workplace Shell to rearrange the icons into rows by selecting **Arrange** from the folder's pop-up menu. You can also ask the Workplace Shell to sort the icons by selecting the **Sort** option from the folder's pop-up menu. You can choose to display the icons for each object in **Normal** size (that is, use the 32 x 32 or 40 x 40 icon) or **Small** size (that is, use the 16 x 16 or 20 x 20 icon).

The options within the Icon text group allow you to specify the colors, fonts, and text visibility used to display the object titles in the various modes of Icon view. You can specify the font for the text, the color of the text, the color of the background for the text, and whether or not to display the text.

Uncensored

There are three different sets of fonts, colors, and text visibility options maintained for Icon view:

- Non-grid Icon view
- Flowed Icon view or Non-flowed Icon view with Normal or Small icons
- Flowed Icon view or Non-flowed Icon view with Invisible icons

Warning

The options within the Icon text group apply only to the selected Icon view of the folder. For example, if the Non-grid option is selected, the colors and font you specify will be seen only if you open the folder in Non-grid Icon view. If you set the colors and fonts, then change the option to Flowed and open the folder, you will see the colors and font associated with Flowed Icon view, not the colors and font you just set.

There are two ways of specifying the color and font options:

1 Use the Edit dialog boxes:

- Select the **Change font...** push button to obtain the Edit Font dialog box (see Figure 6-13), where you can specify the font to be used for the text. See the Font Palette section earlier in this chapter for a detailed explanation on the use of the Edit Font dialog box.
- Select the **Change text color...** push button to obtain the Edit Color dialog box (see Figure 6-11), where you can specify the color to be used for the text.

- Select the **Change bkgnd color...** push button to obtain the Edit Color dialog box, where you can specify the color to be displayed behind the text. See the Color Palette section earlier in this chapter for a detailed explanation on the use of the Edit Color dialog box. If you select the **Transparent bkgnd** check box, the folder background color (either a simple color or the color of the background bitmap) will be the background color for the text. If you turn off the **Transparent bkgnd** check box, the object's title will be surrounded by a rectangle of the specified background color. Note that if you specify a background text color, the **Transparent bkgnd** check box is automatically turned off. We assumed that if you went to the trouble of specifying a background text color, you wanted to see it.

Tip

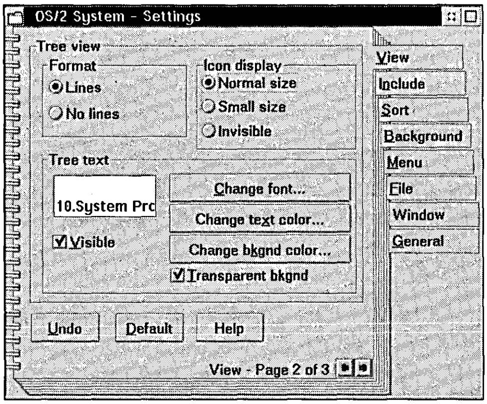
You would normally specify the icon text background color as *transparent* if you assign a simple color to the folder background (see the Background Page section later in this chapter). You would normally specify an icon text background color if you assign a bitmap to the folder background and the icon text is hard to read. If you turn off the **Visible** check box, the object's title will not be displayed — you will see only the icons. (You cannot turn off the **Visible** check box if **Invisible** icons are selected — you must display either the icons or the text or both.)

2 You can also use the Preview window to change the icon text colors and font. You will notice that as you change the font or colors using the Edit Font dialog box or Edit Color dialog box, the Preview window (and the folder itself, if open in Icon view) changes. Instead of using the Edit Font dialog box and Edit Color dialog box, you can use the Font Palette and Color Palette to change the font or colors by dragging and dropping a font or a color on the Preview window. For example, you can open the Font Palette, select a font, and drop it into the preview window. This font is then used to display the object titles. Similarly, you can drop a color into the Preview window, and it will be used as the background text color. If you drop a color into the Preview window, the **Transparent bkgnd** check box is automatically turned off. If you hold **Ctrl** while dropping a color into the Preview window, the color will be used as the text color.

Even though three sets of fonts, text colors, and text visibility options are maintained for the Icon view of a folder, only a single text background color is shared by *all* views of the folder. If you change the text background color on the Icon view page, you will see the same color (including *transparent*) used by Tree view and Details view.

Tree view page

The second View page, as shown in Figure 6-18, is for Tree view. The radio buttons in the Format and Icon display groups allow you to select the way the information is presented when a Tree view of the folder is open.



OS/2 System - Settings

Tree view

Format
● Lines
○ No lines

Icon display
● Normal size
○ Small size
○ Invisible

Tree text
10.System Prc
☑ Visible
☑ Transparent bkgnd

View - Page 2 of 3

Figure 6-18. Folder Object – Tree View Page

- ✗ **Lines** shows the lines outlining the hierarchy of subfolders within the folder. You can choose to display the icons for each object in **Normal** size (that is, use the 32 x 32 or 40 x 40 icon), **Small** size (that is, use the 16 x 16 or 20 x 20 icon), or **Invisible** (that is, don't display the icon, only display the object titles). If you choose invisible icons, you see just the hierarchy of names of objects.
- ✗ **No lines** just displays the icons and titles of the subfolders within the folder in a hierarchical fashion. You can choose to display the icons for each object in **Normal** size (that is, use the 32 x 32 or 40 x 40 icon), **Small** size (that is, use the 16 x 16 or 20 x 20 icon), or **Invisible** (that is, don't display the icon, only display the object titles). If you choose invisible icons, you see just the hierarchy of names of objects.

The options within the **Tree text** group allow you to specify the colors, fonts, and text visibility used to display the object titles in the various modes of Tree view. You can specify the font for the text, the color of the text, the color of the background for the text, and whether or not to display the text.

There are two different sets of fonts, colors, and text visibility options maintained for Tree view:

- ✗ Tree view with Normal or Small icons
- ✗ Tree view with Invisible icons

Warning

The options within the Tree text group apply only to the selected Tree view of the folder. For example, if the Normal icon option is selected, the colors and font you specify will be seen only if you open the folder in Tree view with Normal or Small icons. If you set the colors and fonts, then change the option to Invisible icons and open the folder, you will see the colors and font associated with Tree view with invisible icons, not the colors and font you just set.

There are two ways of specifying the color and font options:

1 Use the Edit dialog boxes:

   - Select the **Change font...** push button to obtain the Edit Font dialog box (see Figure 6-13), where you can specify the font to be used for the text.
   - Select the **Change text color...** push button to obtain the Edit Color dialog box (see Figure 6-11), where you can specify the color to be used for the text.
   - Select the **Change bkgnd color...** push button to obtain the Edit Color dialog box, where you can specify the color to be displayed behind the text.

   If you select the **Transparent bkgnd** check box, the folder background color (either a simple color or the color of the background bitmap) will be the background color for the text. If you turn off the **Transparent bkgnd** check box, the object's title will be surrounded by a rectangle of the specified background color. If you specify a background text color, the **Transparent bkgnd** check box is automatically turned off.

Tip

   You would normally specify the Tree text background color as transparent if you assign a simple color to the folder background (see the Background Page section later in this chapter). You would normally specify a Tree text background color if you assign a bitmap to the folder background and the text is hard to read. If you turn off the **Visible** check box, the object's title will not be displayed — you will see only the icons.

2 Use the Preview window. You will notice that as you change the font or colors using the Edit Font dialog box or Edit Color dialog box, the Preview window (and the folder itself, if open in Tree view) changes. You can use the Font Palette and Color Palette to change the font or colors by dragging and dropping a font or a color on the Preview window.

For example, you can open the Font Palette, select a font, and drop it into the Preview window. This font is then used to display the object titles. Similarly, you can drop a color into the Preview window, and it will be used as the background text color. If you drop a color into the Preview window, the **Transparent bkgnd** check box is automatically turned off. If you hold **Ctrl** while dropping a color into the Preview window, the color will be used as the text color.

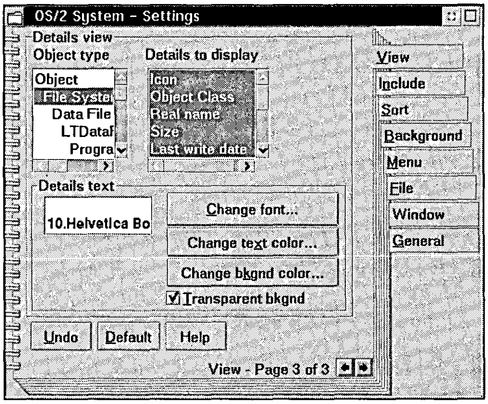
Warning

Even though two sets of fonts, text colors, and text visibility options are maintained for the Tree view of a folder, only a single text background color is shared by all views of the folder. If you change the text background color on the Tree view page, you will see the same color (including *transparent*) used by Icon view and Details view.

Details view page

The third View page, shown in Figure 6-19, is for Details view. The list boxes at the top of the page allow you to select the information that is presented when a Details view of the folder is open.

Figure 6-19.
Folder Object — Details View Page



Object type: Object, File Syste, Data File, LTDataF, Progra

Details to display: Icon, Object Class, Real name, Size, Last write date

10.Helvetica Bo

☑ Transparent bkgnd

Object type specifies the predominant class of the objects within the folder. The specified object class controls what information is available to be displayed. For example, if **File System** is selected, most of the objects within the folder are File System objects (normally other folders or data files), and the File System object class specifies the data that is displayed.

Details to display allows you to select which data will be displayed. The default is that all data supplied by the specified object class is displayed. You can deselect some or all of the data, limiting the number of columns displayed.

The options within the Details text group allow you to specify the colors and font used to display the data in Details view. You can specify the font for the text, the color of the text, and the color of the background for the text.

There are two ways of specifying the color and font options:

1 Use the Edit dialog boxes:

- Select the **Change font...** push button to obtain the Edit Font dialog box (see Figure 6-13), where you can specify the font to be used for the text.
- Select the **Change text color...** push button to obtain the Edit Color dialog box (see Figure 6-11), where you can specify the color to be used for the text.
- Select the **Change bkgnd color...** push button to obtain the Edit Color dialog box, where you can specify the color to be displayed behind the text.

  If you select the **Transparent bkgnd** check box, the folder background color (either a simple color or the color of the background bitmap) will be the background color for the text. If you turn off the **Transparent bkgnd** check box, the object's title will be surrounded by a rectangle of the specified background color. If you specify a background text color, the **Transparent bkgnd** check box is automatically turned off.

  In this case, if you assigned a background bitmap to the folder, the bitmap will be seen only in the area below the information for the last object contained within the folder. You would normally specify the details text background color as transparent if you assign a simple color to the folder background (see the Background Page section later in this chapter). You would normally specify a details text background color if you assign a bitmap to the folder background and the details text is hard to read.

2 Use the Preview window. You will notice that as you change the font or colors using the Edit Font dialog box or Edit Color dialog box, the Preview window (and the folder itself, if open in Tree view) changes. You can use the Font Palette and Color Palette to change the font or colors by dragging and dropping a font or a color on the Preview window.

For example, you can open the Font Palette, select a font, and drop it into the Preview window. This font is then used to display the details data. Similarly, you can drop a color into the Preview window, and it will be used as the background text color. If you drop a color into the preview window, the **Transparent bkgnd** check box is automatically turned off. If you hold **Ctrl** while dropping a color into the Preview window, the color will be used as the text color.

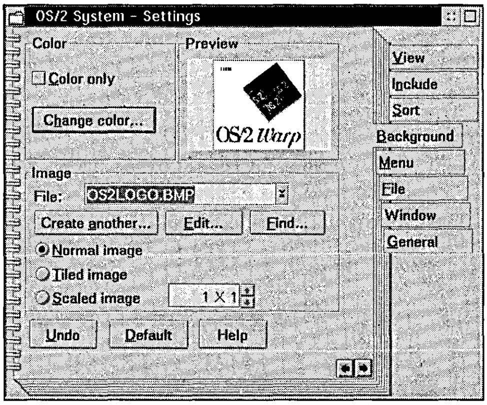
Warning

A single text background color is shared by all views of the folder. If you change the text background color on the Details view page, you will see the same color (including transparent) used by Icon view and Tree view.

Background page

The Background Page (see Figure 6-20) is used to specify the background for the folder. This background is shared by *all* views of the folder. Because the Desktop is a folder, you can specify the background for the Desktop through the Settings Notebook for the Desktop Folder.

Figure 6-20.
Folder Object — Background Page



OS/2 System - Settings

□ Color only

File: OS2LOGO.BMP

● Normal image
○ Tiled image
○ Scaled image 1 X 1

The folder background can be a simple color, a bitmap that is centered within the open folder view and surrounded by the background color, or a bitmap (or series of bitmaps) that covers the entire open folder view window.

The Preview window shows a sample of what the folder view background will look like. The Preview window is not an exact picture of what you will see when you open a folder view. If you specify a background bitmap, the bitmap may not be displayed in the same proportion within the Preview window as in the actual open view window. Also, if the **Workplace shell palette aware** option is selected, the Workplace Shell will display the background bitmap with the correct color set when displaying the open folder view; however, it uses the default color set in the Preview window.

The Color group is used to specify the background color. If the **Color only** check box is selected, the entire folder view window is filled with the specified color, and the information in the Image group is ignored. You can select the **Change color...** push button to obtain the Edit Color dialog box to specify the background color. Or you can open the Color Palette and drop a color into the Preview window to specify the background color.

If the **Color only** check box is not selected, the background bitmap may fill the entire Preview window. If this is the case, you will not see the background color, even if you drop a new color into the Preview window from the Color Palette. Don't worry — the system remembers the background color even if it can't display it.

The Image group is used to specify a background bitmap and its characteristics. This group is enabled only if the **Color only** check box is turned off. You can select the name of the bitmap to be displayed from the File combination box. This box lists all of the bitmaps that reside in the \OS2\BITMAP directory on the boot drive. There are two special entries in the list:

- (none) — This option specifies that no bitmap is to be displayed. It is equivalent to selecting the **Color only** check box.
- *OS2LOGO — This option specifies that the OS/2 logo be drawn as the background bitmap. It does not require a bitmap file. However, because the logo is drawn (instead of loaded from a file), it takes much longer to build the background using this option.

Tip

If you want to use this option, I suggest using it sparingly — you may use it for the Desktop (where it is drawn only once), but don't use it for a lot of folders.

The push buttons below the File combination box allow for alternate methods to specify the background bitmap. The **Create another...** push button invokes the Icon Editor to create a new bitmap.

Warning

Two problems occur when using this option. First, the Icon Editor can edit only a relatively small bitmap, so you can't create a bitmap that covers the screen with this option, unless it is tiled or scaled. The second problem is that when you save the new bitmap, its name is not added to the File list until you close and reopen the Settings Notebook.

The **Edit...** push button invokes the Icon Editor to edit the bitmap specified by the File option. Again, because the Icon Editor can edit only small bitmaps, it may not be able to edit the specified bitmap. The **Find...** push button displays the **Find objects** dialog box to allow you to find a bitmap file anywhere on your system, not just in the \OS2\BITMAP directory. When the find results are displayed, you just have to select the bitmap you want to use and select the **OK** push button. The selected bitmap is automatically added to the File list and is made the background bitmap.

An alternate method to specify a background bitmap is to open a folder containing the bitmap file you want to use, drag the bitmap file into the Preview window, and drop it there.

Warning

The background bitmap must be contained within an OS/2 or Windows Bitmap (.BMP) File. The Workplace Shell doesn't support any other image files, such as GIF, JPEG, or TIFF files, as background bitmaps.

The radio buttons below the **Create another...**, **Edit...**, and **Find...** push buttons allow you to specify the characteristics of the bitmap.

- ✗ The **Normal** option specifies that the bitmap is to be displayed at its original size, centered within the folder view window and surrounded by the background color. If the bitmap is larger than the open folder view window, the bitmap is scaled down (in its original proportions) to fit the window.

✗ The **Tiled** option specifies that the bitmap is replicated like tiles on a tiled floor to fill up the window. The number of rows of tiles may be different from the number of columns, depending on the horizontal and vertical dimensions of the bitmap being tiled and the size of the window. Each copy of the bitmap is the size of the original bitmap. If the replicated bitmaps are larger than the window, the right column and bottom row of the replicated bitmaps are cut off.

✗ The **Scaled** option specifies that the bitmap is to be replicated the specified number of times horizontally and vertically, where the number of copies is the value in the spin button that follows the **Scaled** option. The bitmap is scaled so it exactly fills the window after it is replicated. For example, if 1 x 1 is specified, the bitmap is scaled to fit the window. If 3 x 3 is specified, the bitmap is scaled so exactly three copies fill the window horizontally and three copies fill the window vertically.

Because the bitmap may be scaled differently horizontally and vertically, it may be displayed out of proportion.

Tip

The bitmap file must be present when the folder view is opened, so don't specify a bitmap that resides on a removable volume, such as a diskette, CD-ROM, or network-attached drive. If you want to use a bitmap file that is on a removable volume, copy it to your fixed disk first. For example, you may copy a bitmap from a CD-ROM to the \OS2\BITMAP directory, then specify the bitmap in the \OS2\BITMAP directory as the background bitmap.

Uncensored

You may notice that if you open different views of the folder (like Icon view and Tree view) that have different size windows, the background bitmap can be displayed differently in each window. We maintain a different image of the bitmap for each open view window so it can be independently tiled or scaled to fit each window properly. Also, in Details view, two independent panes make up the Details view window. The background bitmap is displayed in each pane, but it is tiled or scaled to fit the entire window.

Tip

If you are using a 256-color bitmap for your background, especially for your Desktop background, you may want to turn on the Workplace shell palette aware option so the Workplace Shell will use the Palette Manager to display the bitmap with the correct colors. However, because there are side effects to using the Palette Manager, you will have to decide for yourself whether the system looks better with the option turned on or off.

Developer's Desktop

For example, on my own system, I have a 256-color bitmap on my Desktop. I have the **Workplace shell palette** aware option turned on so the Desktop bitmap looks correct. When I run another palette-managed application program, like the Windows version of the Prodigy software, the background bitmap looks correct, even though Prodigy changes the color set.

Tip

If you are using a simple color for the folder background, you would normally specify the background text color as transparent for the views you normally open. If you are using a bitmap for the folder background, it may be hard to read the icon text in some areas of the screen, where the text color and the background bitmap are using the same or similar colors. In this case, you would specify a background text color that contrasts well with the text color. You may want to have the background text color contrast with the background bitmap's color scheme so the text stands out, or you can select a background text color that blends in with the background bitmap in most areas, allowing the text to be read but not to "jump out" at you. The choice is yours — it depends on how you want your folder views to appear.

The Scheme Palette

The Scheme Palette (see Figure 6-21) can be used to specify all of the color, font, and background information for the Desktop or a folder in a single operation. It can also specify that the system default values for colors, fonts, and folder backgrounds be changed.

Uncensored

If you installed OS/2 Warp over a previous version of OS/2, you will see the **old** Scheme Palette. We don't automatically replace the old Scheme Palette with the new Scheme Palette because we don't know if you customized the schemes and want to retain them. If you want to create a new Scheme Palette and see the new schemes defined for OS/2 Warp, you can run the SCHEME command, which renames your old Scheme Palette to Old Scheme Palette and creates a new Scheme Palette in the System Setup folder. It also adds the **delete** option to the Old Scheme Palette object so you can delete it if you wish. The SCHEME command file is found in the \OS2\INSTALL directory for OS/2 Warp Version 3 with WIN-OS/2.

CD-ROM

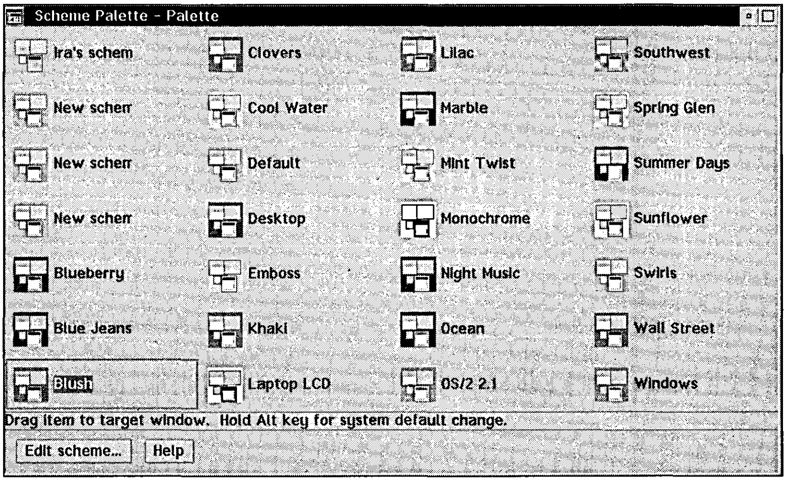
If your system is OS/2 Warp Version 3, you can find the SCHEME.CMD file on the CD-ROM included with this book.

To use the Scheme Palette, open the System Setup folder, and then open the Scheme Palette. You are presented with a set of 28 schemes that you can use. The first four schemes are labeled New scheme. They are there for you to customize. I will describe customizing your own schemes a little later. The other 24 schemes contain predefined sets of colors, fonts, and Desktop backgrounds that we have provided:

- **OS/2 2.1** gives you the same colors that OS/2 Version 2.1 and OS/2 Version 2.11 used as the default color set.
- **Windows** gives you the same colors that Microsoft Windows 3.1 uses.
- **Default** gives you the set of colors that OS/2 Warp Version 3 uses if you don't change them.

The names of the other schemes were chosen to suggest the color theme used. For example, **Blue Jeans** is predominantly blue, and the Desktop background looks like blue denim; **Clovers** is mainly green, and the Desktop background looks like a pine forest; and **Sunflower** has a yellow theme, and the Desktop background is a group of sunflowers.

Figure 6-21. Scheme Palette



Drag item to target window. Hold Alt key for system default change.

To use a scheme, you just have to drag its icon to an open window you want to change and drop it on the window. The window elements of that window will then change to the values contained within the scheme. The scheme can also be used to set the system default values by holding **Alt** while dropping the scheme on a window.

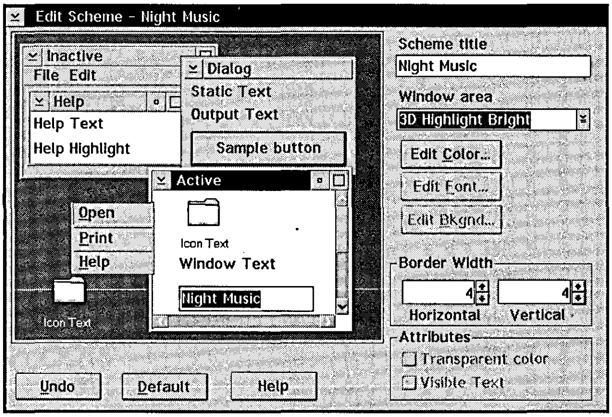
To see what effect the predefined schemes will have on your system, you can drag the schemes to the Desktop. If you like the Desktop colors, you can then hold **Alt** and drag the scheme to the Desktop again. This will change the system default values to match the scheme. Many of the predefined schemes have Desktop bitmaps defined. Others have Desktop background colors and active title bar colors that match a particular theme.

Creating a custom scheme

You may not like any of the schemes we have provided, so you can create your own scheme.

You can edit any of the schemes in the Scheme Palette, but I suggest that you customize only the schemes whose names are New scheme so you still have the predefined schemes if you ever want to use them.

To create a customized scheme, open the Scheme Palette and double-click on the icon for the scheme that you want to change. You are then presented with the Edit Scheme dialog box, as seen in Figure 6-22. If you are customizing one of the New schemes, the first thing you should do is change its name by typing a new name in the Scheme title entry field. That way, you will remember what scheme you customized and for what purpose.


Edit Scheme - Night Music

Scheme title: Night Music

Window area: 3D Highlight Bright

Border Width: Horizontal 4, Vertical 4

Attributes:
□ Transparent color
□ Visible Text

Figure 6-22. Edit Scheme Dialog Box

The main area within the Edit Scheme dialog box is a picture of many window elements: active and inactive windows, the Desktop and a folder containing icons, a menu, and so on. These elements show you the colors that are currently defined for the scheme. However, if the scheme contains a background bitmap for either the Desktop or a folder, the background bitmap is not shown — only the background color is displayed.

To change any window element, you can move the mouse pointer to any part of the picture and press mouse button two. This will give you a menu containing the list of window elements represented by that part of the picture. You can then select one of the window elements from the menu (see Table 6-10 for a complete list of the window elements):

- If the element represents the color of a window element, the Edit Color dialog box will magically appear, allowing you to select the color for that element.
- If the element represents a text font, the Edit Font dialog box will appear, allowing you to specify the font.
- If the element represents the background for the Desktop or a folder, the Edit Background dialog box will appear, allowing you to select the color and background bitmap, along with the background bitmap's characteristics.

Tip

The Scheme Palette maintains two sets of color values for folders — one for the Desktop and a second set for all other folders. The values that are maintained separately are the folder background color and bitmap (and its characteristics), the icon text color, the icon text background color, and the icon text visibility option. All other folder characteristics, including the icon text font, are shared between the Desktop and all other folders.

Warning

The Icon text font and the Shadow Icon text font are the same font.

You can also directly select the window element to change by selecting one of the entries from the **Window Area** combination box. Table 6-10 is a list of the window elements that you can select. The columns in the table show the following information:

- **Window Area** — The name of the window element in the Window Area list box.
- C — You can use the **Edit Color...** push button to edit the color.

- F — You can use the **Edit Font...** push button to edit the font.
- B — You can use the **Edit Bkgnd...** push button to edit the background.
- T — You can use the **Transparent Color** check box to set the color to transparent.
- V — You can use the **Visible Text** check box to make the text invisible.
- **Element That Is Set** — The name of the window element (from Tables 6-1 through 6-7) that you can set using this Window Area name.

Table 6-10. Window Areas In The Scheme Palette

Window Area	C	F	B	T	V	Window Element That Is Set
3D Highlight Bright	✓					Button Light Color
3D Highlight Dark	✓					Button Dark Color
Active Border	✓					Window Border Color – focus window
Active Title Background	✓					Title Bar Color – focus window
Active Title Text	✓	✓				Title Bar Text Color – focus window Title Bar Text Font – all windows
Active Title Text Background	✓					Title Bar Text Background Color – focus window
Application Workspace	✓					Application Workspace Color
Desktop Background			✓			Folder Background Color – Desktop folder Folder Background Bitmap – Desktop folder Presentation Manager Desktop Background Color (if **Alt** is held while dropping this scheme on a window)
Desktop Icon Text	✓				✓	Icon Text Color – Desktop folder
		✓				Icon Text Font – Desktop folder

Window Area	C	F	B	T	V	Window Element That Is Set
Desktop Icon Text Background	✓			✓		Icon Text Background Color – Desktop folder
Dialog Background	✓					Dialog Background Color
Drop Shadow	✓					Shadow Color
Entryfield/Listbox ground	✓					Entry Field and List Box Background Back-Color
FolderBackground			✓			Folder Background Color – non-Desktop folders Folder Background Bitmap – non-Desktop folders
Folder Icon Text	✓				✓	Icon Text Background Color – non-Desktop folders
		✓				Icon Text Font – non-Desktop folders
Folder Icon Text Background	✓			✓		Icon Text Background Color – non-Desktop folders
Frame	✓					Window Frame Color
Help Background	✓					Help Background Color
Help Highlight Text	✓					Help Highlight Color
Help Text	✓					Help Text Color
Inactive Area	✓					Field Background Color
Inactive Border	✓					Window Border Color – non-focus windows
Inactive Title Background	✓					Title Bar Background Color – non-focus windows
Inactive Title Text	✓					Title Bar Text Color – non-focus windows
		✓				Title Bar Text Font – all windows
Inactive Title Text Background	✓					Title Bar Text Background Color – non-focus windows
Menu Background	✓					Menu Background Color
Menu Disabled Text	✓					Menu Disabled Text Color
		✓				Menu Font

continued

Table 6-10. Window Areas In The Scheme Palette, continued

Window Area	C	F	B	T	V	Window Element That Is Set
Menu Highlight Background	✓					Menu Highlight Background Color
Menu Highlight Text	✓	 ✓				Menu Highlight Text Color Menu Font
Menu Text	✓	 ✓				Menu Text Color Menu Font
Notebook Page Background	✓					Page Background Color
Output Text	✓					Output Text Color
Pushbutton Background	✓					Button Middle Color
Pushbutton Default Border	✓					Button Default Color
Scroll Bar Background	✓					Scroll Bar Background Color
Selection Background	✓					Selection Highlight Background Color
Selection Foreground	✓					Selection Highlight Foreground Color
Shadow Highlight Background	✓					Shadow Background Highlight Color
Shadow Highlight Text	✓					Shadow Foreground Highlight Color
Shadow Text	✓					Shadow Text Color
Static Text	✓					Window Static Text Color
Title Bottom	✓					Title Bar Bottom Color
Window Text	✓	 ✓				Window Text Color Window Text Font

Once you have selected a window element, you may use the following buttons to change the selected window element:

- ✗ **Edit Color...** push button — Displays the Edit Color dialog box (see Figure 6-11) to edit the color.

- **Edit Font...** push button — Displays the Edit Font dialog box (see Figure 6-13) to edit the font.
- **Edit Bkgnd...** push button — Displays the Edit Background dialog box to edit the folder background. The Edit Background dialog box looks exactly like the Folder Background Page (see Figure 6-20) except that it has a title bar.
- **Transparent Color** check box — Sets the selected color to transparent. This allows you to define an Icon Text Background Color that allows the Folder Background Color or Bitmap to show around the Icon Text.
- **Visible Text** check box — Makes the selected text visible or invisible. If you turn off this text box, the icons are displayed without the titles of the objects they represent.

There are two other window characteristics that you can set using the Scheme Palette. These are set by changing the values in the spin buttons in the Border Width group.

- **Horizontal** — This is the width (in pixels) of the left and right sizing borders of the window.
- **Vertical** — This is the height (in pixels) of the top and bottom sizing borders of the window.

Summary

You now know how to change the colors, fonts, and folder backgrounds to make the Workplace Shell look the way you want it to. If you like, you can make each of your folders a different color so you can have a rainbow on your screen when you are really busy.

Developer's Desktop

On my system at work, where I usually have a lot of windows open all the time, I use the POOL.BMP background on my Desktop, because it is a soft, nonintrusive pattern, black for my Desktop icon text, and a shade of blue that blends in with the pool for the icon text background color. At home, where I often have only one or two windows open at a time, I use a photograph (if you must know, a color photograph of the cast of “Star Trek: The Next Generation”) for my Desktop background, black for the Desktop icon text, and a bright cyan for the icon text background color. This scheme gives me something interesting to look at when the system is idle.

Tip

You can personalize your system even more by taking a photograph of yourself, having it converted to a bitmap (by using a scanner), and using the bitmap as your Desktop background or the background of a folder you use often. The possibilities are only limited by your imagination.

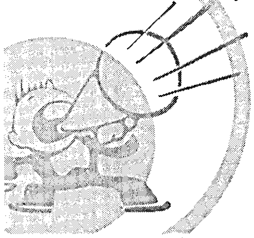
Ira H. Schneider is an Advisory Programmer on the Workplace Shell Development Team. He has been instrumental in developing OS/2 Warp, including performance improvements and functional enhancements to the Workplace Shell. He has also worked on the OS/2 Swapper and on OS/2 System Initialization, and he was Lead Architect for OS/2 1.2. Ira has been with IBM since 1969.

Chapter 7

Customizing the Workplace Shell

Look! A Path Just for Us!

There are many behaviors of the Workplace Shell that you can customize to your own liking. They include the operation of the mouse and keyboard, the resolution of your display, common settings values, sounds, the operation of the print spooler, the default behavior of WIN-OS/2 sessions, country information, and the appearance of the system clock. In this chapter, I will introduce you to the various facilities of the Workplace Shell you can use to customize the system so that it works the way you want.



The Mouse Object

The Mouse object in the System Setup folder can be used to specify the characteristics of your pointing device and the mouse pointer. In this section, I will refer to your pointing device as a "mouse," even though you could be using a trackball or other pointing device instead. These characteristics include

- **Timing** — How fast the mouse pointer moves when you move the mouse and how long it takes the system to determine if you have double-clicked either of the mouse buttons
- **Left-handed versus right-handed operation** of the mouse buttons
- **Mapping** — Which mouse buttons and keyboard keys map to which functions
- **Pointers** — The appearance of the most common mouse pointers
- **Comet Cursor** — Whether or not the mouse pointer leaves a trail when it moves

These characteristics can be specified through the Settings Notebook of the Mouse object. To open the Settings Notebook, just double-click on the Mouse object in the System Setup folder.

Timing page

The Timing page allows you to specify the double-click speed and tracking speed for the mouse (see Figure 7-1).

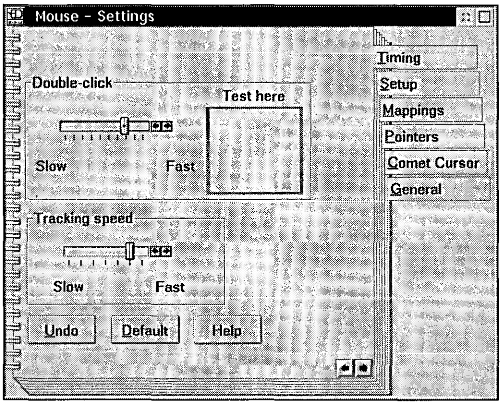


Figure 7-1. Mouse Object — Timing Page

Double-click

The Double-click group is used to specify how much time the system allows between clicks of the mouse in order to recognize a double-click.

- ✗ If you often find that you are clicking the mouse twice and the system is not recognizing this as a double-click, you might have the double-click time set too fast. Move the slider bar to the left to allow more time between clicks for the system to recognize a double-click.
- ✗ If, on the other hand, you find that you are accidentally double-clicking on objects when you intend to make two single clicks, move the slider bar to the right to shorten the maximum time between the two clicks that constitute a double-click.

Shhh! Tip

After adjusting the slider bar, you can see the effect of your change by double-clicking the mouse on the **Test here** window. If the system recognizes your double-click, it changes the color of the **Test here** window. You can test various timings and adjust the slider bar until you are comfortable with the change.

Tracking speed

The Tracking speed group is used to specify how responsive the mouse pointer is to movement of the mouse.

- ✗ If the mouse pointer doesn't move as quickly as you would like when you move the mouse, move the slider bar to the right. This will increase the speed at which the mouse pointer moves in response to movements of the mouse.
- ✗ If you find the mouse pointer racing across the screen too fast or if you can't control the exact positioning of the mouse pointer with small movements of the mouse, move the slider bar to the left. This will decrease the speed at which the mouse pointer moves in response to movements of the mouse.

Shhh! Tip

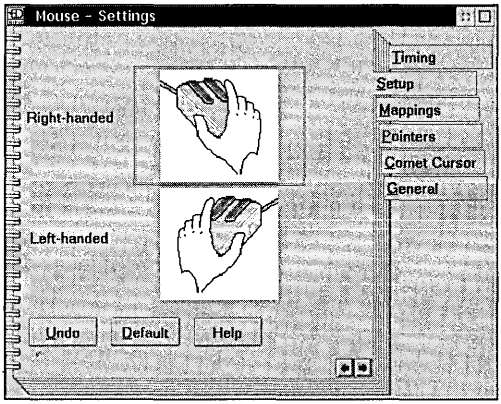
You can test the effect of your changes immediately by moving the mouse across the screen and seeing how responsive it is. With some pointing devices, the amount of movement of the mouse pointer varies with the speed at which you move the mouse. For example, if you move the mouse at a slow speed, the mouse pointer will move very little, but if you move the mouse faster, the mouse pointer may move farther for the same movement of the mouse. You will have to experiment with the tracking speed adjustment until you get a comfortable range of mouse pointer movement for the various speeds at which you move the mouse.

The double-click speed and tracking speed that you can adjust with the Mouse object affect the mouse only when the mouse is within an OS/2 window — they have no effect when the mouse is inside a WIN-OS/2 window. In order to change your mouse settings within a WIN-OS/2 window, you have to open the Windows Control Panel and select **Mouse Settings.** In the Mouse Settings dialog box, you can adjust the Mouse Tracking Speed and Double Click Speed to set your mouse preferences for WIN-OS/2 sessions.

Setup page

The Setup page allows you to specify whether you prefer the mouse to be treated as a right-handed or left-handed mouse (see Figure 7-2).

Figure 7-2.
Mouse Object — Setup Page



- ✘ If you select **Right-handed** operation, the left mouse button is referred to as mouse button 1 and the right mouse button is referred to as mouse button 2.
- ✘ If you select **Left-handed** operation, the right mouse button is referred to as mouse button 1 and the left mouse button is referred to as mouse button 2.

What does this mean to you? If you are right handed, you normally keep the mouse on the right side of your keyboard and move the mouse with your right hand. In this case, you probably operate the left mouse button with your thumb or index finger and the right mouse button with your index or third finger. Since the Workplace Shell uses mouse button 1 to select and open objects, it seems natural to most people to use their thumbs or index fingers for this operation.

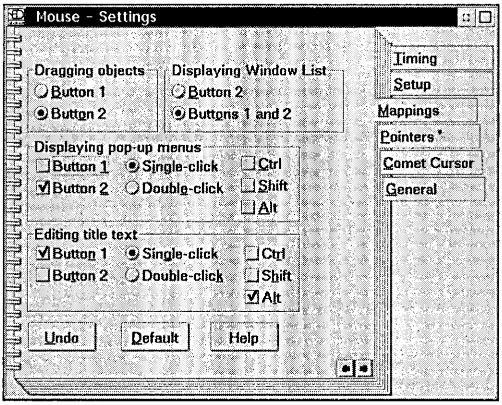
If you are left handed, you normally keep the mouse on the left side of your keyboard and move the mouse with your left hand. In this case, you would still like to operate mouse button 1 with your thumb or index finger. The left-handed mouse setting allows you to do this. Of course, if you want, you can specify the left-handed operation of the mouse even if you operate the mouse with your right hand. Specifying the left-handed operation of the mouse just reverses the functions of the mouse buttons.

This setting applies to all windows, including WIN-OS/2 windows, so you do not have to use the Windows Control Panel to change this setting for WIN-OS/2 windows.

Mappings page

The Mappings page allows you to change the default mappings for combinations of mouse buttons and keyboard keys that translate into Workplace Shell commands (see Figure 7-3).

Figure 7-3.
Mouse Object — Mappings Page



Mouse - Settings

Dragging objects
○ Button 1
● Button 2

Displaying Window List
○ Button 2
● Buttons 1 and 2

Displaying pop-up menus
□ Button 1 ● Single-click □ Ctrl
☑ Button 2 ○ Double-click □ Shift
□ Alt

Editing title text
☑ Button 1 ● Single-click □ Ctrl
□ Button 2 ○ Double-click □ Shift
☑ Alt

I suggest that you not change these settings unless you absolutely hate one or more of the default mappings. Nonstandard mappings can be very confusing if someone else uses your system or if you try to use someone else's system.

The mappings are all specified in terms of mouse button 1 or mouse button 2, not left mouse button and right mouse button, because the left-handed versus right-handed mouse setting affects which button is which.

Warning

Make sure that you don't assign two different functions to the same mouse button. For example, the default mapping for "Displaying pop-up menus" is a single click of mouse button 2. If you also select mouse button 2 for "Displaying Window List" and then move the mouse over the Desktop and press and release mouse button 2, the Window List will be displayed, and then it will disappear and the pop-up menu for the Desktop will appear. You can fix this problem by assigning a different combination of mouse buttons and keyboard keys to the "Displaying pop-up menus" function.

The mappings that can be changed are

- **Dragging objects** — The default setting is mouse button 2. You can move the mouse pointer over an object and then press and hold mouse button 2 to drag the object to its destination. You can switch to mouse button 1 for dragging objects if you wish. Other mouse operations will not be affected.
- **Displaying Window List** — This command shows the Window List if you perform the specified action when the mouse is over the Desktop. The default mapping is to display the Window List when mouse button 1 and mouse button 2 are pressed simultaneously. Since the default action involves the use of both mouse buttons, it is not affected by the left-handed versus right-handed setting. If you don't like to press both mouse buttons to see the Window List, you can choose the alternate mapping, which is mouse button 2.
- **Displaying pop-up menus** — This function requests the pop-up menu for an object. The default mapping is a single click of mouse button 2. You can change this function to be a single- or double-click of either mouse button. You can also specify that you have to be holding some combination of the **Ctrl**, **Shift**, and **Alt** keys on the keyboard when you single- or double-click the specified mouse button in order to activate this function.

  If you are really masochistic, you can specify that you have to hold the **Ctrl**, **Shift**, and **Alt** keys on the keyboard while simultaneously pressing the left and right mouse buttons to display the pop-up menu. Or you can specify that you have to hold the **Ctrl**, **Shift**, and **Alt** keys while double-clicking either mouse button.

Shhh! Tip

I suggest that you keep the combination simple — the most complicated combination I would want to use is double-clicking on one of the mouse buttons.

- **Editing title text** — This function edits the name of an object directly from an open view of the folder that contains the object. For example, if you see an object in an open Icon view of a folder, you can change the name of the object by moving the mouse pointer over the icon or title of the object and pressing the mouse button and key combination you select here. The system then creates an entry field for you to edit the object's name. When you are finished, you simply move the mouse pointer out of the entry field and click either mouse button. The entry field is replaced with the changed name of the object.

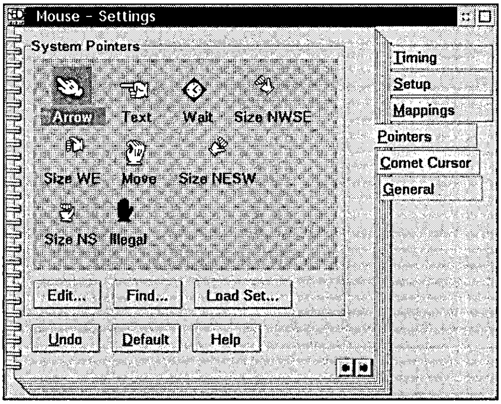
  The default mapping for this function is a single click of mouse button 1 while you hold **Alt**. Again, you can change the default mapping to any combination of mouse buttons and the **Ctrl**, **Shift**, and **Alt** keys.

Remember, unless you have very high dexterity, keep the combination simple — perhaps holding one key while single- or double-clicking one of the mouse buttons.

Pointers page

The Pointers page allows you to change the appearance of some of the most frequently used mouse pointers (see Figure 7-4).

Figure 7-4.
Mouse Object — Pointers Page



The mouse pointers you can specify are

- **Arrow** — This is the normal mouse pointer.
- **Text** — This mouse pointer is used when the mouse is within an entry field. The Text mouse pointer marks the cursor within the text to be edited.

✗ Wait — This mouse pointer is displayed when the system (or the window the mouse pointer is over) is busy and can't accept any input.

✗ Size NWSE — This is the mouse pointer you see when you move the mouse over the upper-left or lower-right corner of a sizeable window.

The name of this pointer, Size NorthWest to SouthEast, indicates the orientation of the default mouse pointer — a double-headed arrow pointing to the top-left and bottom-right corners of the screen, indicating the directions in which you can drag the sizing border of the window.

✗ Size WE — This is the mouse pointer you see when you move the mouse over the left or right border of a sizeable window.

The name of this pointer, Size West to East, indicates the orientation of the default mouse pointer — a double-headed arrow pointing left and right, indicating the directions in which you can drag the sizing border of the window.

✗ Move — This is the mouse pointer you see in some application programs, such as the Dialog Editor that comes with the OS/2 Programmer's Toolkit, indicating you can move a window. This pointer is not used by the Workplace Shell.

✗ Size NESW — This is the mouse pointer you see when you move the mouse over the upper-right or lower-left corner of a sizeable window.

The name of this pointer, Size NorthEast to SouthWest, indicates the orientation of the default mouse pointer — a double-headed arrow pointing to the top-right and bottom-left corners of the screen, indicating the directions in which you can drag the sizing border of the window.

✗ Size NS — This is the mouse pointer you see when you move the mouse over the top or bottom border of a sizeable window.

The name of this pointer, Size North to South, indicates the orientation of the default mouse pointer — a double-headed arrow pointing up and down, indicating the directions in which you can drag the sizing border of the window.

✗ Illegal — This is the mouse pointer you see when you try to drag an object over a place where you can't drop it. For example, if you try to drag a nonprintable object over a Printer icon, you will see the Illegal mouse pointer.

Note

The System Pointers area shows you the current set of mouse pointers being used by the system. These pointers do not correspond to a Pointer Set (a folder containing pointer definitions — see the following section); they are simply the pointers that are currently active. There are two ways to change the pointers:

- ✗ You can move the mouse pointer over the picture of the pointer you want to change and double-click mouse button 1. This action invokes the Icon Editor for the selected pointer. When you leave the Icon Editor, the changed mouse pointer is made into the current pointer.
- ✗ You can select the pointer by single-clicking on the pointer in the System Pointers area, then pressing the **Edit...** push button. This action also invokes the Icon Editor for the specified pointer.

Note

The problem with directly editing the mouse pointers in the System Pointers area is that you don't keep a copy of your changed pointer anywhere you can access. This means that if you want to move to another system, you can't take your customized pointers with you; you will have to recreate them. Also, if you want to restore some, but not all, of the original pointers, you will have no way to selectively restore your customizations. Using Pointer Sets, as described below, can help you overcome these limitations.

You can also replace a single pointer by selecting the pointer in the System Pointers area and then pressing the **Find...** push button. This displays the Find Objects dialog box, from which you can specify where to look for pointer objects. These objects must be files that have the .PTR file extension. The default directory is \OS2\POINTERS on the boot drive. This directory contains several pointers you can use and is the anchor point for the Pointer Sets.

Shhh! Tip

If you find a pointer you like, you only have to select the pointer in the Find Results container and click on the **OK** push button. The pointer you choose will replace the selected mouse pointer. If you don't find a suitable pointer object, just double-click on the **System** menu for the Find Results container, and the mouse pointer won't be changed.

The **Load Set...** push button can be used to load a Pointer Set.

The **Undo** push button restores all of the mouse pointers to what they were when the Mouse Pointers Page was opened; all the changes you made will be discarded.

The **Default** push button restores all of the mouse pointers to their default appearances. This removes all changes you have ever made to the mouse pointers.

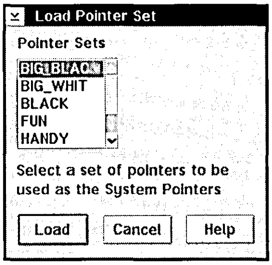
Pointer Sets

A Pointer Set is simply a folder that contains one or more pointer objects. The pointers in a Pointer Set can be used to replace the current system pointers. We associate the pointer files in the Pointer Set to the corresponding system pointers by the names of the pointer files. The files in the Pointer Set must conform to the following naming conventions (the names in parentheses are the names of the corresponding system pointers in the System Pointers area):

- ARROW.PTR — Replaces the normal mouse pointer (Arrow)
- TEXT.PTR — Replaces the text mouse pointer (Text)
- WAIT.PTR — Replaces the busy mouse pointer (Wait)
- SIZENWSE.PTR — Replaces the upper-left/lower-right corner sizing mouse pointer (Size NWSE)
- SIZEWE.PTR — Replaces the left/right sizing mouse pointer (Size WE)
- MOVE.PTR — Replaces the move window mouse pointer (Move)
- SIZENESW.PTR — Replaces the upper-right/lower-left corner sizing mouse pointer (Size NESW)
- SIZENS.PTR — Replaces the top/bottom sizing mouse pointer (Size NS)
- ILLEGAL.PTR — Replaces the illegal operation mouse pointer (Illegal)

Using existing Pointer Sets. If you press the **Load set...** push button, you will be presented with the Load Pointer Set dialog box (see Figure 7-5), which contains the list of Pointer Sets known to the system. You can choose one of the Pointer Sets by double-clicking on its name or by single-clicking on its name and then pressing the **Load** push button.

When you load a Pointer Set, the system replaces the current system pointers with the corresponding validly defined pointers within the Pointer Set. If no pointer object corresponding to a system pointer is defined in the Pointer Set, the system pointer is not changed. (It is not reset to its default value — it is simply not affected by the Load Pointer Set operation.)



Load Pointer Set

Pointer Sets

BIG_BLAC
BIG_WHIT
BLACK
FUN
HANDY

Select a set of pointers to be used as the System Pointers

Figure 7-5. Load Pointer Set Dialog Box

Building your own Pointer Sets. You can build your own Pointer Sets. Pointer Sets reside in the \OS2\POINTERS directory on the boot drive. You can open the \OS2\POINTERS folder, create a new folder to hold your Pointer Set, and populate the Pointer Set folder with pointer objects that follow the required naming convention. You can copy existing pointers into the Pointer Set folder or create new pointer objects using the Icon Editor.

Tip

If you use the Icon Editor, make sure you specify the type of object being created as a Pointer, not an Icon or a Bitmap, and specify the *hotspot* for each pointer you create. The hotspot is the actual pixel within the pointer that indicates where the mouse is pointing. For example, for the normal move pointer, the hotspot is the tip of the arrow.

Uncensored

The system allows the pointers you create to be either 16-color pointers or black-and-white pointers. If the pointers you are creating contain only black and white, like the default pointer set, you should select the BW form for your pointers. The color form will work, but in certain circumstances the system performance will be better with a black-and-white pointer than with a color pointer. This is because some display adapters allow a black-and-white pointer to be loaded into a hardware register on the adapter and manipulated in the hardware. You might notice that the Pointer Sets supplied with OS/2 Warp all contain colored pointers, even though they use only black and white. The reason is simple — we didn't understand the performance impact until after we shipped the product!

Tip

If the mouse pointer starts blinking when you move it over certain areas of the screen, it is likely that the pointer is a color pointer. If you change the pointer to a BW form pointer, it will probably stop blinking.

CD-ROM

If you are using one of the Pointer Sets supplied with OS/2 Warp, you can replace them with the corresponding Pointer Sets on the CD-ROM included with this book. We reconstructed the Black, White, Big Black, and Big White Pointer Sets to contain only BW form pointers.

There is another way to customize the mouse pointers using the Pointers page. You can open a folder containing a mouse pointer that you want to use and drag it into the System Pointers area of the Pointers page.

- If you drop a single pointer object on top of one of the pointers in the System Pointers area, the corresponding system pointer is changed.
- If you drag and drop one or more pointer objects (you can select multiple pointer objects, then drag them as a group) into an unoccupied space within the System Pointers area, the objects that conform to the naming convention used for Pointer Sets replace the corresponding system pointers.
- Also, you can drag and drop the icon representing a folder into an unoccupied space within the System Pointers area. The folder is taken to be a Pointer Set, and all pointer objects within the folder that conform to the Pointer Set naming convention replace the corresponding system pointers.

With the facilities we have provided, you can customize your system pointers by building a custom Pointer Set containing pointer objects that you gathered from various sources, and then loading that Pointer Set.

CD-ROM

In case you haven't already found them, OS/2 Warp has a couple of alternative pointers in the OS2\POINTERS folder that you can use. Also, we have provided some more pointers on the CD-ROM included with this book.

Comet Cursor

The Comet Cursor feature (see Figure 7-6) was included to make it easier for you to find your cursor when you are using a laptop or notebook computer.

Developer's Desktop

However, I think this feature is cute, and it makes it easier to find my cursor even on my desktop system, so I use it all the time.

Figure 7-7 is a sample of what the Comet Cursor looks like. It isn't exactly a screen capture of a real Comet Cursor because I found the Comet Cursor to be as camera-shy as a vampire. However, it gives you a reasonable idea of what the Comet Cursor looks like when you activate it.

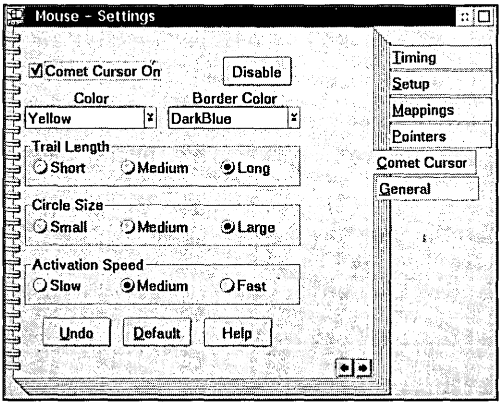


Figure 7-6.
Mouse Object —
Comet
Cursor Page

☑ Comet Cursor On

Color: Yellow

Border Color: DarkBlue

Trail Length: ○ Short ○ Medium ● Long

Circle Size: ○ Small ○ Medium ● Large

Activation Speed: ○ Slow ● Medium ○ Fast

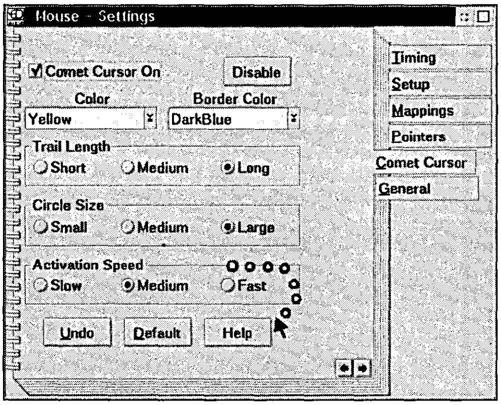


Figure 7-7.
Sample
Comet Cursor

☑ Comet Cursor On

Color: Yellow

Border Color: DarkBlue

Trail Length: ○ Short ○ Medium ● Long

Circle Size: ○ Small ○ Medium ● Large

Activation Speed: ○ Slow ● Medium ○ Fast

The Comet Cursor leaves a trail of dots where the cursor has been. You can customize the speed at which the cursor leaves its droppings, how big the dots are, their color, and how long the trail is. You can also turn the Comet Cursor on or off at will.

To use the Comet Cursor, you first have to check the **Comet Cursor On** check box, then shut down and restart your system. Once the system is restarted, you can reopen the Mouse object and customize the Comet Cursor. The **Disable** push button temporarily turns off the Comet Cursor. You can restart the Comet Cursor by checking the **Comet Cursor On** check box again, but you don't have to shut down and reboot your system this time.

Several settings can be changed on the Comet Cursor page:

- You can choose the color used for the inside of the Comet Cursor's dots from the Color combination box. If you select **Clear,** the dots will be transparent. With this option selected, the Comet Cursor is a series of hollow circles. If you select **Invert,** each circle is the inverse of the color the cursor is passing over, and you can't specify a border color.
- You can choose the color used for the circles surrounding the Comet Cursor's dots from the Border Color combination box (unless you specified Invert as the dot color).

Developer's Desktop

I prefer to use a border color that contrasts with the dot color. By selecting a contrasting set of colors, you will ensure that you will always be able to see the Comet Cursor, no matter what color area the cursor is traveling over.

- You can select the length of the Comet Cursor's trail from the Trail Length group. You will have to experiment with the **Short, Medium,** and **Long** selections to find the one that you like best.
- You can select the size of the Comet Cursor's dots from the Circle Size group. Again, you will have to experiment until you find out whether you like the **Small, Medium,** or **Large** circles.
- The Activation Speed group is used to select how fast you have to move the cursor before the Comet Cursor starts leaving its trail. **Slow** specifies that the Comet Cursor's trail is generated even if you move the mouse slowly. **Medium** specifies an intermediate speed. **Fast** specifies that the Comet Cursor's trail is generated only if you move the mouse quickly.

Since these settings all take effect immediately, you can choose an option and see how it works by just moving the cursor around within the dialog box.

Uncensored

If you use the Comet Cursor, you might notice that it doesn't always start when you restart your system. This is due to a bug in OS/2 Warp. If you specify **SET RESTARTOBJECTS=STARTUPFOLDERSONLY** (or some other similar option) in your CONFIG.SYS file, the Comet Cursor won't start. You can easily fix this by typing the statement **START COMETRUN.EXE** into your STARTUP.CMD file.

The Keyboard Object

The Keyboard object in the System Setup folder can be used to specify characteristics of your keyboard. These characteristics include

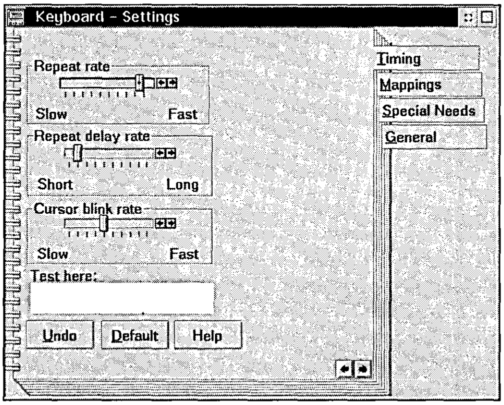
- **Timing** — How fast a key repeats if you hold it down, the delay before a key starts repeating, and the cursor blink rate
- **Mappings** — How key sequences are mapped to special functions
- **Special Needs** — How to customize the keyboard to make it easier for people with special needs to use

These characteristics can be specified through the Settings Notebook of the Keyboard object. To open the Settings Notebook, just double-click on the **Keyboard** object in the System Setup folder.

Timing page

The Timing page allows you to specify the repeat rate, repeat delay rate, and cursor blink rate for the keyboard (see Figure 7-8).

Figure 7-8. Keyboard Object — Timing Page



Repeat rate

The Repeat rate group is used to specify how fast a keystroke is repeated if you hold down the key:

- If you find that keystrokes are repeated too quickly when you hold down a key, move the slider bar to the left.

- If keystrokes are repeating too slowly, move the slider bar to the right.

Tip

After adjusting the slider bar, you can see the effect of your change by moving the cursor to the Test here window and holding down a key. Adjust the slider bar until the key repeats at the rate you want.

Repeat delay rate

The Repeat delay rate group is used to specify how long the system waits before starting to repeat keystrokes:

- If you have to hold a key down too long before the system starts repeating keystrokes, move the slider bar to the right.
- If the system starts repeating keystrokes too quickly while you are holding down a key, move the slider bar to the left.

Tip

After adjusting the slider bar, you can see the effect of your change by moving the cursor to the Test here window and holding down a key. Adjust the slider bar until the system starts repeating the keystroke at the rate you want.

Cursor blink rate

The Cursor blink rate group is used to specify how fast the cursor blinks:

- If the cursor blinks too slowly for you, move the slider bar to the right.
- If the cursor blinks too fast, move the slider bar to the left.

Tip

After adjusting the slider bar, you can see the effect of your change by moving the cursor to the Test here window and observing the speed at which the cursor blinks. Adjust the slider bar until the cursor blinks at the rate you want.

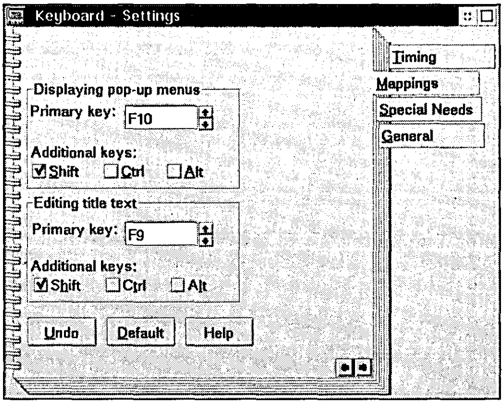
Warning

The keyboard rates you can adjust with the Keyboard object affect the keyboard only when an OS/2 window has the focus — they have no effect when a WIN-OS/2 window has the focus. In order to change your keyboard rates within a WIN-OS/2 window, you have to open the Windows Control Panel and select **Keyboard Settings.** Within the Keyboard Settings dialog box, you can adjust the Delay Before First Repeat and Repeat Rate to set your keyboard preferences for WIN-OS/2 sessions.

Mappings page

The Mappings page is used to choose the keystrokes or key combinations that activate certain special Workplace Shell functions (see Figure 7-9).

Figure 7-9. Keyboard Object — Mappings Page



Displaying pop-up menus
Primary key: F10
Additional keys:
☑ Shift ☐ Ctrl ☐ Alt

Editing title text
Primary key: F9
Additional keys:
☑ Shift ☐ Ctrl ☐ Alt

Just as with the mouse mappings, make sure that you don't define the same key combination for more than one function, or you will get unpredictable results.

The following mappings can be changed:

- **Displaying pop-up menus** — This function requests the pop-up menu for an object.

  The default mapping for this function is Shift-F10. You can change this mapping to any of the function keys (F1 through F12), or to the PgUp, PgDn, Home, End, Insert, Tab, or Enter keys. You can also specify that you have to be holding some combination of the Ctrl, Shift, and Alt keys while pressing the selected key in order to activate this function. If you are really masochistic, you can specify that you have to hold the Ctrl, Shift, and Alt keys on the keyboard while simultaneously pressing the F10 key to display the pop-up menu.

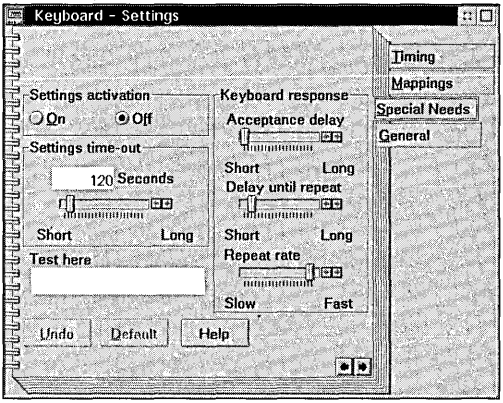
  I suggest that you keep the combination simple — the most complicated combination I would want to use is Shift and one of the function keys.

- **Editing title text** — This function is used to edit the name of an object directly from an open view of the folder that contains the object. For example, if you see an object in an open Icon view of a folder, you can change the name of the object by moving the selection cursor to the icon (using the left- and right-arrow keys) and pressing the key combination you specify for Editing title text. The system creates an entry field for you to edit the object's name. When you are finished, simply move the mouse pointer out of the entry field and click either mouse button. The entry field is then replaced with the changed name of the object.

The default mapping for this function is Shift-F9. Again, you can change the default mapping to any combination of the function keys (F1 through F12), PgUp, PgDn, Home, End, Insert, Tab, or Enter with Ctrl, Shift, or Alt. Keep it simple, though — perhaps holding down Shift while pressing one of the function keys.

Special Needs page

The Special Needs page is used to customize the keyboard to make it easier to use for people who have special needs (see Figure 7-10).



Keyboard - Settings

Settings activation
○ On ● Off

Settings time-out
120 Seconds

Keyboard response
Acceptance delay
Delay until repeat
Repeat rate

Figure 7-10. Keyboard Object — Special Needs Page

The Special Needs functions include changing several keyboard rates and activating the Sticky Keys input method. The Sticky Keys input method lets you press and release one or more keys and have the system behave as if those keys were being held down.

To activate the Special Needs functions, press the **On** radio button in the Settings activation group. To deactivate the Special Needs functions, press the **Off** radio button. The ability to activate and deactivate the Special Needs functions at will is particularly useful if you are sharing your computer with other people, some of whom can make use of the Special Needs functions.

The Settings time-out group can be used to specify a time limit after which the Special Needs settings will automatically be deactivated.

The Special Needs page also offers the following keyboard adjustments:

- ✗ **Acceptance delay** — You can use the Acceptance delay slider bar to set the amount of time a key has to be pressed and held down before the system accepts the keystroke. If you have trouble hitting the correct key immediately, you can move the slider bar to the right. Moving the slider bar to the right allows you to press and release wrong keys, then hold the correct key for at least the Acceptance delay time so that the system will register the correct keystroke. After adjusting the slider bar, you can test your new setting by moving the cursor to the Test here entry field and pressing some keys. Adjust the slider bar until the system accepts your keystrokes after the delay that you want.
- ✗ **Delay until repeat** — You can use the Delay until repeat slider bar to set the amount of time a key has to remain pressed before the system starts repeating the keystroke. After adjusting the slider bar, you can test your new setting by moving the cursor to the Test here entry field and pressing some keys. Adjust the slider bar until the system delays repeating your keystrokes for the right amount of time.
- ✗ **Repeat rate** — You can use the Repeat rate slider bar to select the rate at which keystrokes are generated while you hold down a key. If the system repeats keystrokes too slowly, move the slider bar to the right. If the system repeats keystrokes too quickly, move the slider bar to the left. After adjusting the slider bar, you can test your new setting by moving the cursor to the Test here entry field and pressing some keys. Adjust the slider bar until the system repeats your keystrokes at the rate you want.

While the Special Needs settings are active, you can use the Sticky Keys method of input. This allows you to use key combinations without physically holding down more than one key at a time. To define a key that you want to be "sticky" — that is, to behave as if it were being held down — press **Shift** three times, then press the key you want to be "stuck." For example, to display the Window List, you would normally press and hold down **Ctrl**, then press **Esc**. Using the Sticky Keys function, you can press **Shift** three times, press and release **Ctrl** (this key is now "stuck"), and then press and release **Esc**.

The System Object

The System object can be used to specify the resolution of your display adapter and various default options for the Workplace Shell.

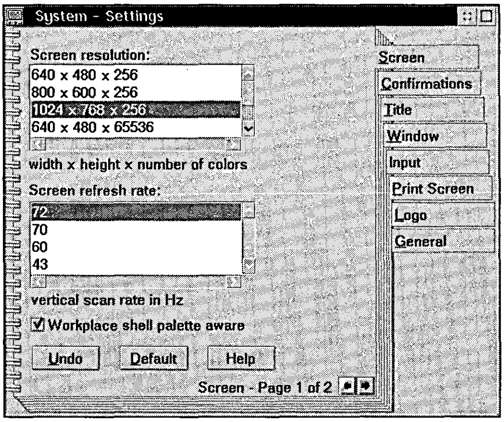
The options you can specify include

- ✗ **Screen** — The resolution of your display adapter
- ✗ **Confirmations** — What operations you want the Workplace Shell to confirm before performing the operation
- ✗ **Title** — How the system handles your trying to create two objects in a folder with the same title
- ✗ **Window** — System default characteristics of windows
- ✗ **Input** — Whether type ahead is enabled or disabled
- ✗ **Print Screen** — Whether print screen is enabled or disabled
- ✗ **Logo** — Time for the logo screen for application programs

These characteristics can be specified through the Settings Notebook of the System object. To open the Settings Notebook, just double-click on the System object in the System Setup folder.

Screen pages

The Screen pages allow you to specify the characteristics of your display adapter (see Figure 7-11).



System - Settings

Screen resolution:
640 x 480 x 256
800 x 600 x 256
1024 x 768 x 256
640 x 480 x 65536
width x height x number of colors

Screen refresh rate:
72
70
60
43
vertical scan rate in Hz

☑ Workplace shell palette aware

Figure 7-11. System Object — Screen Page 1

Screen resolution

The Screen resolution list box has two possible meanings:

- ✗ If your display adapter doesn't support dynamically changing the screen resolution (if it is a VGA display adapter, for instance), the list box contains a single resolution. This is the resolution currently being used by your display adapter. For example, a VGA display might be set to a resolution of 640 × 480 × 16 colors, and the only entry in the list box would be 640 × 480 × 16. If your display is capable of supporting other resolutions by loading a different device driver, you would have to use Selective Install or some other method to change resolutions. Check your display adapter's documentation to determine how to change its resolution.
- ✗ The second form of the Screen resolution list box is a list of possible resolutions supported by your display adapter, with the resolution that your display adapter is currently set to highlighted. In this case, you can specify the resolution you want by selecting the resolution (single-click on the resolution in the list box). You will then have to shut down the system and restart it for the specified resolution change to take effect.

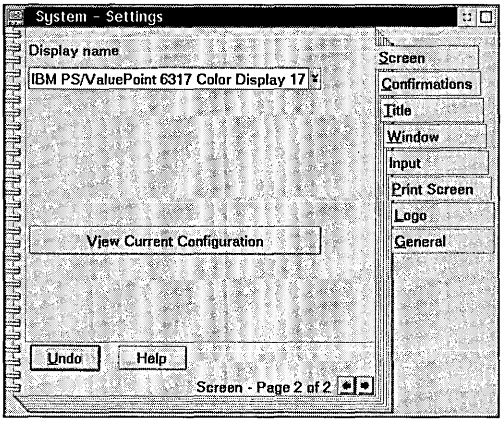
Screen refresh rate

The Screen refresh rate list box, if displayed, shows you the vertical scan rate that your display adapter is set to. If you want to change the vertical scan rate, select another scan rate from the list box. Again, you will have to shut down the system and restart it for the specified scan rate change to take effect.

Workplace shell palette aware

If the Workplace shell palette aware check box is not disabled (that is, grayed out), you can check this box to have the Workplace Shell participate as a palette-managed application when displaying bitmaps in the background of folders or the Desktop. See the Palette Manager section in Chapter 6 for an explanation of the effect of turning on the Workplace Shell palette aware option.

The second Screen page in the System object's Settings Notebook, if it exists, allows you to specify the type of display that is attached to your display adapter (see Figure 7-12). For some display adapters, the variety of screen resolutions and the range of vertical refresh rates is partially determined by the characteristics of the display you are using. If this is the case, the second Screen page is presented.


System - Settings

Display name: IBM PS/ValuePoint 6317 Color Display 17

Screen - Page 2 of 2

Figure 7-12. System Object — Screen Page 2

Display name

You can select the model of the display you are using from the Display name combination box. If you select a different display, you might find that your choices on the first Screen page have changed.

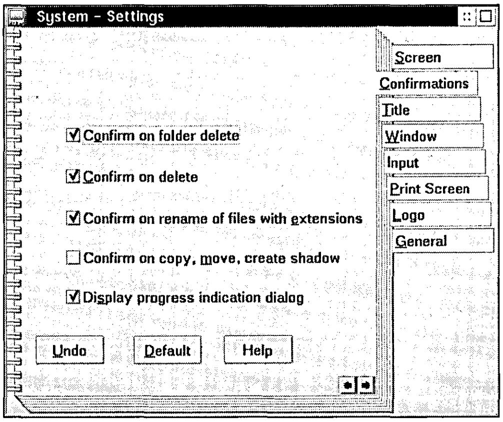
Be very careful when specifying the name of the display you are using. If you give the system incorrect information and then select a Screen resolution: or vertical refresh rate: that is not supported by your display, you could cause the system to select a mode of operation that renders the system inoperative. Even worse, for certain displays, it might be possible to actually damage the display if you set the system to a display mode that it is incapable of using.

View Current Configuration

You can press the **View Current Configuration** push button to ask the system to display information about the capabilities of the display you have defined as attached to your display adapter.

Confirmations page

The Confirmations page can be used to specify what operations the Workplace Shell will ask you to confirm before performing the operation (see Figure 7-13).



☑ Confirm on folder delete

☑ Confirm on delete

☑ Confirm on rename of files with extensions

□ Confirm on copy, move, create shadow

☑ Display progress indication dialog

Figure 7-13.
System Object — Confirmations Page

Confirm on folder delete

The Confirm on folder delete check box can be used to make the Workplace Shell display a confirmation dialog before deleting a folder. This confirmation is displayed if you specify the Delete option from the pop-up menu of the folder, drag the folder to the Shredder object, or delete a folder containing another folder. The confirmation dialog contains four choices:

- If you select **Yes**, the folder will be deleted if you also answer **Yes** to all confirmation dialogs about objects or folders contained within the folder.
- If you select **Yes to all**, the folder will be deleted and no more confirmation dialogs will be presented.
- If you select **No**, the folder will not be deleted, but the Workplace Shell will still attempt to delete any other objects you specified.
- If you select **Cancel**, the folder will not be deleted, and the Workplace Shell will not try to delete any other objects.

Confirm on delete

The Confirm on delete check box can be used to make the Workplace Shell display a confirmation dialog before deleting any object other than a folder. This confirmation is displayed if you choose the **Delete** option from the pop-up menu of an object, drag the object to the Shredder object, or delete a folder containing objects. The confirmation dialog contains four choices:

- If you select **Yes**, the object will be deleted.

- If you select **Yes to all**, all selected objects will be deleted and no more confirmation dialogs will be presented.
- If you select **No**, the object will not be deleted, but the Workplace Shell will still attempt to delete any other objects you specified.
- If you select **Cancel**, the object will not be deleted, and the Workplace Shell will not try to delete any other objects.

Confirm on rename of files with extensions

The Confirm on rename of files with extensions check box can be used to make the Workplace Shell display a confirmation dialog before renaming a file if the new name has a different file extension from the current name, or if the new name has no file extension and the current name does. This confirmation reminds you that if you have a data file that is associated with an executable program by the file extension, you might lose that association by renaming the file.

Confirm on copy, move, create shadow

The Confirm on copy, move, create shadow check box can be used to make the Workplace Shell display a confirmation dialog before copying or moving an object, or creating a shadow object.

Display progress indication dialog

The Display progress indication dialog check box can be used to make the Workplace Shell display a dialog during a move, copy, or delete operation to show you how much of the operation has been performed.

This is particularly important if you tell the Workplace Shell to operate on multiple objects. For example, if you select several objects within a folder and drag the set of objects to the Shredder, the Progress Indication dialog shows you which objects are still waiting to be deleted. You can then get some idea of how long it will take the Workplace Shell to complete the operation. If it's going to take a long time and you want to stop the process, you can press the **Cancel** push button.

Bypassing Confirm on delete

Warning

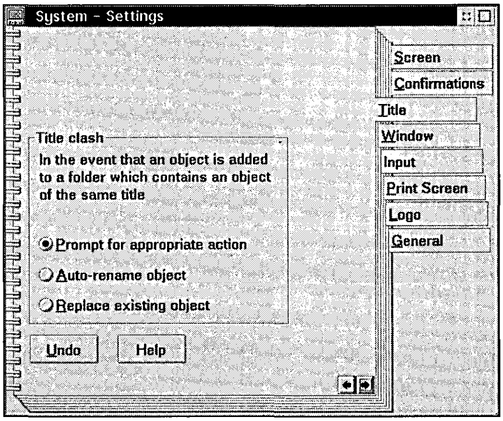
We added a new feature into OS/2 Warp that partially bypasses your Confirm on delete selection. If you move one or more nondeletable objects into a folder and try to delete the folder, the Workplace Shell will always prompt you to confirm the deletion of these objects, regardless of the Confirm on delete setting.

If you respond that it is OK, the Workplace Shell will delete these objects even though they are marked nondeletable. This feature allows you to get rid of objects that you really don't want. For example, the Solid Color Palette and the Mixed Color Palette are both marked as nondeletable. If you want to remove the Solid Color Palette object from the System Setup folder, you can create a new folder, move the Solid Color Palette object into this new folder, then delete the folder. After you confirm the deletion, both the new folder and the Solid Color Palette object will be gone.

Title page

The Title page allows you to specify what action you want the Workplace Shell to take when you ask it to create an object in a folder that has the same name as another object already in the folder (see Figure 7-14). This can occur if you copy or move an object into a folder or if you create an object from a template.

Figure 7-14. System Object — Title Page


System - Settings

Title clash

In the event that an object is added to a folder which contains an object of the same title

● Prompt for appropriate action

○ Auto-rename object

○ Replace existing object

Prompt for appropriate action

The Prompt for appropriate action check box can be used to make the Workplace Shell display a dialog asking you what action to perform when this problem occurs. The dialog allows you to choose one of three options:

- ✗ **Rename object to** — Allows you to choose another title for the new object
- ✗ **Replace existing object** — Specifies that the old object with the same title is to be deleted before the new object is copied or moved into the folder
- ✗ **Append to existing object** — Specifies that the object being copied or moved into the folder is to be appended to the old object. This option applies only to copying or moving a data file into a folder that already contains a data file with the same name.

Once you have selected the option you want, press the **OK** push button to have the Workplace Shell continue the operation. Or press the **Cancel** push button to cancel the move or copy request.

Auto-rename object

The Auto-rename object check box can be used to make the Workplace Shell automatically rename the new object. The name chosen by the Workplace Shell is the old name followed by a colon and a sequence number. For example, if you try to copy an object named XYZ into a folder that already contains an XYZ object, the Workplace Shell will name the new object XYZ:1.

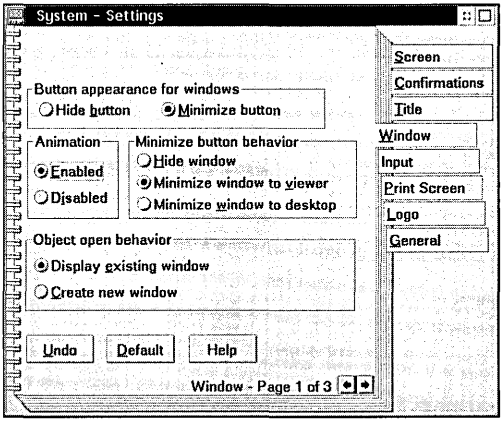
Replace existing object

The Replace existing object check box can be used to make the Workplace Shell automatically replace the old object with the new object. This is similar to the system action if you copy a file (from an OS/2 or DOS window) into a directory that already contains that file.

Window pages

The Window pages allow you to specify the system default values for the behavior of windows.

The first Window page (see Figure 7-15) lets you specify the system default values for the behavior of open views of an object. These defaults apply to all objects unless you specify different values in the first Window page of the Settings Notebook for the object.



Button appearance for windows
○ Hide button ● Minimize button

Animation
● Enabled
○ Disabled

Minimize button behavior
○ Hide window
● Minimize window to viewer
○ Minimize window to desktop

Object open behavior
● Display existing window
○ Create new window

Window - Page 1 of 3

Figure 7-15.
System Object — Window Page 1

Button appearance for windows

The Button appearance for windows option specifies the system default value for whether windows contain Hide buttons or Minimize buttons.

- ✗ If you select **Hide button**, windows will contain a Hide button on the title bar, and they will be removed from the screen (hidden) when you press the Hide button.
- ✗ If you select **Minimize button**, windows will contain a Minimize button on the title bar, and they will be minimized to the Desktop, minimized to the Minimized Window Viewer, or hidden whenever you press the Minimize button on the window's title bar, based on what you choose for the Minimize button behavior option.

If you select one of the options under Button appearance for windows in an object's Settings Notebook, that value determines which button appears in all windows for that object instead of the system default value.

Minimize button behavior

The Minimize button behavior option specifies the system default value for the behavior of Minimize buttons on windows.

- ✗ If you select the **Hide window** radio button, windows will be hidden when you click on their Minimize buttons.

- ✘ If you select the **Minimize window to viewer** radio button, windows will be minimized to the Minimized Window Viewer when you click on their Minimize buttons.
- ✘ If you select the **Minimize window to desktop** radio button, windows will be minimized to the Desktop when you click on their Minimize buttons.

The system default value applies to windows of all objects for which you don't select an option under Minimize button behavior in the first Window page of the Settings Notebook.

Animation

The Animation option specifies whether or not window animation is to be performed.

- ✘ If you select **Enabled**, an animation sequence is drawn whenever a window is opened or closed. When a window is opened, a sequence of frames is drawn on the screen showing the window expanding from its original size to its normal size, at which point the full window is painted. When a window is closed, a sequence of frames is drawn showing the window contracting from its normal size to its original size.
- ✘ If you select **Disabled**, windows simply open and close without the animation sequences being drawn.

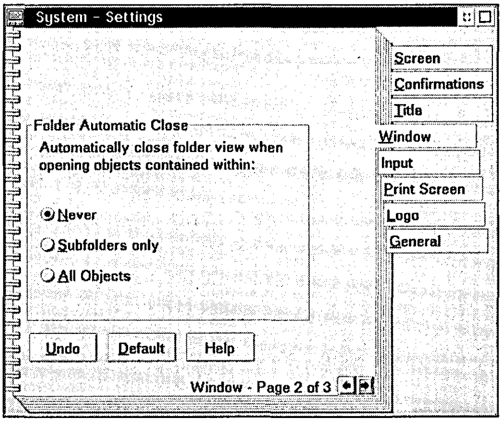
Object open behavior

The Object open behavior option is used to specify the system default value for what happens when you try to open a view of an object when that view is already open.

- ✘ If you select the **Display existing window** radio button, when you try to open a view of an object that is already open, the system will bring the view's window to the front of the screen and give focus to the window.
- ✘ If you select the **Create new window** radio button, when you try to open a view of an object that is already open, the system will open another view of the object.

The system default value is used for every object unless you specify one of the Object open behavior options on the first Window page for the object.

The second Window Page (see Figure 7-16) allows you to specify the action taken when you open an object from within a folder.



Folder Automatic Close
Automatically close folder view when opening objects contained within:

● Never
○ Subfolders only
○ All Objects

Figure 7-16.
System Object — Window Page 2

Folder Automatic Close

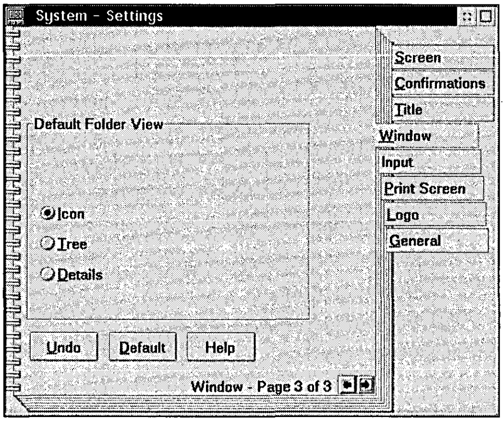
The Folder Automatic Close option allows you to specify the system default value for the action the system takes when it opens an object from within a folder.

- If you specify the **Never** option, the system will never close a folder when you open an object within the folder.
- If you specify the **Subfolders only** option, the system will automatically close a folder when you double-click on a folder within it.
- If you specify the **All objects** option, the system will automatically close a folder when you double-click on any object within it.

The system default value applies to every folder unless you specify one of the Folder Automatic Close options in the Settings Notebook for the folder.

The Folder Automatic Close option doesn't apply to any folder that is marked as a Work Area folder, including the Desktop. Also, you can override the Folder Automatic Close option by holding down **Shift** when you double-click on an object within a folder. In this case, the system will automatically close the folder (unless it is a Work Area folder), regardless of the setting of the Folder Automatic Close option.

The third Window page allows you to specify the default view for all folders in the system (see Figure 7-17). The system default value can be overridden for any particular folder by setting a default view on the Menu page of the Settings Notebook for that folder.



Default Folder View

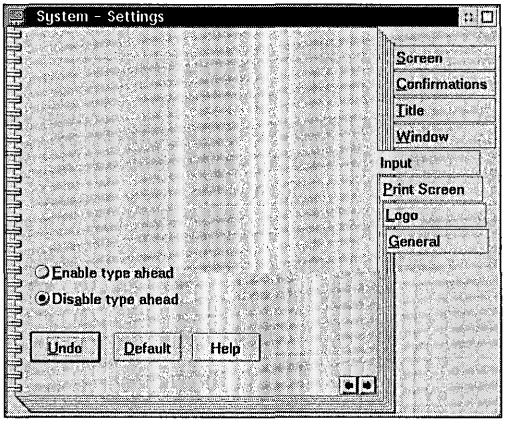
● Icon
○ Tree
○ Details

Figure 7-17.
System Object —
Window Page 3

You can select Icon, Tree, or Details view as the system default folder view by selecting the corresponding radio button under Default Folder View.

Input page

The Input page allows you to specify whether or not type ahead is to be enabled for Presentation Manager programs (see Figure 7-18).



○ Enable type ahead
● Disable type ahead

Figure 7-18.
System Object —
Input Page

If you specify **Enable type ahead**, keyboard and mouse input are saved while a Presentation Manager program is starting. Once the program has been started, the saved input keystrokes, mouse movement, and mouse button clicks are passed on to the program. This allows you to begin a Presentation Manager program and begin typing input for the program while the program is being started.

If you specify **Disable type ahead**, keyboard and mouse input are not saved while a Presentation Manager program is starting. You will have to wait until the program is completely started until you begin typing.

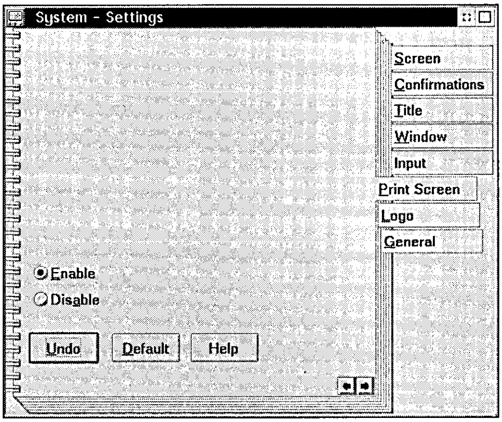
Warning

The **Enable type ahead** option should be used carefully because it increases the time taken to start Presentation Manager programs.

Print Screen page

The Print Screen page allows you to specify whether or not the Print Screen key has any effect (see Figure 7-19).

Figure 7-19.
System Object — Print Screen Page



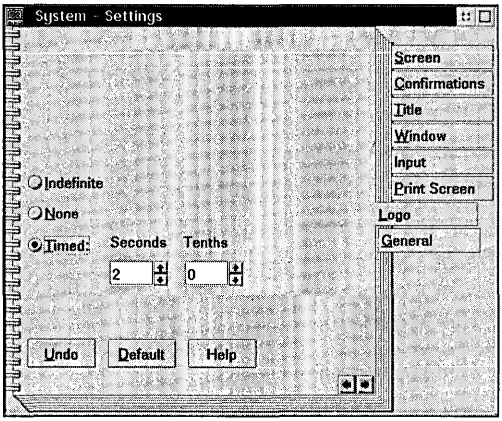
● Enable
○ Disable

✗ If you specify the **Enable** option, you can use the Print Screen key to print either the image of the entire screen or the contents of an OS/2 window. Once the Print Screen option is enabled, you can move the mouse pointer to an open space on the Desktop and press the **Print Screen** key to print an image of the entire screen. If you move the mouse pointer over an OS/2 window and press the **Print Screen** key, you will print an image of the window.

✗ If you specify the **Disable** option, pressing the Print Screen key will have no effect.

Logo page

The Logo page is used to specify how long the logo panel for application programs is to stay on the screen (see Figure 7-20).



○ Indefinite
○ None
● Timed: Seconds 2 Tenths 0

Figure 7-20. System Object — Logo Page

- ✘ If you select the **Indefinite** radio button, the logo panel for an application program will stay on the screen until you explicitly dismiss the panel, usually by pressing the **OK** push button.
- ✘ If you select the **None** radio button, the logo panel for an application program will not be displayed.
- ✘ If you select the **Timed** radio button, the logo panel for an application program will stay on the screen for the amount of time specified by the **Seconds** and **Tenths** spin buttons. The default time is 2.0 seconds. You can change the amount of time the logo panel stays on the screen by changing the values in the **Seconds** or **Tenths** spin buttons.

Note

This option is the system default value, and it affects only application programs that are correctly coded to use the system default value. If you haven't specified this option, it defaults to 2.0 seconds.

Tip

If an application program's logo panel doesn't disappear from the screen within two seconds, try opening the System object, selecting the Logo page, selecting the **None** option, and then selecting the **Timed** option. This sometimes corrects the problem. If it doesn't, the problem is probably in the application program, and you should report the problem to the program's supplier.

The Sound Object

The Sound object in the System Setup folder can be used to specify sound characteristics of your system. These characteristics include

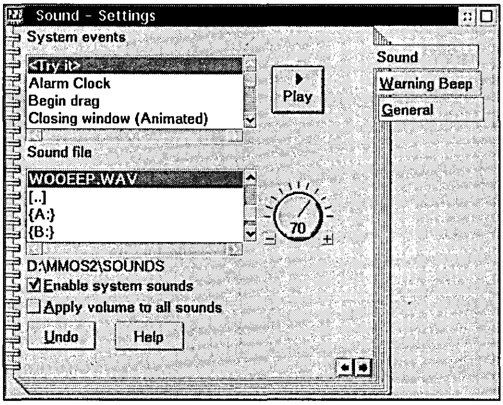
- **Sound** — Assigns sounds to be played during system events
- **Warning Beep** — Specifies whether a beep is sounded when the system warns you about a problem

These characteristics can be specified through the Settings Notebook of the Sound object. To open the Settings Notebook, just double-click on the Sound object in the System Setup folder.

Sound page

The Sound page is present in the Settings Notebook if you installed the multimedia features of OS/2 (see Figure 7-21). The Sound page allows you to select sounds to be played when certain system events occur.

Figure 7-21.
Sound Object — Sound Page



Sound - Settings

System events

<Try it>
Alarm Clock
Begin drag
Closing window (Animated)

Sound file

WOOEEP.WAV
[..]
{A:}
{B:}

D:\MMOS2\SOUNDS

☑ Enable system sounds

□ Apply volume to all sounds

The System events list box displays a list of the system events for which you can assign sounds. These events are:

- **Alarm Clock** — The time specified for an alarm in the System Clock has been reached.
- **Begin drag** — You have started dragging an object in the Workplace Shell.

- ✘ **Closing window (Animated)** — A window is closing; this sound plays only if the Animation option on the Window page of the System object is selected as enabled.
- ✘ **End drag** — You have dropped the object that you are dragging.
- ✘ **Error** — The Workplace Shell or a Presentation Manager application program has displayed an error message.
- ✘ **Information** — The Workplace Shell or a Presentation Manager application program has displayed an informational message.
- ✘ **Lockup** — The Lockup time-out value has expired and the system is locking the keyboard and mouse, or you told the system to lock the keyboard and mouse (either from the LaunchPad or from the Desktop's pop-up menu).
- ✘ **Opening window (Animated)** — A window is opening; this sound plays only if the Animation option on the Window page of the System object is selected as enabled.
- ✘ **Printer error** — The system has issued an error message indicating that an error occurred while printing a file.
- ✘ **Shred** — You have dragged an object to the Shredder object and answered the confirmation dialog (if enabled); the object you dropped on the Shredder is being deleted.
- ✘ **System shutdown** — The system is being shut down; you can start the shutdown process either through the LaunchPad or from the Desktop's pop-up menu.
- ✘ **System startup** — The system is being started or restarted; this sound plays while the Workplace Shell is being initialized.
- ✘ **Warning** — The Workplace Shell or a Presentation Manager program has displayed a warning message.

In order to assign a sound to a system event, you first select the event from the System events list box. You then assign a sound file to the specified system event by selecting it in the Sound file list box.

The entries in the Sound file list box are:

- ✘ <None> — This indicates that no sound is to be played when the system event occurs.
- ✘ *xxxxxxxx*.WAV — This is the name of the sound file that is to be played when the system event occurs. The directory that contains the sound files is displayed immediately below the Sound file list box.

- [..] — This indicates that you want to look for sound files in the parent directory of the directory that is displayed below the Sound file list box.
- [*xxxxxxxx*] — This indicates that you want to look for sound files in the *xxxxxxxx* subdirectory.
- {*x*:} — This indicates that you want to look for sound files on the *x*: drive

You can look for sound files anywhere on your system by double-clicking on the [..], [*xxxxxxxx*], and {*x*:} entries in the Sound file list box to navigate through the directories on your system.

For you to hear a sound when a system event occurs, the sound file must be present at the time the event occurs. Therefore, don't select a sound file on a removable medium (such as a diskette or CD-ROM). If you want to use a sound file that is on a removable medium, first copy the sound file to some directory on your fixed disk (such as the \MMOS2\SOUNDS directory on the boot drive) and specify that copy of the sound file to be used when the system event occurs.

You can press the **Play** push button to hear the sound file you have selected. If you want to hear a sound file without assigning it to a system event, choose **<Try it>** in the System events list box, select the sound file you want to hear, and then press the **Play** push button. You can adjust the volume at which the sound effects you have assigned to system events will be played by adjusting the volume control, which is to the right of the Sound file list box, with the mouse.

The actual volume at which system sounds are played is a function of both the volume you set here and the volume set by the Master Volume Control. The numbers on both this volume control and the Master Volume Control are percentages of the maximum volume. If you want the system sounds to play at their loudest possible volume, move both this volume control and the Master Volume Control to 100 (that is, 100 percent). If this is too loud, you can adjust the volume on your speakers (if they have a volume control), lower the Master Volume Control (which lowers the volume of all sounds played through your sound card), or lower the volume control for the system sounds. You will have to experiment with these three controls until you find the settings that give you the most pleasing volume for the system sounds, sounds played from CDs, and other sounds played by application programs.

The Enable system sounds check box enables or disables all system sounds. If this box is checked, the sounds you have specified will be played when the appropriate system events occur. If you remove the check from this check box, no sounds will play when system events occur.

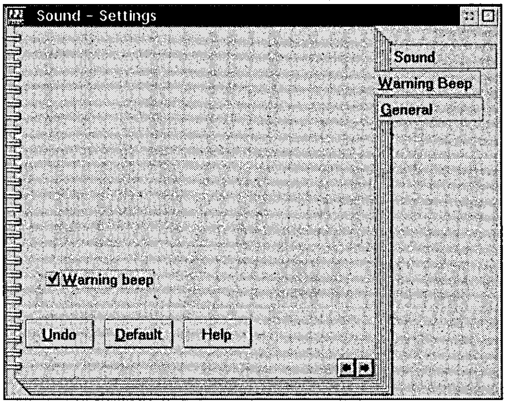
You can use the Apply volume to all sounds check box to specify whether the volume control setting is to apply to only the system sound that is currently selected or to all system sounds. If you check this box, a single specified volume is assigned to all system sounds. If you remove the check from this box, the specified volume is assigned only to the currently selected system sound. This allows you to customize the volume for each system sound individually, in case some sounds were recorded louder than others.

Note

When a system sound is playing and another system event occurs that has a sound associated with it, the first system sound stops playing and the new sound file is played instead. For example, if you have a sound file associated with the System startup event, and you also have a sound file associated with the Opening window event, the System startup sound will be terminated when the first window (like the Desktop) is opened.

Warning Beep page

The Warning Beep page allows you to specify whether the system sounds a tone on the system speaker when the Workplace Shell or a Presentation Manager application program displays a warning message or you press an invalid key (see Figure 7-22).


Sound - Settings

☑ Warning beep

Figure 7-22. Sound Object — Warning Beep Page

If you select the Warning beep check box, the system will sound a tone whenever it displays a warning message. If you remove the check from the Warning beep check box, the system will not sound a tone when displaying a warning message.

The Spooler Object

The Spooler object in the System Setup folder can be used to specify options for the print spooler. These options include:

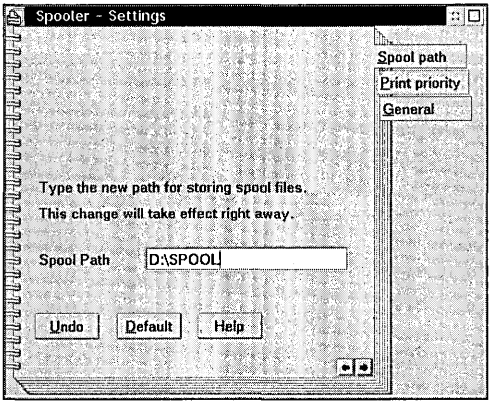
- **Spool path** — Assigns the path used to hold spooler files
- **Print priority** — Specifies the relative priority of the print spooler

These characteristics can be specified through the Settings Notebook of the Spooler object. To open the Settings Notebook, just double-click on the Spooler object in the System Setup folder.

Spool Path page

The Spool Path page allows you to specify the drive and path on which you want to create print spooler files (see Figure 7-23).

Figure 7-23.
Spooler Object — Spool Path Page



Type the new path for storing spool files.
This change will take effect right away.

Spool Path D:\SPOOL

You can specify the path for spool files in the Spool path entry box. The default is the \SPOOL directory on the boot drive.

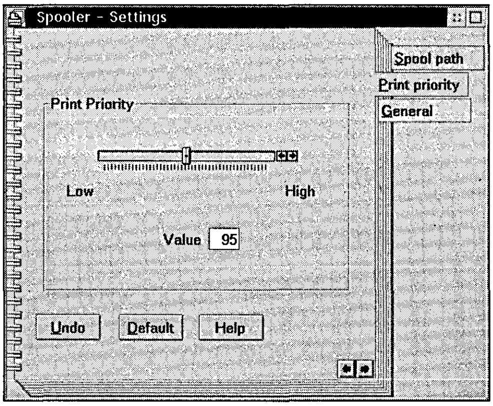
You might want to change the print spool path to another drive if the boot drive is almost full. For example, if the swapper file is on the boot drive and you often use an application program that uses a lot of virtual memory, the swapper file might grow to almost fill the boot partition. In this case, you might want to move the spool files to another drive so that they don't take space away from the swapper. Also, if the swapper file has taken up most of the partition, you might not have enough room for spool files. This means that you probably won't be able to print large amounts of data unless you move the spool files to another drive.

Warning

You have to close the Spooler object before the change to the spool path takes effect.

Print Priority page

The Print Priority page allows you to specify the relative priority of the print spooler (see Figure 7-24). The priority varies from 1 to 189. The default value is 95.



Value 95

Figure 7-24.
Spooler Object — Print Priority Page

If application programs slow down too much when files are printing, you can move the Print Priority slider to the left to lower the priority of the print spooler. If your printer doesn't run at full speed when printing a file, you can move the Print Priority slider to the right to raise the priority of the print spooler.

Note

You will have to experiment to find the best setting for the Print Priority. If you make the priority of the print spooler too high, the performance of the Workplace Shell and application programs may be degraded. On the other hand, if you set the priority of the print spooler too low, it may take too long to print files.

Pop-up menu items

You can specify one other option for the Spooler object — whether or not the print spooler is active. This option is specified by selecting an item from the pop-up menu for the Spooler object. The items you may select are

- ✗ **Disable spooler** — If you disable the print spooler, any program that generates printer output communicates directly with the printer driver. If several programs are running simultaneously and they all try to print, their output may be intermixed since the print spooler is not keeping their output separate (see Chapter 14).

Warning

Disabling the print spooler is not recommended. The only reasons you may want to disable the print spooler are if you are printing data that has a high security classification and you don't want the data to appear in a spool file, or if the application program you are using generates many small print jobs instead of one large job.

- **Enable spooler** — This allows the print spooler to control printing and ensures that output from each program is kept together.

The WIN-OS/2 Setup Object

The WIN-OS/2 Setup object in the System Setup folder can be used to specify system default options for Windows programs. These options include:

- **3.1 Session** — Specifies the characteristics of Windows programs
- **Data Exchange** — Specifies the sharing characteristics for dynamic data exchange and the clipboard

These characteristics can be specified through the Settings Notebook of the WIN-OS/2 Setup object. To open the Settings Notebook, just double-click on the **WIN-OS/2 Setup** object in the System Setup folder.

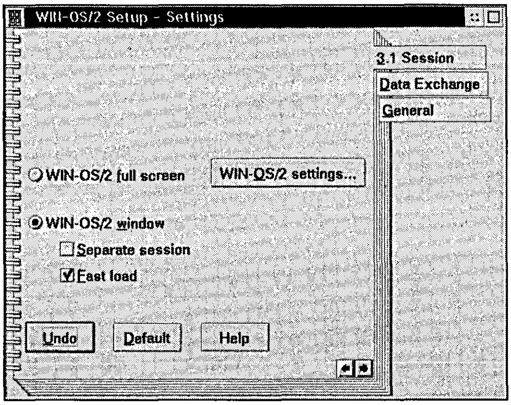
3.1 Session page

The 3.1 Session page allows you to specify the system default values for Windows programs (see Figure 7-25). These values are used for all Windows programs started after the changes are made; they do not affect any currently running Windows programs. They also do not affect any Windows programs started from a Program Reference object where you have specified the characteristics for the Windows program.

- Select the **WIN-OS/2 full screen** radio button if you want all Windows programs to be started in a full-screen session. If you specify this option, the Windows program will occupy the entire screen and each program will be started in an independent Windows environment. You can switch to the Desktop or other windows by pressing **Ctrl-Esc** to display the Task List, from which you can select another window to display.
- Select the **WIN-OS/2 window** radio button if you want all Windows programs to run in the same session as the other windows on your Desktop. If you specify this option, the Windows program will occupy a window on the Desktop, just

like a Presentation Manager application program or a windowed OS/2 or DOS program. You can then select whether Windows programs will be started in independent Windows environments or in a single environment.

Figure 7-25. WIN-OS/2 Setup Object — 3.1 Session Page



○ WIN-OS/2 full screen

● WIN-OS/2 window

□ Separate session

☑ Fast load

Separate session check box

If you select the Separate session check box, each Windows program will be started in an independent Windows environment. This means that each program will start in its own Virtual DOS Machine and cannot be affected by other Windows programs. If the program causes a catastrophic failure of the Windows environment, only that program will terminate. However, it takes longer to start a Windows program in a separate session because the Windows subsystem has to be loaded and initialized for each program.

If you remove the check from the Separate session check box, all Windows programs will be started in a single Windows environment. This means that all programs will run in the same Virtual DOS Machine and can be affected by other Windows programs. If one program causes a catastrophic failure of the Windows environment, all programs running in that Virtual DOS Machine will be terminated. The main reason for choosing to load all Windows programs in the same session is that it takes much less time to start Windows programs after the first one because the Windows environment has to be loaded and initialized only once.

Note

Determining which session setting you prefer is not as simple as it seems. Checking the Separate session check box does force each Windows program to run in a separate Windows environment. But removing the check from the Separate session check box does not guarantee that all Windows programs will run in the same Windows environment.

A Windows program can be started in the same environment as an already-running Windows program only if both programs have the same value for the WIN_RUN_MODE. If a Standard mode program is running and you try to start an Enhanced mode program, the programs cannot run in the same Windows environment. The second program will start in its own Windows environment even if the Separate session check box is not selected.

Fast load option

Your choices don't end here. You also have a Fast load option. If you select this option, a dummy Windows program is automatically started when the Workplace Shell starts. The program, which is invisible to you, causes the Windows environment to be loaded and initialized so that all other Windows programs can be loaded quickly. However, this procedure increases the time it takes to initialize the Workplace Shell, so you have to trade off the additional start-up time against the time you save in starting Windows programs while the system is running. Also, only Windows programs that specify the WIN-OS/2 window option and the system default value for WIN_RUN_MODE can be added to the Fast load Windows environment.

If you normally run Windows programs like word processors or spreadsheets, you can decrease the time it takes to load these programs by specifying the **Fast load** option and by making sure each of your Windows programs runs in the same session. However, you will have to wait a little longer for the system to start. Only you can decide whether this trade-off makes sense.

Sound considerations

If you configure your Windows environment to include a Windows sound device driver, there is another factor you have to consider when trying to decide whether to use Fast load and whether to start Windows programs in the same or separate sessions. Most of the sound device drivers available for OS/2 Warp do not allow the use of your sound card to be shared between multiple DOS programs or between DOS and OS/2 programs.

This limitation exists within the sound device drivers, not the sound cards. For example, the device drivers for the Crystal Semiconductor sound cards support sharing the sound card between DOS and OS/2 sessions. Also, the device drivers for the MediaVision PAS 16 cards allow you to use the SoundBlaster-compatible part of the sound card with DOS programs while you use the PAS part of the sound card for OS/2 sessions. However, the device drivers for the SoundBlaster cards do not allow sharing of the sound card between different sessions.

Each Windows environment is viewed as a separate DOS program, and each Windows session will attempt to gain use of your sound card, but only the first session will succeed. If you specify **Fast load**, the sound card will be allocated to the hidden Windows environment. Any Windows program started in the same session will have access to the sound card. However, any Windows program loaded as a full-screen session or a separate windowed session will not be able to use the sound card.

In fact, you may get some error messages about the sound card not being installed when trying to load a second Windows program. Even worse, for most sound device drivers, once a DOS program (in this case, the preloaded Windows environment) has the sound card allocated, no other program, including the OS/2 system sounds, can use the sound card. This means that no DOS program will be able to use the sound card, your system sounds will not play, the Audio Player won't be able to play sound files, the Video Player won't be able to play the audio track of Ultimotion movies, the CD Player won't be able to play CDs, and so on.

Tip

To circumvent this problem, you could disable the sound support in the Windows environment, but then you wouldn't be able to run any multimedia Windows programs. A better solution is to configure your shared Windows environment without sound support and to start Windows programs that require sound in a separate Windows environment configured for sound. This allows your OS/2 programs to use the sound card until you start a Windows session that needs the sound card. You would have to limit your use of the sound card to one program at a time.

You can do this by following these steps:

1 Configure your Windows environment without the Windows sound device driver installed. Save the Windows SYSTEM.INI file as NOSOUND.INI.

2 Configure your Windows environment with the Windows sound device driver installed. Save the Windows SYSTEM.INI file as SOUND.INI.

3 In your AUTOEXEC.BAT file, copy the NOSOUND.INI file to SYSTEM.INI in your Windows directory. This will cause most Windows programs to start in an environment that doesn't have access to the sound card.

4 Copy your AUTOEXEC.BAT file to a file named AUTOMM.BAT. In the AUTOMM.BAT file, replace the statement that copies the NOSOUND.INI file to SYSTEM.INI with a statement that copies SOUND.INI to SYSTEM.INI.

5 In the DOS settings for any Windows program requiring sound, specify **AUTOMM.BAT** in the DOS_AUTOEXEC option and either select the WIN-OS/2 full-screen option, or select the **WIN-OS/2** window option and the **Separate session** option on the Session page. This will cause the specified Windows program to start in a separate Windows environment that has access to the sound card. You will be able to run only one of these special programs at a time, but you will be able to run multiple Windows, DOS, and OS/2 programs that don't need sound support simultaneously.

Developer's Desktop

On my system at home, I selected the **Fast load** option since I often run a Windows word processor, a Windows calendar program, and a Windows financial program. It takes about 20 seconds longer to start my system because of the Fast load option, but Windows programs start much faster. I have most of my Windows sessions configured without sound support, since I don't need sound in my "production" programs. I have a specially configured SYSTEM.INI file that includes the sound device drivers and sets the display resolution to 640 × 480 × 256 (which is the best resolution for most multimedia Windows programs). I use this specially configured file for multimedia Windows programs, like games and multimedia reference programs. I only run one of these multimedia programs at a time, so I don't lose any functionality.

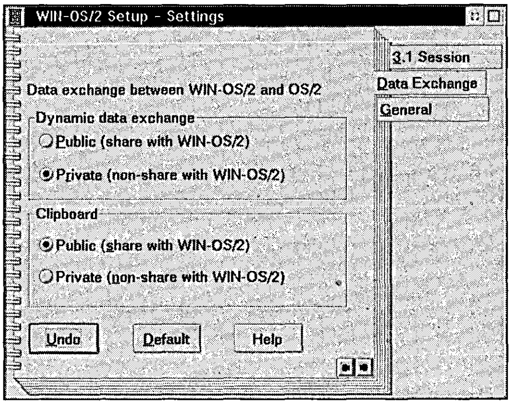
The **WIN-OS/2 settings...** push button allows you to specify the DOS and Windows settings to be used when a Windows program is started. See Chapter 16 for an explanation of how to use these settings. If you want to use Adobe Type Manager (ATM) fonts in your Windows programs, you have to specify the WIN_ATM setting as **On** either on the 3.1 page (to activate ATM for all Windows programs) or in the Settings Notebook for the specific programs for which you want to use ATM fonts.

Tip

Specifying the WIN_ATM as **On** increases the time taken to initialize the Windows environment, so you may want to specify this option only for programs that actually use ATM fonts. On the other hand, if you are going to use a shared Windows environment (including the one established by the Fast load option), the first Windows program that is loaded into the environment must specify the WIN_ATM option as On if any of the programs in that shared environment intend to use ATM fonts.

Data Exchange page

The Data Exchange page allows you to specify whether data is shared between the Windows environment and other environments (see Figure 7-26).



WIN-OS/2 Setup - Settings

Data exchange between WIN-OS/2 and OS/2

Dynamic data exchange

○ Public (share with WIN-OS/2)

● Private (non-share with WIN-OS/2)

Clipboard

● Public (share with WIN-OS/2)

○ Private (non-share with WIN-OS/2)

Figure 7-26. WIN-OS/2 Setup Object — Data Exchange Page

Dynamic data exchange

The Dynamic data exchange option specifies whether data can be shared between a Windows program and an OS/2 or DOS program using the Dynamic Data Exchange protocol. If **Public** is selected, data can be exchanged between any two programs. If **Private** is selected, data can be exchanged only between two Windows programs running in the same Windows environment — a Windows program can't exchange data with a DOS program, an OS/2 program, or another Windows program running in a different Windows environment.

Clipboard

The Clipboard option specifies whether data can be shared between the Windows clipboard and the OS/2 clipboard.

✗ **Public** — If **Public** is selected, the Windows clipboard is public, and data can be exchanged between the Windows clipboard and the OS/2 clipboard. This means that you can copy data to the clipboard from a Windows program and paste the data into an OS/2 program. You can also copy data to the clipboard from an OS/2 program and paste it into a Windows program. Of course, for the cut-and-paste operation to work from one type of program to another, both programs must understand a common data format.

For example, if the Windows program only understands how to copy graphics images to the clipboard and the OS/2 program only understands how to paste text from the clipboard, they will not be able to communicate via the clipboard, even if the Windows clipboard is public.

- **Private** — If **Private** is selected, the data copied to the Windows clipboard cannot be accessed by OS/2 programs or by Windows programs running in a different Windows environment. Data can still be copied to the clipboard by one Windows program and pasted into another Windows program running in the same Windows environment.

The Country Object

The Country object in the System Setup folder can be used to specify the country-specific information for your system. This information includes

- **Country** — What country you live in, what system of measurement you use, and what character separates elements of a list
- **Time** — The format you want used when the system or an application program displays the time of day
- **Date** — The format you want used when the system or an application program displays a date
- **Numbers** — The format you want used when the system or an application program displays numeric information

This information can be specified through the Settings Notebook of the Country object. To open the Settings Notebook, just double-click on the **Country** object in the System Setup folder. You will have to close the **Country** object Settings Notebook for the changes to take effect.

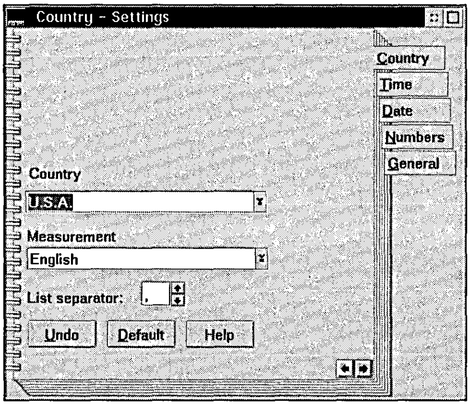
Tip

If you have an application program running when you change the Country settings, you might have to terminate the application program and restart it in order to see the changes take effect. For open views of objects in the Workplace Shell, you will have to close the view, then reopen it in order to see the changes. For example, if you change the date format and have the System Clock open, you will not see the change in the format of the date on the System Clock until you close the system clock and then reopen it.

Country page

The Country page allows you to specify information about the country you live in (see Figure 7-27).

Figure 7-27.
Country Object — Country Page



Country
U.S.A.

Measurement
English

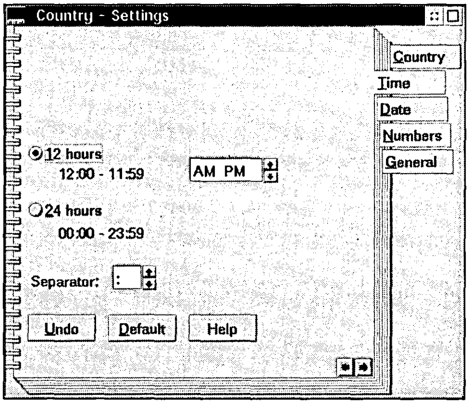
List separator: ,

- You can select the country you live in from the Country combination box. This automatically sets all of the other options to the values that are commonly used in the selected country. You can override any of these values if you want.
- You can select the measurement system to be used from the Measurement combination box. Your options are English (inches), Metric (centimeters), Points ($^{1}/_{72}$ of an inch), or Picas ($^{1}/_{6}$ of an inch). The default value is the measurement system commonly used in your country.
- You can select the character used to separate numbers in a list from the List separator combination box. You can choose either the comma (,), colon (:), or semicolon (;). The default value is the character commonly used in your country.

Time page

The Time page can be used to specify the format of the time of day when it is displayed by the system or by an application program (see Figure 7-28). Note that this will only work if the application program is coded to correctly query the time format before displaying the time of day. If the time format displayed by an application program is not of the form you selected, contact the supplier of the application program and tell it about the problem.

Figure 7-28.
Country Object — Time Page



Country - Settings

● 12 hours
12:00 - 11:59 AM PM

○ 24 hours
00:00 - 23:59

Separator: :

Select the **12 hours** radio button if you want time displayed in 12-hour format. In this format, the hours are displayed as 1 (indicating 1 a.m. or 1 p.m.) through 12 (indicating noon or midnight). If you select this format, you can also choose the characters you want displayed following the time, if any, to show whether the time indicates a.m. or p.m.:

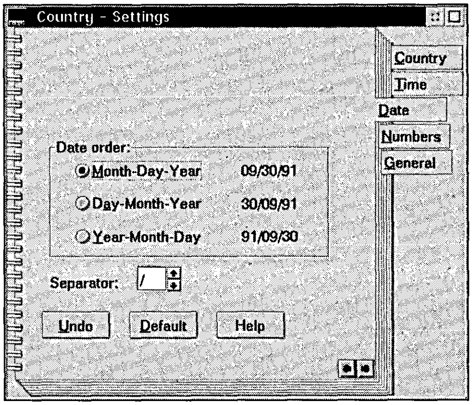
- Choose **(none)** if you want the time to be displayed without any indication of a.m. or p.m.
- Choose AM PM if you want the time to be followed by either AM or PM (in uppercase characters).
- Choose **am pm** if you want the time to be followed by either am or pm (in lowercase characters).

Select the **24 hours** radio button if you want time displayed in 24-hour format. In this format, the hours are displayed as 0 (indicating midnight) through 23 (indicating 11 p.m.).

You can also select the character used to separate the hours from the minutes from the Separator spin button. Your choices are the colon (:) or the period (.).

Date page

The Date page can be used to specify the format of a date when it is displayed by the system or by an application program (see Figure 7-29). Note that this will only work if the application program is coded to query the date format correctly before displaying a date. If the date format displayed by an application program is not of the form you selected, contact the supplier of the application program and tell it about the problem.



Date order:

● Month-Day-Year 09/30/91

○ Day-Month-Year 30/09/91

○ Year-Month-Day 91/09/30

Separator: /

Figure 7-29. Country Object — Date Page

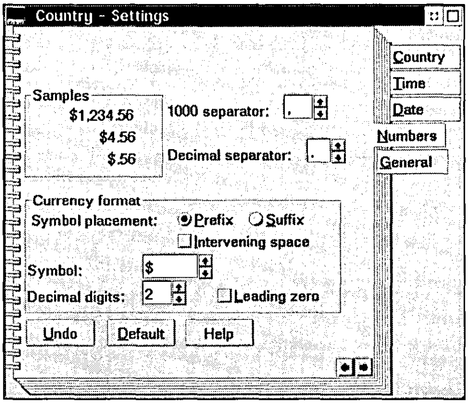
You can select the order in which you want parts of the date displayed from the Date order group. Your options are **Month-Day-Year** (the American standard), **Day-Month-Year** (the standard used by most of the rest of the world), or **Year-Month-Day** (a format that can be easily sorted into ascending or descending date order).

You can choose what character you want to separate the parts of the date with the **Separator** spin button. Your choices are dash (-), period (.), the letter *x,* or slash (/).

Numbers page

The Numbers page can be used to specify the format of numbers displayed by the system or by an application program (see Figure 7-30). Note that this will only work if the application program is coded to query the number format correctly. If numbers displayed by an application program are not of the form you selected, contact the supplier of the application program and tell it about the problem.

You can use the 1000 separator spin button to select the character that is used to separate groups of three digits to the left of the decimal point. This character is used to the right of the thousands digit, the millions digit, and so on. You can choose the comma (,), period (.), quote (‘), semicolon (;), or blank.



Samples
$1,234.56
$4.56
$.56

1000 separator: ,

Decimal separator: .

Currency format

Symbol placement: ● Prefix ○ Suffix

□ Intervening space

Symbol: $

Decimal digits: 2 □ Leading zero

Figure 7-30.
Country Object — Numbers Page

You can use the Decimal separator spin button to select the character that is used to separate a decimal fraction from the units digit. You can choose either the period (.) or the comma (,).

You would normally not specify the same character to be used for both the 1000 separator and the Decimal separator, because that would make it more difficult to read the number. In the United States, the 1000 separator is normally a comma and the decimal separator is normally a period, so a number might be displayed as "1,234,567.89." In other countries, the 1000 separator is the period and the decimal separator is the comma, leading to a display of the same number as "1.234.567,89." The choice is yours.

You can specify values in the Currency format group to select the way you want monetary values displayed:

- The symbol selected from the Symbol spin button is the currency symbol. It will be displayed in front of the number if you select the **Prefix** radio button or after the number if you select the **Suffix** radio button; the custom depends on the country you live in.
- If you check the Intervening space box, there will be a blank character between the currency symbol and the number.
- You can select the number of digits to be displayed following the decimal separator character from the Decimal digits spin button (you can choose 0, 1, 2, or 3 decimal digits).

- Finally, if you check the Leading zero check box, all monetary values will be displayed with at least one digit to the left of the Decimal separator character. A value with a zero unit value will be displayed as the digit 0 followed by the Decimal separator character and the decimal digits (for example, $0.29 for a value of 29 cents). If you remove the check from this box, a value with a zero unit value will not have any digits displayed to the left of the Decimal separator character (for example, $.29 for a value of 29 cents).

The System Clock Object

The System Clock object in the System Setup folder can be used to display the current date or time of day. The System Clock object's Settings Notebook can be used to specify how you want the clock to look. The characteristics you can modify include

- **Date/Time** — Allows you to set the current date and time
- **View** — Allows you to specify how the clock looks
- **Alarm** — Notifies you when the specified time is reached (like an alarm clock)

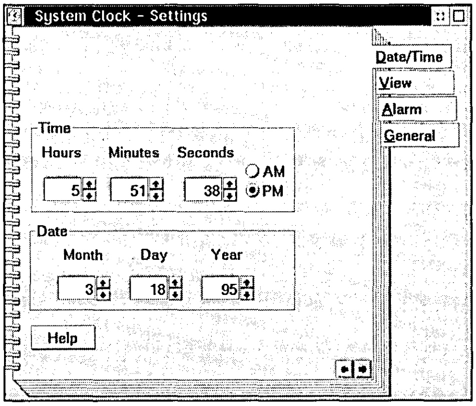
These characteristics can be specified through the Settings Notebook of the System Clock object. To open the Settings Notebook, place the mouse pointer on the **System Clock** object in the System Setup folder, press mouse button 2, and select the Settings item from the pop-up menu. If the System Clock object is open (that is, the clock is currently displaying the date or time), you can point the mouse anywhere within the System Clock object, press mouse button 2, and select the **Settings** item from the pop-up menu.

Tip

It will be useful to open the System Clock object while you are setting it up, so you can see the effect of your changes. To open the System Clock, just double-click on the **System Clock** object in the System Setup Folder.

Date/Time page

The Date/Time page can be used to set the date or time of the system clock. The exact format of the Date/Time page depends on the options specified for date and time format in the Country object. The sample Date/Time page shown in Figure 7-31 shows the results when the 12-hour time format is being used and the date order is Month-Day-Year.



Time

Hours 5 Minutes 51 Seconds 38 ○ AM ● PM

Date

Month 3 Day 18 Year 95

Figure 7-31. System Clock Object — Date/Time Page

To set the current time of day, change the spin buttons (and the AM and PM radio buttons, if present) in the Time group. To set the current date, change the spin buttons in the Date group.

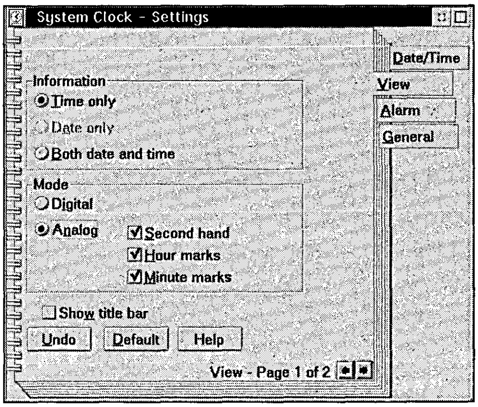
When you change the values on the Date/Time page, the changes are immediately reflected in the open view of the System Clock object. Also, the changes are written into the hardware clock on your computer, so you don't have to update the date and time every time you start your system. However, if you are using a facility that automatically synchronizes your hardware clock with another time base (for example, if you are using the IBM LAN Requestor, it automatically resets your clock to the time and date stored in the LAN Server that you log onto), the changes you make on the Date/Time page might not be maintained the next time you restart your system.

View pages

The View pages can be used to specify how the System Clock looks when it is open. Once you have the System Clock customized to look the way you want to see it, you can have it automatically started every time the system is booted by creating a shadow of the System Clock object in the Startup folder.

Specifying the form of the System Clock

The first View page can be used to specify the form of the System Clock (see Figure 7-32).



System Clock - Settings

Information

● Time only

○ Date only

○ Both date and time

Mode

○ Digital

● Analog ☑ Second hand ☑ Hour marks ☑ Minute marks

□ Show title bar

Figure 7-32. System Clock Object — View Page 1

Select one of the radio buttons in the Information group to specify whether you want to display the **Time only**, the **Date only**, or **Both date and time.**

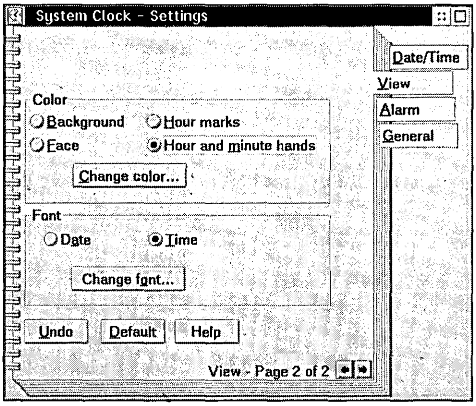
Select one of the radio buttons in the Mode group to specify the form of clock you want to see.

- Select the **Digital** radio button if you want the time or date displayed in digital form. If you request that both the date and time be displayed, the time is displayed above the date.
- Select the **Analog** radio button if you want the time displayed like an analog clock. The date, if requested, is displayed digitally in the middle of the analog clock (if there is room).

  If you select **Analog**, the time is always displayed as a clock face. You can optionally check the **Second hand** box to display a second hand, check the **Hour marks** box to display square marks for each hour, or check the **Minute marks** box to display lines for each minute between the hour. You can select any combination of these check boxes.
- You can specify that a title bar should be displayed at the top of the System Clock window by checking the **Show title bar** box. If you remove the check from this check box, the System Clock window only has a sizing border around it. If the title bar is not displayed, you can resize the clock window, but you can't move it around on the screen. In order to customize the clock, you should display the title bar, move the clock to the location on the screen where you want it to be, and select the digital or analog clock. You can then turn off the title bar display and make the clock window the size you want it to be.

The second View page can be used to specify the fonts and colors for the various parts of the System Clock (see Figure 7-33).

Figure 7-33. System Clock Object — View Page 2



System Clock - Settings

Color

○ Background ○ Hour marks

○ Face ● Hour and minute hands

Font

○ Date ● Time

View - Page 2 of 2

Specifying colors for the System Clock

The Color group is used to specify the colors for parts of the System Clock. To change the colors, you select the radio button corresponding to the part of the clock you want to change, then press the **Change color...** push button. This opens an Edit Color dialog to allow you to specify the color you want for the selected clock element.

- The **Background** radio button is used to change the color of the background of the clock. This is the area outside the clock face for the analog clock, or the background color of the clock window for the digital clock.

Tip

If you are displaying the digital clock, you should specify a nondithered color for the background; otherwise, it might be difficult to read the date or time values. You also want to choose a background color that contrasts well with the color you select for the date or time characters.

- The **Face** radio button is used to change the color of the face of the analog clock. This includes the background for the date, if the clock is displaying the date. This color is not used for the digital clock.

Tip

If you are displaying the date, you might want to specify a nondithered color to make it easier to read the date, and you should probably choose a light color because the date is always displayed using black characters.

- The **Hour marks** radio button is used to change the color of the hour marks on the analog clock. This color is not used for the digital clock.

- ✘ The **Hour and minute hands** radio button is used to change the color of the hour and minute hands on the analog clock and the color of the numbers on the digital clock.

Note

The color of the date on the analog clock is always black and cannot be changed.

You can also use the Color Palette to change the background colors on the System Clock. If you select a color on the Color Palette and drop it on the digital clock's window, you will change the background color to the selected color. If you select a color on the Color Palette and drop it on the analog clock's window, you will change the background color and the clock face color.

Uncensored

The analog clock supports only solid colors for the background and clock face, so the colors might not be changed to exactly the colors you selected. If the selected color is a dithered color, the color will be changed to the solid color that is "closest" to the selected color, but this color might not be even remotely related to the color you selected. For example, if you select a light pink color, the closest solid color might be light gray. You will have to experiment until you get the color you want.

Specifying fonts for the System Clock

Note

The Fonts group is used to specify the fonts for the date and time. To change the fonts, you select the radio button corresponding to the type of text you want to change, then press the **Change font...** push button. This opens an Edit Font dialog, which allows you to specify the font for the selected clock element.

- ✘ The **Date** radio button is used to change the font in which the date (on either the analog or digital clock) is displayed.
- ✘ The **Time** radio button is used to change the font in which the time on the digital clock is displayed. This font is not used on the analog clock.

Uncensored

The font you specify should be a scalable font such as Helvetica. (See the discussion of fonts in Chapter 6.) If you specify a bitmapped font, like Helv, that font is ignored and the System Clock substitutes a scalable font. Likewise, the System Clock will ignore the size you specified for the font and automatically size the time or date to fill the System Clock window. Any emphasis options you specified will also be ignored. Only the name of the font and its style will be used.

You can also use the Font Palette to change the fonts on the System Clock. If you select a font on the Font Palette and drop it on the clock's window, you will change both the date and time font to the selected font.

Uncensored

Both the date and time fonts are changed regardless of whether you selected the analog or the digital clock.

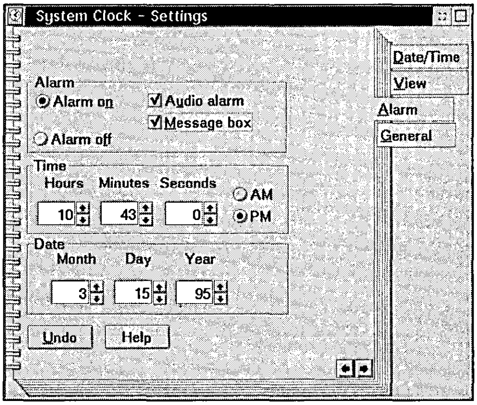
Developer's Desktop

I like to keep the System Clock on my screen all the time. I customized the System Clock to be a digital clock displaying only the time, with a blue background and white digits. I use the Helvetica Bold font for the time, since I think it is more readable than the default font. I made the clock as small as possible, positioned it in the lower-left corner of the screen, and removed the title bar. I then put a shadow of the System Clock object in the Startup folder, so the clock always appears on my screen at start-up. It is small enough that if I size my application program windows to be just a little bit smaller than full screen, I can always see the clock.

Alarm page

The Alarm page can be used to set a time for the System Clock to sound an alarm or display a message, like an alarm clock (see Figure 7-34).

Figure 7-34. System Clock Object — Alarm Page



System Clock - Settings

Alarm
- ● Alarm on
- ○ Alarm off
- ☑ Audio alarm
- ☑ Message box

Time
- Hours: 10
- Minutes: 43
- Seconds: 0
- ○ AM
- ● PM

Date
- Month: 3
- Day: 15
- Year: 95

Select the **Alarm on** radio button to activate the alarm clock feature. Select the **Alarm off** radio button to turn off the alarm clock feature. If the alarm clock feature is turned on, you must indicate how you want to be notified:

- ✗ Check the **Audio alarm** box for the system to play the Alarm Clock sound file (specified in the Sound object) when the alarm time is reached.

- ✗ Check the **Message box** if you want the system to display a message box when the alarm time is reached. You can check both check boxes if you want to both see the message box and hear the sound file played when the alarm time is reached. If you don't check either check box, you will not be notified when the alarm time is reached.

Note

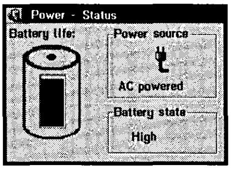
Set the spin buttons in the Time group to the time at which you want to be notified. Don't forget to set the **AM/PM** option (if you are not using 24-hour time) correctly. Also set the spin buttons in the Date group to the date on which you want the alarm to sound. If you don't set the date correctly, you will not be notified when the alarm time is reached. The date doesn't default to today's date — it defaults to whatever date you set last.

The Power Object

The Power object is installed in the System Setup folder if your computer supports Advanced Power Management (APM). This includes laptop and notebook computers and computers that support the Rapid Resume feature, such as the IBM PS/1 and Aptiva computers.

The Power object can be used to monitor the power status of your computer. The Settings Notebook for the Power object can be used to specify the characteristics of the Advanced Power Management support and the power status display.

When you open the Power object, you see the power status display, as shown in Figure 7-35. The Battery life graphic is a pictorial representation of the charge state of your computer's battery. The rectangle inside the picture of the battery indicates how charged the battery is. The higher the filled-in area within the rectangle, the more charge your battery has. When this display gets near the bottom, you should replace the battery with a fresh battery or plug in the computer's AC power cord.



Power source: AC powered

Battery state: High

Figure 7-35. Power Object — Power Status Display

The Power source box tells you what the current power source is for your computer:

- **Battery powered** — Your computer is running on its internal battery.
- **AC powered** — Your computer is running on power supplied through the AC power cord.

Note

If your computer's AC power cord is plugged in but the Power source is shown as Battery powered, you may have plugged your computer into an AC outlet that is switched off, or the power cord may not be properly plugged into the computer or the outlet. Verify the connection between the computer and the AC outlet and make sure the outlet is turned on. If the Power source display continues to say that your computer is powered by its battery, plug the cord into a different AC outlet. (Of course, if you turned off the automatic refresh of the power status display, the Power source indication could simply be out of date.)

The Battery state box gives an indication of the state of your computer's battery. The values you see here are

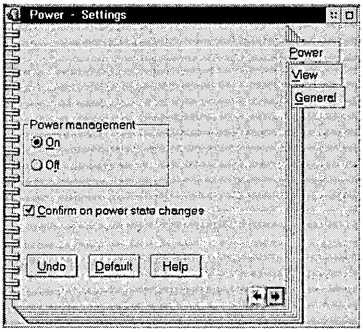
- **High** — The battery contains more than enough power to run your computer.
- **Low** — The battery is running out of power and should be replaced or recharged.
- **Critical** — The battery is almost out of power and soon will not be able to power your computer. The battery should be replaced or the AC power cord should be plugged in immediately.
- **Charging** — The battery is being charged through the AC power cord.
- **Unknown** — The computer cannot determine the state of the battery.

The Settings Notebook for the Power object has two unique pages: the Power page and the View page.

Power page

The Power page (see Figure 7-36) allows you to specify whether or not Advanced Power Management is active and whether or not the system asks for confirmation when you request that your computer's operation be suspended.

Figure 7-36.
Power Object —
Power Page



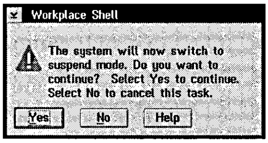
Power management
● On
○ Off

☑ Confirm on power state changes

The Power Management group is used to specify whether or not the Advanced Power Management feature is active. Select the **On** radio button if you want APM to be active, or select the **Off** radio button to turn APM off.

The Confirm on power state changes check box is used to specify whether or not the system is to display a message box (see Figure 7-37) whenever you select the **Suspend** item from the Power object's pop-up menu.

Figure 7-37.
Power Object —
Suspend
Message Box

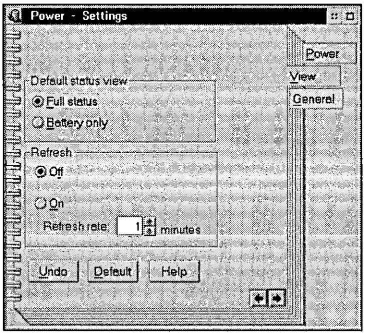


The system will now switch to suspend mode. Do you want to continue? Select Yes to continue. Select No to cancel this task.

View page

The View page (see Figure 7-38) allows you to customize the Power object's status display.

Figure 7-38.
Power Object —
View Page



Default status view
● Full status
○ Battery only

Refresh
● Off
○ On
Refresh rate: 1 minutes

The Default status view group can be used to specify whether you see the entire status display or just a part of it:

- If you select the **Full status** radio button, the Power object's status display will include the graphic showing the current charge state of your computer's battery, a box showing the current power source for your computer, and a text indication of the current state of your battery.
- If you select the **Battery only** radio button, the Power object's status display will only show the graphic of the state of your computer's battery.

Note

Changing your selection in the Default status view group doesn't affect the open view of the Power object. You have to close the Power object and reopen it in order for your change to take effect.

The Refresh group can be used to specify whether or not the Power object's status display is automatically updated and, if so, how often.

- If you select the **Off** radio button, the Power object's status display is not automatically updated. You have to select the **Refresh now** item from the Power object's pop-up menu whenever you want to see the current state of your system.
- If you select the **On** radio button, the Power object's status display is automatically updated. You can specify the number of minutes between updates by changing the value in the **Refresh rate** spin button.

Tip

You can also turn the automatic update of the power status display on or off by selecting the **Refresh** option on the Power object's pop-up menu. This displays a list from which you can choose **On** or **Off**.

Tip

If you normally use your laptop or notebook computer with its battery, you might want to have the power status display on your Desktop all the time. You can follow these steps to be able to continuously monitor the state of your computer's battery:

1. Make sure that APM is on by selecting the **On** radio button on the Power page of the Power object's Settings Notebook.
2. Specify the **Battery only** option on the View page. This will make the power status display as unobtrusive as possible.

3 Specify the refresh option as **On** and set the refresh rate to 1 minute. This allows the system to check your computer's battery state every minute and update the display. You can set the refresh rate to a more frequent value if you want, but it is not necessary.

4 Open the Power object and position it where you want it on the Desktop.

5 Put a shadow of the Power object into the Startup folder.

Now, whenever you start OS/2, the Power object will automatically be displayed in its current location on the Desktop.

Suspend and resume

If your computer supports Advanced Power Management, it supports the Suspend and Resume feature.

Suspending operations

When you invoke the Suspend function, your computer is put into a low power usage state. The computer is not totally turned off – it simply powers down the components that use the most power. You cannot use the computer when it is in the suspended state. However, the programs you had running (including OS/2 itself) are put into a state of suspended animation and can be reinstated by invoking the Resume function.

You can invoke the Suspend function by selecting the **Suspend** item from the Power object's pop-up menu. The computer screen will become blank while the computer is suspended.

Resuming operations

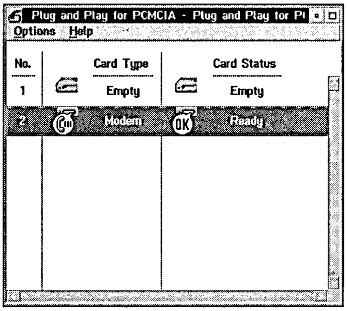
The method for resuming normal operations varies from one computer to another. Consult the operating manual for your computer for the procedure you have to perform to resume normal operations.

On some computers, closing the cover automatically invokes the Suspend function, and opening the cover resumes normal operation. On other computers, you have to press a designated key on the keyboard to resume normal operation of your system.

The Plug and Play for PCMCIA Object

The Plug and Play for PCMCIA object is installed in the System Setup folder if your computer contains one or more PCMCIA slots. This includes most laptop and notebook computers.

The Plug and Play for PCMCIA object (see Figure 7-39) can be used to display the contents of your computer's PCMCIA slots and to specify what actions are to be taken when you insert or remove a PCMCIA card.

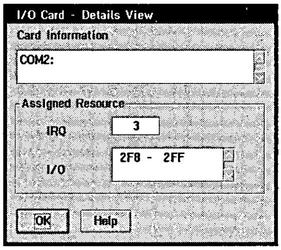


No.	Card Type	Card Status
1	Empty	Empty
2	Modem	Ready

Figure 7-39. Plug and Play for PCMCIA Display Window

The Plug and Play for PCMCIA object must be open in order for it to perform any of its automatic actions. If you want the system to monitor the PCMCIA slots and automatically start programs when you insert PCMCIA cards into the slots, put a shadow of the Plug and Play for PCMCIA object into the Startup folder. This ensures that the object is automatically opened when the system starts and will be ready to monitor for PCMCIA card activity.

If you have one or more PCMCIA cards installed in your computer, you can get information about each card by double-clicking on the card in the display window. This will present a dialog similar to the one shown in Figure 7-40, which is the details display for a PCMCIA I/O card, such as a modem.



Card Information

COM2:

Assigned Resource

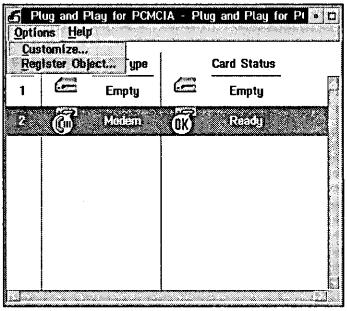
IRQ 3

I/O 2F8 - 2FF

Figure 7-40. Plug and Play for PCMCIA Object — I/O Card Dialog Box

Uncensored

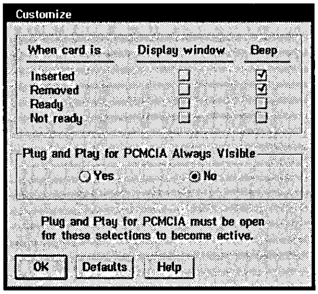
The Plug and Play for PCMCIA object is actually a Presentation Manager program; it is not a Workplace Shell object. In order to customize this object, you select its **Options** menu (see Figure 7-41), then select either the **Customize...** or **Register Object...** options.



	ype	Card Status
1	Empty	Empty
2	Modem	Ready

Figure 7-41. Plug and Play for PCMCIA Object — Options Menu

The Customize... option displays the Customize dialog box (see Figure 7-42), which can be used to specify what actions are to be taken when the state of one of the PCMCIA slots changes. You can specify the actions to be taken when you insert or remove a PCMCIA card or when a PCMCIA card changes its state to Ready or Not Ready. You can have the system sound a tone on the speaker or display the Plug and Play for PCMCIA object's window.



When card is	Display window	Beep
Inserted	□	☑
Removed	□	☑
Ready	□	□
Not ready	□	□

Plug and Play for PCMCIA Always Visible

○ Yes ● No

Plug and Play for PCMCIA must be open for these selections to become active.

Figure 7-42. Plug and Play for PCMCIA Object — Customize Dialog Box

Note

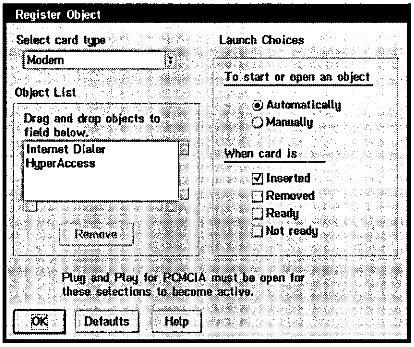
The display window options have an effect only when the Plug and Play for PCMCIA object is hidden or minimized. If the display window is already visible, no action takes place when the state of a PCMCIA card changes.

The radio buttons in the Plug and Play for PCMCIA Always Visible group specify how the object's icon behaves when it is minimized to the Desktop:

- ✘ If the Yes radio button is selected, the minimized icon floats above any window that attempts to cover it, so the icon is always visible on the screen.

✗ If the **No** radio button is selected, the minimized icon can be covered by another window.

The Register Object... option displays the Register Object dialog box (see Figure 7-43), which can be used to specify what programs are to be started when the state of a PCMCIA card changes.



Select card type: Modem

Object List

Drag and drop objects to field below.

Internet Dialer
HyperAccess

Launch Choices

To start or open an object

● Automatically
○ Manually

When card is

☑ Inserted
□ Removed
□ Ready
□ Not ready

Plug and Play for PCMCIA must be open for these selections to become active.

Figure 7-43. Plug and Play for PCMCIA Object — Register Object Dialog Box

You can register different actions for each different type of PCMCIA card. You select the card type that you want to configure by choosing an item from the Select card type combination box. Once you select the type of card to be configured, all other options apply to that card.

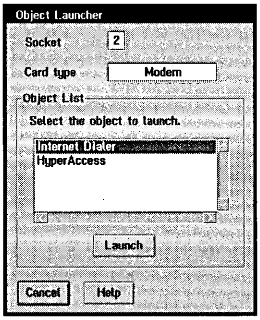
You can specify one or more programs to be started or objects to be opened using the Object List box. For each object you want to be opened, including program references for programs to be started, open the folder containing the object and drag the object into the Object List box. When you drop the object into the box, the name of the object is added to the list box.

If you want to remove an object from the Object List, select the object in the list box, then click on the **Remove** push button.

You can specify when the objects in the Object List box are to be opened in the Launch Choices group. The **To start or open an object** radio buttons select whether the object is to be opened automatically when the conditions listed in the When card is area are met or whether you want to open the objects manually.

If you put only one object into the Object List box and you select the **Automatically** radio button, this object is automatically opened when any of the When card is conditions are met. If you put more than one object into the Object List box or you select the **Manually** radio button, the system displays the Object Launcher dialog box (see Figure 7-44) when one of these conditions is met to let you select which object, if any, you want to open.

Figure 7-44. Plug and Play for PCMCIA Object — Object Launcher Dialog Box



Object Launcher

Socket: 2

Card type: Modem

Object List

Select the object to launch.

Internet Dialer

HyperAccess

The check boxes under the heading When card is allow you to specify the conditions under which the specified objects are opened. You can select any combination of these options, as follows:

- ✗ **Inserted** — The objects are opened when this type of card is inserted into any PCMCIA slot.
- ✗ **Removed** — The objects are opened when this type of card is removed from its slot.
- ✗ **Ready** — The objects are opened when this type of card changes its state from **Not Ready** to **Ready.** For example, a modem may change to the **Ready** state when it detects an active telephone line plugged into it.
- ✗ **Not ready** — The objects are opened when this type of card changes state from **Ready** to **Not Ready.**

The Inserted condition is automatically met if you open the **Plug and Play for PCMCIA** object while this type of card is installed in a PCMCIA slot. The Ready condition is also met if the card is in the Ready state when the object is opened. The Object Launcher dialog box allows you to choose an object to be opened. The Socket number tells you which PCMCIA slot triggered the Object Launcher. The Card type field tells you what type of PCMCIA card is involved.

The Object List box lists the objects that you specified in the Register Object dialog for this card type. You can select one object from the list and click on the **Launch** push button to open that object. If you don't want to launch any of these objects, click on the **Cancel** push button.

Summary

You can now customize the Workplace Shell to work the way you want it. You can change the mappings and speed of your input devices (keyboard and mouse) for your convenience. You can set the screen resolution so that objects appear in a size that you like. You can set the default behavior of many aspects of the Workplace Shell so that it will do what you want. And most important, you know how to set up the System Clock so that it displays in the most pleasant form.

With the techniques you learned in this chapter and the previous chapter, you can now personalize the Workplace Shell to make it look and feel most friendly to you.

Ira H. Schneider is an Advisory Programmer on the Workplace Shell Development Team. He has been instrumental in developing OS/2 Warp, including performance improvements and functional enhancements to the Workplace Shell. He has also worked on the OS/2 Swapper and on OS/2 System Initialization, and he was Lead Architect for OS/2 1.2. Ira has been with IBM since 1969.

Chapter 8

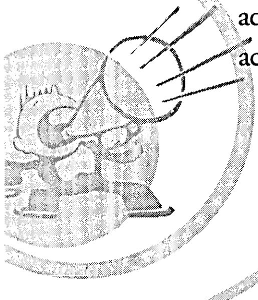
Using REXX Utilities

Quick! To the Batch Cave!

The official name of REXX is OS/2 Procedures Language 2/REXX, but I will simply refer to it as REXX (the **RE**structured e**X**tended e**X**ecutor language).

REXX is a procedural language that is used to build command procedures for OS/2. It is much more powerful than the procedure language built into CMD.EXE. REXX is a general purpose programming language containing structured programming instructions and many useful built-in functions.

You can read about the basic structure of the REXX language in the REXX Information book in the OS/2 Information folder. In this chapter, I describe how to use the advanced features of REXX, namely the REXX utility functions. These functions allow access to many OS/2 facilities, including the Workplace Shell.



Loading the REXX Utilities

You can build a REXX command file using any text editor, such as the System Editor or the TEDIT command that comes with OS/2. I will refer to a command file containing REXX commands as a REXX script.

Tip

The first line of every REXX script must be a REXX comment (that is, it must start with the characters "/*" and end with the characters "*/"). This alerts CMD.EXE that the file is a REXX script, not just a plain command file. There must be no blank lines preceding the REXX comment line.

The REXX utilities are not part of REXX itself but are kept in a separate dynamic link library. In order to use any of the REXX utilities, this dynamic link library must be loaded and the functions must be made available to REXX. This can be accomplished by putting the following statements at the beginning of your REXX script:

```
/* This loads the REXX Utilities */
call RxFuncAdd 'SysLoadFuncs','RexxUtil','SysLoadFuncs'
call SysLoadFuncs
```

Once the REXX utilities are loaded, they are available to all OS/2 sessions, not just the REXX script that you loaded them from.

Tip

If you use the REXX Utility functions frequently, you could convert your STARTUP.CMD file into a REXX script and load the REXX utility functions in the STARTUP.CMD script. If you do this, the REXX utilities will be available to all REXX scripts without each script having to explicitly load the dynamic link library.

Tip

If you don't want to load the REXX utilities in your STARTUP.CMD file, you can use the following code to load the REXX utilities only if they are needed:

```
If RxFuncQuery("SysLoadFuncs") Then
  Do
    Call RxFuncAdd "SysLoadFuncs","REXXUTIL","SysLoadFUncs"
    Call SysLoadFuncs
  End
```

REXX Utility Functions

Each REXX function has two forms:

- ✗ Result = function(parameters) — This form is used when you want to examine the result of the function

✗ Call function parameters — This form is used when you don't care about the result of the function

In the documentation of each REXX utility function, I will show you the forms that are applicable to the particular function. If a function returns a result, both forms can be used. If a function doesn't return a result, only the second form is applicable.

Note

REXX is not case sensitive. I have shown the calls to REXX functions as mixed case (similar to the syntax shown in the REXX Information book), but you can enter the function names in all uppercase, all lowercase, or any combination of upper- and lowercase characters. For example, "SysCls" is the same as "syscls," "SYSCLS," or "SyScLs." Also, the parameters that you specify to REXX functions can be of any case. If a parameter is specified as a word (such as DEFAULT or <WP_OS2SYS>), the parameter must be entered in quotes. For example, you might enter the following command:

```
call SysOpenObject "<WP_OS2SYS>",'default',1
```

The first two parameters are strings and must be entered within single (') or double quotes ("), and the third parameter is a number. If a parameter contains single quotes, you must enter it enclosed within double quotes. For example, you might enter the following command:

```
call SysOpenObject "d:\ira's_de","icon",1
```

Note

I have indicated optional parameters in the syntax for each function by enclosing the parameter in square brackets ("[parameter]"). You do not enter the square brackets when typing the command.

I have broken down the REXX utility functions into four groups to make it easier to find related functions. Table 8-1 shows which REXX utility function is in which group.

Table 8-1. Groups of REXX Utility Functions

REXX Utility Function	*Group*
RxMessageBox	User Interface
SysCls	User Interface
SysCopyObject	Workplace Shell
SysCreateObject	Workplace Shell

continued

Table 8-1. Groups of REXX Utility Functions, continued

REXX Utility Function	*Group*
SysCreateShadow	Workplace Shell
SysCurPos	User Interface
SysCurState	User Interface
SysDeregisterObjectClass	Workplace Shell
SysDestroyObject	Workplace Shell
SysDriveInfo	File Access
SysDriveMap	File Access
SysDropFuncs	Miscellaneous
SysFileDelete	File Access
SysFileSearch	File Access
SysFileTree	File Access
SysGetEA	File Access
SysGetKey	User Interface
SysGetMessage	User Interface
SysIni	File Access
SysMkDir	File Access
SysMoveObject	Workplace Shell
SysOpenObject	Workplace Shell
SysOS2Ver	Miscellaneous
SysPutEA	File Access
SysQueryClassList	Workplace Shell
SysRegisterObjectClass	Workplace Shell
SysRmDir	File Access
SysSaveObject	Workplace Shell

REXX Utility Function	*Group*
SysSearchPath	File Access
SysSetIcon	File Access
SysSetObjectData	Workplace Shell
SysSleep	User Interface
SysTempFileName	File Access
SysTextScreenRead	User Interface
SysTextScreenSize	User Interface
SysWaitNamedPipe	File Access

User Interface Functions

This group of functions can be used to build a user interface. They include access to the keyboard and screen.

RxMessageBox

The RxMessageBox function is used to display a message box. You have complete control over the information displayed in the message box.

Warning

The RxMessageBox function can only be used if the REXX command is running under a Presentation Manager environment. You must use PMREXX or another Presentation Manager program to invoke any REXX script containing a call to the RxMessageBox function. You cannot execute these scripts directly from an OS/2 command prompt.

Syntax

```
action = RxMessageBox(text,[title],[button],[icon])
call RxMessageBox text,[title],[button],[icon]
```

Parameters

- text — The message to be displayed within the message box.
- title — The title for the message box. If you don't specify a title, "Error!" is used as the title.

✗ **button** — The specification of what buttons are to be displayed in the message box. This parameter may have one of the following values:

- OK — OK button
- OKCANCEL — OK and Cancel buttons
- CANCEL — Cancel button
- ENTER — Enter button
- ENTERCANCEL — Enter and Cancel buttons
- RETRYCANCEL — Retry and Cancel buttons
- ABORTRETRYIGNORE — Abort, Retry, and Ignore buttons
- YESNO — Yes and No buttons
- YESNOCANCEL — Yes, No, and Cancel buttons

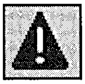
If this parameter is not specified, only an OK button is displayed.

✗ **icon** — The specification of which icon is to be displayed within the message box. This parameter may have one of the following values:

- NONE — no icon

- ERROR or HAND

- QUERY or QUESTION



- WARNING or EXCLAMATION

- INFORMATION or ASTERISK

If this parameter is not specified, no icon is displayed.

Result

The result value indicates which button was pressed. This may be one of the following values:

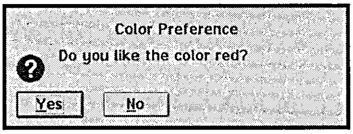
- 1 — OK button
- 2 — Cancel button
- 3 — Abort button
- 4 — Retry button
- 5 — Ignore button
- 6 — Yes button
- 7 — No button
- 8 — Enter button

Example

Display the message box shown in Figure 8-1:

```
Action = RxMessageBox ('Do you like the color red?',,
                       'Color Preference',,
                       'query',,
                       'yesno')
```

Figure 8-1.
Sample Message Box



Color Preference

Do you like the color red?

SysCls

The SysCls function is used to clear the display screen.

Syntax

```
call SysCls
```

Parameters

None

Result

None

Example

Clear the display screen:

```
call SysCls
```

SysCurPos

The SysCurPos function is used to find out the current position of the cursor on the screen or to move the cursor to a specified location on the screen.

If you invoke the REXX script from within an OS/2 Window, the cursor can only be moved around within the window.

Syntax

```
pos = SysCurPos([row,column])
call SysCurPos row,column
```

Parameters

- **row** — The text row you want to move the cursor to. Row 0 is the top of the window. The maximum value for the row parameter is number of text rows in the window minus 1. For example, if the window is 24 rows of 80 characters, the value for row must be a number between 0 and 23. You can find out the size of the window you are running using the SysTextScreenSize function.

- **column** — The text column you want to move the cursor to. Column 0 is the left side of the screen. The maximum value for the column parameter is the number of text columns in the window minus 1. For example, if the window is 24 rows of 80 characters, the value for column must be a number between 0 and 79.

If the row and column parameters are not specified, the cursor position is not changed.

Result

The result is the current cursor position in the form 'row column'. You can use the parse function value to separate these two values.

Examples

Save the current cursor location. Notice that even though you don't specify the row and column parameters on the call to SysCurPos, you must specify the parentheses.

```
parse value SysCurPos() with OldRow OldColumn
```

Save the current cursor location and move the cursor to the top-left corner of the screen. After writing elsewhere on the screen, put the cursor back where it was:

```
parse value SysCurPos(0,0) with OldRow OldColumn
     ...
call SysCurPos OldRow,OldColumn
```

SysCurState

The SysCurState function is used to make the cursor visible or invisible.

Syntax

```
call SysCurState state
```

Parameters

✗ state — The new cursor state. This must be one of the following values:

- ON — make the cursor visible
- OFF — make the cursor invisible

Result

None

Example

Hide the cursor, do some processing, then make the cursor visible again:

```
call SysCurState('off')
     ...
call SysCurState('on')
```

SysGetKey

The SysGetKey function is used to read the next keystroke from the keyboard buffer. If the keyboard buffer is empty, the SysGetKey function waits until a key is pressed. The difference between the SysGetKey function and the Charin function is that Charin waits until the Enter key is pressed before returning a keystroke while SysGetKey returns the keystroke immediately.

Syntax

```
key = SysGetKey([echo-option])
call SysGetKey [echo-option]
```

Parameters

- ✗ **echo-option** — Indicates whether you want the keystroke echoed to the screen. This value may be:
  - ECHO — Echo the pressed key to the screen.
  - NOECHO — Don't echo the pressed key.

If the echo-option parameter is not specified, the keystroke is echoed to the screen.

Result

The result is name of the key that was pressed.

Examples

Wait for the user to press any key:

```
say "Press any key to continue"
call SysGetKey
```

Wait for the user to press the Y or N key:

```
say "Do you like the color red?"
say "Press Y for yes or N for no"
do
  key = SysGetKey
until (key = 'Y' | key = 'N')
```

SysGetMessage

The SysGetMessage function is used to retrieve a message from an OS/2 message file and optionally fill in variables in the message text.

Any program, or REXX script, that is intended to be used with languages other than English should put all of its messages into a message file so that the messages can easily be translated to another language. OS/2 stores all of its messages in the file OSO001.MSG. You can build a message file using the MKMSGF utility, which comes with the OS/2 Developer's Toolkit.

Syntax

```
msg = SysGetMessage(message-number,[message-file],[items])
```

Parameters

- ✗ **message-number** — The number of the message you want to retrieve.

- **message-file** — The name of the message file containing the message. If this parameter is not specified, the message is retrieved from the OSO001.MSG file.

Note

SysGetMessage searches for the specified message file in the root directory of the boot drive, the current directory, and all directories listed in the DPATH environment variable.

- **items** — 1 to 9 strings used as substitution strings for variables in the message text. The message text should contain variables of the form %1 through %9, which correspond to the strings specified in the call to SysGetMessage. If not enough strings are specified, the excess substitution variables in the message text are not replaced. For example, if a message contains the variables %1 and %2 but you only specify one substitution string, the variable %1 is replaced by the substitution string and the variable %2 remains in the message text.

Result

The result is the text of the specified message with the variables replaced by the substitution text.

Examples

Retrieve message 190 from the OSO001.MSG file and display it without replacing any substitution variables:

```
msg = SysGetMessage(190)
say msg

The result is "SYS0190: The operating system cannot run
program %1"
```

Retrieve message 190 from the OSO001.MSG file and display it, substituting the string "Program xyz" for the first substitution variable:

```
msg = SysGetMessage(190,,"Program xyz")
say msg

The result is "SYS0190: The operating system cannot run
Program xyz"
```

SysSleep

The SysSleep function is used to delay the REXX script for a period of time.

Note

This function is normally used to wait for the user or the system to perform some action. For example, you could prompt the user to insert a diskette into the A: drive, then wait 5 seconds before checking whether or not the user complied with your request.

Syntax

```
call SysSleep delay-time
```

Parameters

✗ delay-time — The number of seconds to delay the execution of the REXX script.

Result

None

Example

Delay execution of the REXX script for 5 seconds:

```
call SysSleep(5)
```

SysTextScreenRead

The SysTextScreenRead function is used to read the characters from a portion of the window in which your REXX script is running.

Warning

The SysTextScreenRead function only returns the characters, not the color attributes. If you save the screen using the "SysTextScreenRead" function then try to restore it using the "Say" keyword instruction, the previous color settings will be lost.

Syntax

```
string = SysTextScreenRead(row, column [,length])
```

Parameters

✗ row — The starting text row number. Row 0 is the top of the window. This must be a number between 0 and the number of text rows in the window minus 1. You can use the SysTextScreenSize function to determine the number of text rows and columns in the window.

- ✗ **column** — The starting text column number. Column 0 is the left side of the window. This must be a number between 0 and the number of text columns in the window minus 1.
- ✗ **length** — The number of characters to be read. If this parameter is not specified, the SysTextScreenRead function returns all of the characters from the starting location to the end of the window.

Result

The result is a string of characters containing the text from the specified portion of the window.

Examples

Read the first line of the window:

```
parse value SysTextScreenSize() with rows columns
line1 = SysTextScreenRead(0,0,columns)
say "The first line in the window is"
say line1
```

Save the contents of the screen, do some processing, then restore the text characters to the screen:

```
parse value SysTextScreenSize() with rows columns
WindowText = SysTextScreenRead(0,0,rows*columns)
call SysCls
   ...
call SysCls
call SysCurPos(0,0)
say WindowText
```

SysTextScreenSize

The SysTextScreenSize function is used to determine the number of rows and columns in the window in which your REXX script is running.

Syntax

```
size = SysTextScreenSize()
```

Parameters

None

Result

The result is the screen size formatted as 'rows columns'. You can use the Parse function to separate the number of rows and columns into separate variables.

Example

Find the number of rows and columns in the current window:

```
parse value SysTextScreenSize() with rows columns
say "The window is "rows" rows high and "columns" ||,
columns wide"
```

Workplace Shell Functions

This group of functions can be used to communicate with the Workplace Shell.

Referencing Workplace Shell Objects

Many of these functions allow you to reference Workplace Shell objects. There are two ways that you can reference an object — by its path name or by its object identifier.

Using Path Names

If the object you want to reference is a Folder or Data File Object, you can specify the fully qualified path name of the corresponding directory or file. For example, to reference the Desktop Folder, you might specify **C:\DESKTOP** as the path name.

Warning

The path name may not be the same as the object's title if you are using the FAT file system. For example, the Productivity folder is within the OS/2 System folder on the Desktop. If your boot partition was formatted using the HPFS file system, the fully qualified path name is d:\Desktop\OS/2 System\Productivity; for the FAT file system the path name is d:\DESKTOP\OS!2_SYS\PRODUCTI, where d: is the boot drive.

Tip

You can find the fully qualified path name for any file system object by opening the Settings Notebook for the object and turning to the File page. You can construct the fully qualified path name by taking the path specified in the Path field and concatenating it to the filename in the File field.

Using Object Identifiers

Each object may have an Object Identifier assigned to it when it was created or an Object Identifier may be added later. In either case, you can use the Object Identifier to refer to the object in a REXX Utility function. An Object Identifier always starts with the less-than character ("<") and ends with the greater-than character (">").

Warning

Even though in general REXX is not case sensitive, the Object Identifier passed to a REXX utility function must be in the proper case. Therefore, if you are trying to reference the Desktop Folder, you must specify the Object Identifier as "<WP_DESKTOP>", not "<wp_desktop>".

Many of the objects installed by OS/2 have predefined Object Identifiers that you can use. Table 8-2 lists the Object Identifiers for all objects installed by System Installation.

Table 8-2. OS/2 Object Identifiers

Object	*Object Identifier*	*Notes*
Desktop Folder	<WP_DESKTOP>	
OS/2 System Folder	<WP_OS2SYS>	
System Setup Folder	<WP_CONFIG>	
Add Programs	<WP_MIGAPP>	
Color Palette	<WP_CLRPAL>	This is the OS/2 2.1 Color Palette. This object exists only if you installed OS/2 Warp over a previous version of OS/2.
Country	<WP_CNTRY>	
Create Utility Diskettes	<WP_BOOT>	
Device Driver Install	<WP_DDINST>	
Font Palette	<WP_FNTPAL>	
Keyboard	<WP_KEYB>	
Mixed Color Palette	<WP_HIRESCLRPAL>	
Mouse	<WP_MOUSE>	
Plug and Play for PCMCIA	<WP_CMPNP>	Installed only if you have a PCMCIA slot.
Power	<WP_POWER>	Installed only if your computer supports Advanced Power Management.
Scheme Palette	<WP_SCHPAL28>	This is the OS/2 Warp Scheme Palette.

continued

Table 8-2. OS/2 Object Identifiers, continued

Object	*Object Identifier*	*Notes*
Scheme Palette	<WP_SCHPAL>	This is the OS/2 2.1 Scheme Palette. This object exists only if you installed OS/2 Warp over a previous version of OS/2.
Selective Install	<WP_INST>	
Spooler	<WP_SPOOL>	
System	<WP_SYSTEM>	
System Clock	<WP_CLOCK>	
Touch	<WP_TOUCH>	Installed only if you have a touch screen display.
Selective Uninstall	<WP_UNINST>	
Solid Color Palette	<WP_LORESCLRPAL>	
Sound	<WP_SOUND>	
WIN-OS/2 Setup	<WP_WINCFG>	Installed only if you installed WIN-OS/2 support.
Command Prompts Folder	<WP_PROMPTS>	
DOS from Drive A:	<WP_DOS_DRV_A>	
DOS Full Screen	<WP_DOSFS>	
DOS Window	<WP_DOSWIN>	
Dual Boot	<WP_DBOOT>	Installed only if DOS was previously installed on the boot partition.
OS/2 Full Screen	<WP_OS2FS>	
OS/2 Window	<WP_OS2WIN>	
WIN-OS/2 Full Screen	<WP_WINFS>	Installed only if you installed the WIN-OS/2 support.
WIN-OS/2 Window	<WP_WIN2WIN>	Installed only if you installed the WIN-OS/2 support.

Object	*Object Identifier*	*Notes*
Productivity Folder	<WP_TOOLS>	
Clipboard Viewer	<WP_CLIPV>	
Enhanced Editor	<WP_EPM>	
Icon Editor	<WP_ICON>	
OS/2 System Editor	<WP_SYSED>	
Picture Viewer	<WP_PICV>	
Pulse	<WP_PULSE>	
Seek and Scan Files	<WP_SEEK>	
Games Folder	<WP_GAMES>	
Mahjongg Solitaire Folder	<MAH_FOLDER>	
Mahjongg	<MAH_EXE>	
OS/2 Chess	<WP_CHESS>	
Solitaire — Klondike	<WP_KLOK>	
Drives Folder	<WP_DRIVES>	
Startup Folder	<WP_START>	
Shredder	<WP_SHRED>	
Minimized Window Viewer	<WP_VIEWER>	
Information Folder	<WP_INFO>	
Application Considerations	<WP_APPLBK>	
Command Reference	<WP_CMDREF>	
Glossary	<WP_GLOSS>	
Master Help Index	<WP_MINDEX>	
Multimedia	<WP_MULTIMBK>	
Performance Considerations	<WP_PERFBK>	

continued

Table 8-2. OS/2 Object Identifiers, continued

Object	*Object Identifier*	*Notes*
Printing in OS/2	<WP_PRINTBK>	
Readme	<WP_RDME>	
REXX Information	<WP_REXREF>	
Trademarks	<WP_TRADEMBK>	
Tutorial	<WP_TUTOR>	
Windows Programs in OS/2	<WP_WINOS2BK>	
Templates Folder	<WP_TEMPS>	
Multimedia Folder	<MMPM2_FOLDER>	
Compact Disc	<MMPM_CDPLAYER1>	
Digital Audio	<MMPM_DAPLAYER1>	
Digital Video	<MMPM2_SOFTWARE_MOTION_VIDEO1>	
MIDI	<MMPM_MIDIPLAYER1>	
Movies Folder	<MMPM2_MOVIES>	
Multimedia Application Install	<MMPM2_INSTALL>	
Multimedia Data Converter	<MMPM2_MMCONVERTER>	
Multimedia Setup	<MMPM2_SETUP>	
Multimedia Viewer Folder	<LT_SAMPLE>	
Multimedia With REXX	<MCIREXX_INF>	
Sound (shadow)	<MMPM2_SNDSHAD>	
Sound Bites Folder	<MMPM2_SOUNDS>	
Volume Control	<MMPM2_MASTERVOLUME_D>	
LaunchPad	<WP_LAUNCHPAD>	

Object	*Object Identifier*	*Notes*
Printer	<WP_PDVIEW>	Installed only if you selected a printer during System Installation.
Network	<WP_NETWORK>	Visible only if you are connected to a Local Area Network (LAN).
Nowhere Folder	<WP_NOWHERE>	This folder is normally not visible.
View	<WP_VIEWINF>	

You can find out the Object Identifier currently assigned to any object by using the Object Tool on the CD-ROM included with this book. If the object you want to reference doesn't have an Object Identifier, you can use the Object Tool to assign one to the object.

Setup Strings

Several of the Workplace Shell functions allow you to specify a setup string. This is a string passed to the object to modify the properties of the object.

Setup strings have the form *keyword=value;keyword=value;...*, where *keyword=* specifies what property is to be modified and *value* specifies the new value for that property. Most keywords support single values, either numbers or strings. However, some of the keywords support a list of values, where each value in the list is separated from the other values by a comma or a blank. If more than one property is being specified, each *keyword=value* clause is separated from the next by a semicolon (";") character. For example, you can specify a setup string as **iconfile=sample.ico;objectid=<my_object>.**

Table 8-3 lists all of the setup strings supported by the object classes defined by the Workplace Shell. For each setup string, it shows what object class defines the setup string and points to the table containing the definition of the setup string. Table 8-4 through Table 8-18 show the setup strings defined by each object class. These tables do not list any setup strings recognized by the classes defined in any BonusPak application program.

Table 8-3. Index to Setup String Tables

Keyword	*Defined by Class*	*See Table*
ALWAYSSORT=	WPFolder	8-5
APPDEFAULT=	WPPrinter	8-17
ASSOCFILTER=	WPProgram and WPProgramFile	8-9
ASSOCTYPE=	WPProgram and WPProgramFile	8-9
AUTOSETUP=	WPColorPalette	8-14
AUTOSETUP=	WPSchemePalette	8-16
BACKGROUND=	WPFolder	8-5
CCVIEW=	WPObject	8-4
COLORS=	WPColorPalette	8-14
DEFAULTSORT=	WPFolder	8-5
DEFAULTVIEW=	WPObject	8-4
DEFAULTVIEW=	WPFolder	8-5
DEFAULTVIEW=	WPPrinter	8-17
DETAILSCLASS=	WPFolder	8-5
DETAILSFONT=	WPFolder	8-5
DETAILSTODISPLAY=	WPFolder	8-5
DETAILSVIEW=	WPFolder	8-5
DRAWEROBJECTS=	WPLaunchPad	8-7
DRIVENUM=	WPDrive	8-6
EXENAME=	WPProgram and WPProgramFile	8-9
FONTS=	WPFontPalette	8-15
FPOBJECTS=	WPLaunchPad	8-7
GROUP=	WPProgramGroup	8-10
HELPLIBRARY=	WPObject	8-4
HELPPANEL=	WPObject	8-4

Keyword	*Defined by Class*	*See Table*
HIDEBUTTON=	WPObject	8-4
ICONFILE=	WPObject	8-4
ICONFONT=	WPFolder	8-5
ICONNFILE=	WPFolder	8-5
ICONNRESOURCE=	WPFolder	8-5
ICONPOS=	WPObject	8-4
ICONRESOURCE=	WPObject	8-4
ICONVIEW=	WPFolder	8-5
ICONVIEWPOS=	WPFolder	8-5
JOBDIALOGBEFOREPRINT=	WPPrinter	8-17
JOBPROPERTIES=	WPPrinter	8-17
LPACTIONSYTLE=	WPLaunchPad	8-7
LPCLOSEDRAWER=	WPLaunchPad	8-7
LPDRAWERTEXT=	WPLaunchPad	8-7
LPFLOAT=	WPLaunchPad	8-7
LPHIDECTLS=	WPLaunchPad	8-7
LPSMALLICONS=	WPLaunchPad	8-7
LPTEXT=	WPLaunchPad	8-7
LPVERTICAL=	WPLaunchPad	8-7
MAXIMIZED=	WPProgram and WPProgramFile	8-9
MINIMIZED=	WPProgram and WPProgramFile	8-9
MINWIN=	WPObject	8-4
NETGCOMMENT=	WPNetGrp	8-8
NETID=	WPRPrinter	8-18
NOAUTOCLOSE=	WPProgram and WPProgramFile	8-9

continued

Table 8-3. Index to Setup String Tables, continued

Keyword	*Defined by Class*	*See Table*
NOCOPY=	WPObject	8-4
NODELETE=	WPObject	8-4
NODRAG=	WPObject	8-4
NODROP=	WPObject	8-4
NOLINK=	WPObject	8-4
NOMOVE=	WPObject	8-4
NOPRINT=	WPObject	8-4
NORENAME=	WPObject	8-4
NOSETTINGS=	WPObject	8-4
NOSHADOW=	WPObject	8-4
NOTVISIBLE=	WPObject	8-4
OBJECTID=	WPObject	8-4
OPEN=	WPObject	8-4
OPEN=	WPFolder	8-5
OUTPUTTOFILE=	WPPrinter	8-17
PARAMETERS=	WPProgram and WPProgramFile	8-9
PORTNAME=	WPPrinter	8-17
PRINTDRIVER=	WPPrinter	8-17
PRINTERSPECIFICFORMAT=	WPPrinter	8-17
PRINTWHILESPOOLING=	WPPrinter	8-17
PROGTYPE=	WPProgram and WPProgramFile	8-9
QSTARTTIME=	WPPrinter	8-17
QSTOPTIME=	WPPrinter	8-17
QUEUE=	WPRPrinter	8-18
QUEUEDRIVER=	WPPrinter	8-17

Keyword	*Defined by Class*	*See Table*
QUEUENAME=	WPPrinter	8-17
REFRESHINTERVAL=	WPRPrinter	8-18
REMOVEFONTS=	WPFolder	8-5
SEPARATORFILE=	WPPrinter	8-17
SET	WPProgram and WPProgramFile	8-9
SHADOWID=	WPShadow	8-12
SHOWOBJS=	WPRPrinter	8-18
SORTBYATTR=	WPFolder	8-5
SORTCLASS=	WPFolder	8-5
SRVCOMMENT=	WPServer	8-11
STARTUPDIR=	WPProgram and WPProgramFile	8-9
TEMPLATE=	WPObject	8-4
TITLE=	WPObject	8-4
TREEFONT=	WPFolder	8-5
TREEVIEW=	WPFolder	8-5
WORKAREA=	WPFolder	8-5
XCELLCOUNT=	WPPalette	8-13
XCELLCOUNT=	WPFontPalette	8-15
XCELLCOUNT=	WPSchemePalette	8-16
XCELLGAP=	WPPalette	8-13
XCELLWIDTH=	WPPalette	8-13
YCELLCOUNT=	WPPalette	8-13
YCELLCOUNT=	WPFontPalette	8-15
YCELLCOUNT=	WPSchemePalette	8-16
YCELLGAP=	WPPalette	8-13
YCELLHEIGHT=	WPPalette	8-13

Table 8-4. Setup Strings for All Objects (Defined in WPObject)

Keyword	*Value*	*Description*
CCVIEW=	YES | NO | DEFAULT	Specifies the concurrent view behavior, which may be YES (allow multiple concurrent views), NO (do not allow multiple concurrent views), or DEFAULT (use system default value).
DEFAULTVIEW=	SETTINGS | DEFAULT | n	Specifies the default view for the object, which may be SETTINGS (Settings Notebook view), DEFAULT (the default view defined by the object class), or a number (for a special class-defined view).
HELPLIBRARY=	library	Specifies the name of the help library for this object.
HELPPANEL=	id	Specifies the help panel identifier for this object.
HIDEBUTTON=	YES | NO	Specifies the default button appearance, which may be YES (hide button) or NO (minimize button).
ICONFILE=	filename	Specifies the .ICO file containing the icon to be used for this object.
ICONPOS=	x,y	Specifies the initial (x,y) coordinates of the icon for this object in the icon view of the folder containing the object. These coordinates are specified as percentages (between 0 and 100).
ICONRESOURCE=	id,module	Specifies the icon resource to be used for this object. "module" is the name of the dynamic link library containing the resource, and "id" is the resource identifier of the icon resource.
MINWIN=	HIDE | DESKTOP | VIEWER	Specifies the minimized window behavior, which may be HIDE (hide object), DESKTOP (minimize to the desktop), or VIEWER (minimize to the Minimized Window Viewer).

Keyword	*Value*	*Description*
NOCOPY=	YES | NO	This object can be copied.
NODELETE=	YES | NO	This object can be deleted.
NODRAG=	YES | NO	This object can be dragged.
NODROP=	YES | NO	This object can be the target of a drop operation.
NOLINK=	YES | NO	You can create a shadow of this object. This is the same as the NOSHADOW= keyword.
NOMOVE=	YES | NO	This object can be moved.
NOPRINT=	YES | NO	The object can be printed.
NORENAME=	YES | NO	The object can be renamed.
NOSETTINGS=	YES | NO	The Settings Notebook can be opened for this object.
NOSHADOW=	YES | NO	You can create a shadow of this object. This is the same as the NOLINK= keyword.
NOTVISIBLE=	YES | NO	This object is visible.
OBJECTID=	<identifier>	Specifies the object identifier string.
OPEN=	DEFAULT | SETTINGS	Open the specified view of the object, which may be DEFAULT (open the default view of the object) or SETTINGS (open the Settings Notebook).
TEMPLATE=	YES | NO	This object is a template.
TITLE=	title	Specifies the object's title.

Table 8-5. Setup Strings for All Folders (Defined in WPFolder)

Keyword	*Value*	*Description*
ALWAYSSORT=	YES I NO	This folder is to be kept in sorted order.
BACKGROUND=	[filename], [N I T I S], [scale], [I I C], [red green blue I D]	Specifies the background for the folder. "filename" specifies the fully qualified name of the image file or "(none)" ("?:/...") indicates that the file resides on the boot drive); N indicates normal image, T indicates tiled image, S indicates scaled image; scale is the scaling factor (if S was specified) as a number between 1 and 20; I indicates display image, C indicates color only; red, green, and blue are the amounts of each of the primary colors to be used for the background color, and D indicates use the default background color.
DEFAULTSORT=	index	Specifies the index number of the details data item to be used as the default sort property.
DEFAULTVIEW=	ICON I TREE I DETAILS	Specifies the default view for the object, which may be ICON (icon view), TREE (tree view), or DETAILS (details view).
DETAILSCLASS=	classname	The object class that supplies the formatting information for details view.
DETAILSFONT=	fontname	Specifies the font to be used in details view.
DETAILSTODISPLAY=	index[,...]	Specifies a list of index numbers of details data to be displayed. For example, "DETAILSTODISPLAY=0,2" indicates that only the first and third details data items are to be displayed.
DETAILSVIEW=	NORMAL I MINI	Specifies the icon size in details view. The icon size may be NORMAL (normal size icons) or MINI (small icons).
ICONFONT=	fontname	Specifies the font to be used in icon view.

Keyword	*Value*	*Description*
ICONNFILE=	index,filename	Specifies the .ICO file containing the icon to be used for the animation icon for this folder. The value of "index" must be 1 (indicating the open folder icon).
ICONNRESOURCE=	index,id,module	Specifies the icon resource to be used for the animation icon for this folder. The value of "index" must be 1 (indicating the open folder icon). "module" is the name of the dynamic link library containing the resource and "id" is the resource identifier of the icon resource.
ICONVIEW=	NONGRID | FLOWED | NONFLOWED, NORMAL | MINI | INVISIBLE	Specifies the arrangement and icon size in icon view. The arrangement may be NONGRID (non-grid view), FLOWED (flowed view), or NONFLOWED (nonflowed view). The icon size may be NORMAL (normal size icons), MINI (small icons), or INVISIBLE (no icons).
ICONVIEWPOS=	x,y,cx,cy	Specifies the initial position and size of the icon view, where (x,y) are the coordinates of the lower-left corner of the window and (cx,cy) are the size of the window, in pels.
OPEN=	ICON | TREE | DETAILS	Open the specified view of the object, which may be ICON (open the icon view), TREE (open the tree view), or DETAILS (open the details view).
REMOVEFONTS=	YES | NO	Remove all instance fonts. This allows the default fonts to be used for all folder views.
SORTBYATTR=	index[,...]	Specifies a list of index numbers of details data to be shown in the "Sort by attribute" list for the folder. For example, "SORTBYATTR=0,2" indicates that only the first and third details data items are to be shown as potential sort properties.

continued

Table 8-5. Setup Strings For All Folders (Defined in WPFolder), continued

Keyword	*Value*	*Description*
SORTCLASS=	classname	Specifies the name of the object class that supplies the sorting information. The default is WPFileSystem.
TREEFONT=	fontname	Specifies the font to be used in tree view.
TREEVIEW=	LINES | NOLINES, NORMAL | MINI | INVISIBLE	Specifies the lines option and icon size in tree view. The lines option may be LINES (draw the hierarchy lines) or NOLINES (don't draw the lines). The icon size may be NORMAL (normal size icons), MINI (small icons), or INVISIBLE (no icons).
WORKAREA=	YES | NO	This object is a work area.

Table 8-6. Setup Strings for Drive Objects (Defined in WPDisk)

Keyword	*Value*	*Description*
DRIVENUM=	n	Specifies the logical drive number. "n" must be a number between 1 and 26, corresponding to drive letters "A:" through "Z:".

Table 8-7. Setup Strings for LaunchPad Objects (Defined in WPLaunchPad)

Keyword	*Value*	*Description*
DRAWEROBJECTS=	drawer,id[,...]	Specifies the objects to be added to a drawer of the LaunchPad. "drawer" is the number of the drawer, reading from left to right or bottom to top. You can specify the LaunchPad itself by specifying a drawer number of 0. The drawer number is followed by a list of object identifiers or filenames, separated by commas.

Keyword	*Value*	*Description*
FPOBJECTS=	id[,...]	Specifies objects that are to be on the LaunchPad. If FPOBJECTS= is specified when creating a LaunchPad, these are the only objects on the LaunchPad. If FPOBJECTS= is specified for an existing LaunchPad, the specified objects are added to the LaunchPad. The value is a list of object identifiers or filenames, specifying the objects to be added to the LaunchPad. The objects in the list are separated by commas.
LPACTIONSYTLE=	OFF | MINI | NORMAL | TEXT	Specifies how the action buttons are to be displayed. "OFF" specifies that the action buttons are not displayed on the LaunchPad, "MINI" specifies that the action buttons are to be displayed as small icons, "NORMAL" specifies that the action buttons are to be displayed as normal icons, and "TEXT" specifies that the action buttons are to be displayed as text.
LPCLOSEDRAWER=	YES | NO	Specifies that a drawer on the LaunchPad is to be closed after opening an object in the drawer.
LPDRAWERTEXT=	YES | NO	Specifies that the title of each object is to be shown below the object's icon in the drawers of the LaunchPad.
LPFLOAT=	YES | NO	Specifies that the LaunchPad is to float on top of all other windows.
LPHIDECTLS=	YES | NO	Specifies that the frame controls are not to be displayed around the LaunchPad windows.
LPSMALLICONS=	YES | NO	Specifies that the object icons are to be displayed as small icons.
LPTEXT=	YES | NO	Specifies that the title of each object is to be shown on the LaunchPad. This does not affect the objects in drawers.
LPVERTICAL=	YES | NO	Specifies that the LaunchPad is to be displayed vertically.

Table 8-8. Setup Strings for Network Group Folders (Defined in WPNetgrp)

Keyword	*Value*	*Description*
NETGCOMMENT=	title	Specifies the title for a Network Group folder.

Table 8-9. Setup Strings for Program Reference and Program File Objects (Defined in WPProgram and WPProgramFile)

Keyword	*Value*	*Description*
ASSOCFILTER=	filters	Specifies the filename filter for files associated with this program. Multiple filters are separated by commas.
ASSOCTYPE=	types	Specifies the data types for files associated with this program. Multiple data types are separated by commas.
EXENAME=	filename	Specifies the name of the program file.
MAXIMIZED=	YES | NO	Specifies that the program is to be started in a maximized window.
MINIMIZED=	YES | NO	Specifies that the program is to be started in a minimized window.
NOAUTOCLOSE=	YES | NO	Specifies that the window is to be kept open when the program terminates.
PARAMETERS=	parameters	Specifies the parameter list, which may include substitution characters.

Keyword	***Value***	***Description***
PROGTYPE=	FULLSCREEN | WINDOWABLEVIO | VDM | WINDOWEDVDM | WIN | PROG_30_STD | PROG_31_STD | PROG_31_ENH | PROG_31_STDSEAMLESSCOMMON | PROG_31_ENHSEAMLESSCOMMON | SEPARATEWIN | PROG_31_STDSEAMLESSVDM | PROC_31_ENHSEAMLESSVDM | PM	Specifies how to run the program: •FULLSCREEN — OS/2 full screen •WINDOWABLEVIO — OS/2 windowed •VDM — DOS full screen •WINDOWEDVDM — DOS windowed •WIN — WIN-OS/2 full screen •PROG_30_STD — WIN-OS/2 full screen in Windows 3.0 standard mode •PROG_31-STD — WIN-OS/2 full screen in Windows 3.1 standard mode •PROG_31_ENH — WIN-OS/2 full screen Windows 3.1 enhanced mode •PROG_31_STDSEAMLESSCOMMON — WIN-OS/2 window in standard mode common session •PROG_31_ENHSEAMLESSCOMMON — WIN-OS/2 window in enhanced mode common session •SEPARATEWIN — WIN-OS/2 window in separate session •PROG_31_STDSEAMLESSVDM — WIN-OS/2 window in standard mode separate session •PROG_31_ENHSEAMLESSVDM — WIN-OS/2 window in enhanced mode separate session •PM — PM application
SET	variable=value	Sets the environment variable "variable" to the specified value. This is also used to set the DOS settings.
STARTUPDIR=	pathname	Specifies the name of the working directory.

Table 8-10. Setup Strings for Program Group Folders (Defined in WPProgramGroup)

Keyword	Value	Description
GROUP=	handle	Specifies the program group handle as a decimal ASCII string.

Table 8-11. Setup Strings for Server Folders (Defined in WPServer)

Keyword	Value	Description
SRVCOMMENT=	title	Specifies the title for a Server Folder.

Table 8-12. Setup Strings for Shadow Objects (Defined in WPShadow)

Keyword	Value	Description
SHADOWID=	<object identifier> or filename	Specifies the object identifier or the fully qualified pathname of the object that this is a shadow of.

Table 8-13. Setup Strings for All Palette Objects (Defined in WPPalette)

Keyword	Value	Description
XCELLCOUNT=	columns	Specifies the number of columns in the palette. "columns" must be a decimal number. If this parameter is not specified when a palette is being created, the palette will have 13 columns.
XCELLGAP=	width	Specifies the horizontal gap between each cell in the palette. "width" is specified in pixels.
XCELLWIDTH=	width	Specifies the width of each cell in the palette. "width" is specified in pixels.

Keyword	*Value*	*Description*
YCELLCOUNT=	rows	Specifies the number of rows in the palette. "rows" must be a decimal number. If this parameter is not specified when a palette is being created, the palette will have 9 rows.
YCELLGAP=	height	Specifies the vertical gap between each cell in the palette. "height" is specified in pixels.
YCELLHEIGHT=	height	Specifies the height of each cell in the palette. "height" is specified in pixels.

Table 8-14. Setup Strings for Color Palette Objects (Defined in WPColorPalette)

Keyword	*Value*	*Description*
AUTOSETUP=	HIRES I LORES	Specifies the number of default colors in the color palette. The value of this parameter may be "HIRES" for the 256 color Mixed Color Palette or "LORES" for the 16 color Solid Color Palette. If this parameter is not specified when creating a color palette, the 72 color OS/2 2.x Color Palette is created.
COLORS=	color[,...]	Specifies the colors to be used in the color palette. Each value is of the form 0xrrggbb, where rr is the Red value, gg is the Green value, and bb is the Blue value. Each value is a 2-digit hexadecimal number between 00 and FF (0 to 255 decimal).

Table 8-15. Setup Strings for Font Palette Objects (Defined in WPFontPalette)

Keyword	*Value*	*Description*
FONTS=	font[,...]	Specifies the fonts to be used in the font palette. Each value is of the form "size.name", where "size" is the point size and "name" is the name of the font, including modifiers. For example, specify "9.Helv" for a 9 point Helv font or "18.Tms Rmn Bold Italic" for an 18 point bold italic Times Roman font.
XCELLCOUNT=	columns	Specifies the number of columns in the font palette. "columns" must be a decimal number. If this parameter is not specified when a font palette is being created, the palette will have 2 columns.
YCELLCOUNT=	rows	Specifies the number of rows in the font palette. "rows" must be a decimal number. If this parameter is not specified when a font palette is being created, the palette will have 4 rows.

Table 8-16. Setup Strings for Scheme Palette Objects (Defined in WPSchemePalette)

Keyword	*Value*	*Description*
AUTOSETUP=	<u>YES</u> I NO	Specifies the schemes to be put into each cell. If "YES" is specified (or if this parameter is not specified when creating a Scheme Palette), the first 28 schemes are initialized to the predefined OS/2 Warp schemes, and the excess are initialized to the Default scheme. If "NO" is specified, all of the schemes are initialized to the Default scheme.
XCELLCOUNT=	columns	Specifies the number of columns in the scheme palette. "columns" must be a decimal number. If this parameter is not specified when a scheme palette is being created, the palette will have 4 columns.

Keyword	*Value*	*Description*
YCELLCOUNT=	rows	Specifies the number of rows in the scheme palette. "rows" must be a decimal number. If this parameter is not specified when a scheme palette is being created, the palette will have 7 rows.

Table 8-17. Setup Strings for Printer Objects (Defined in WPPrinter)

Keyword	*Value*	*Description*
APPDEFAULT=	YES | NO	Specifies that this printer object is to be the default printer.
DEFAULTVIEW=	ICON | DETAILS	Specifies the default view for the printer object. This parameter may have the value "ICON" (for icon view) or "DETAILS" (for details view).
JOBDIALOG BEFOREPRINT=	YES | NO	Specifies that the job properties dialog is to be displayed before starting every print job.
JOBPROPERTIES=	filename	Specifies the name of a file containing the default job properties for this printer object.
OUTPUTTOFILE=	YES | NO	Specifies that printer output is to be put into a file instead of printed. You will be asked for a filename for every job sent to this printer.
PORTNAME=	port[,...]	Specifies one or more ports that this printer object is connected to. If more than one port are specified, the names are separated by commas.
PRINTDRIVER=	driver.device[,...]	Specifies one or more printer drivers that this printer can use. Each printer driver must be specified as the full driver name, a period, and the device name (for example, "IBM42XX.IBM 4201 PROPRINTER III". If more than one driver are specified, the names are separated by commas. The printer drivers must already be installed.

continued

Table 8-17. Setup Strings for Printer Objects (Defined in WPPrinter), continued

Keyword	*Value*	*Description*
PRINTER SPECIFICFORMAT=	YES | NO	Specifies that the print jobs contain printer-dependent formatting information.
PRINTWHILE SPOOLING=	YES | NO	Specifies that the printer object is to start printing the job while it is still being created by the application program. The printer will print each page as the program completes it.
QSTARTTIME=	hh:mm	Specifies the time at which this printer is to start printing. The value of this parameter is the starting hour (in 24 hour notation — 0 to 23), a colon, and the starting minute (a number from 0 to 59).
QSTOPTIME=	hh:mm	Specifies the time at which this printer is to stop printing.
QUEUEDRIVER=	driver	Specifies the name of the queue driver to be used. The queue driver must already be installed.
QUEUENAME=	queue	Specifies the local queue name for the printer object. If the queue doesn't exist, the printer object creates it. If the printer is already connected to a queue, this keyword is ignored.
SEPARATORFILE=	filename	Specifies the name of the file containing the separator page that is printed before each job.

Table 8-18. Setup Strings for Remote Printer Objects (Defined in WPRPrinter)

Keyword	*Value*	*Description*
NETID=	networkname	Specifies the full name of the printer resource as it is known to the network.
QUEUE=	queuename	Specifies the name of the local queue to be used for this printer.

Keyword	*Value*	*Description*
REFRESHINTERVAL=	seconds	Specifies how often the open view of the printer object is updated. The value is the number of seconds between refreshes.
SHOWOBJS=	OWN I ALL	Specifies what objects are displayed in the printer object. If "OWN" is specified, only the current user's jobs are displayed. If "ALL" is specified, all jobs are displayed.

SysCopyObject

The SysCopyObject function is used to copy a Workplace Shell object from one folder to another. This has the same effect as holding the Ctrl key while dragging the object to another folder, or using the **Copy...** pop-up menu item for the object.

Syntax

```
RC = SysCopyObject(name,folder)
call SysCopyObject name,folder
```

Parameters

- ✗ **name** — The name of the object to be copied. This may be specified as an Object Identifier or, if the object is a Folder or Data File, the fully qualified path name of the object.
- ✗ **folder** — The name of the folder that the object is to be copied into. This may be specified as an Object Identifier or the fully qualified path name of the folder.

Result

The result is an indication of whether or not the object was copied successfully, as follows:

- ✗ 0 — The object could not be copied.
- ✗ 1 — The object was copied successfully.

Example

Copy the program reference object for an OS/2 Window from the Command Prompts Folder to the Desktop:

```
call SysCopyObject "<WP_OS2WIN>","<WP_DESKTOP>"
```

Note

Notice the use of the Object Identifiers from Table 8-2 in this example. Because both the object and the destination folder are standard OS/2 objects, they both have Object Identifiers which simplifies their references.

SysCreateObject

The SysCreateObject function is used to create a new Workplace Shell object. This is similar to the effect you get when you create a new object by dragging a copy from the Templates folder or by selecting the **Create another...** pop-up menu item for an object. The main difference is that the SysCreateObject function creates a non-customized version of the object, whereas dragging from a template allows you to customize the template before creating objects from it.

Syntax

```
RC = SysCreateObject(classname,title,folder[,setup][,action])
call SysCreateObject classname,title,folder[,setup][,action]
```

Parameters

- ✘ **classname** — The class of the object to be created. This must be the name of a registered object class. This can be any of the Workplace Shell object classes (see Chapter 5) or the name of any object class registered by an application program.
- ✘ **title** — The title of the new object. This is the name that will appear below or next to the object's icon in an open view of the folder that contains the object.
- ✘ **folder** — The name of the folder that the object is to be created in. This may be specified as an Object Identifier or the fully qualified path name of the folder.
- ✘ **setup** — The setup string to be used to initialize the object. This setup string may specify the properties you want the object to have when it is created, such as its icon. If this parameter is not specified, the object's properties are initialized to their default values (see Setup Strings section earlier in this chapter).
- ✘ **action** — The action to be taken if the specified object already exists. This parameter may have one of the following values:
  - FAIL — Fail the SysCreateObject request if the object already exists in the destination folder.

- REPLACE — Delete the existing object and create a new object. If you choose this option, any customizations you had for the original object are lost..
- UPDATE — Pass the setup string (if supplied) to the existing object. This option allows you to create a new object if one doesn't already exist or simply update the old object with properties you wanted to assign to the new object.

You only have to specify the first letter of the action value (that is, "F", "R", or "U").

Result

The result is an indication of whether or not the object was created successfully, as follows:

- ✗ 0 — The object could not be created. This result is returned if the specified object already exists and you specified the "FAIL" action code or if the object could not be created for any other reason.
- ✗ 1 — The object was created successfully.

Examples

Create a new folder on the desktop whose name is "My Folder", make it a work area folder, and assign an Object Identifier of "<MY_FOLDER>" to the new folder:

```
call SysCreateObject "wpfolder",,
                     "My Folder",,
                     "<WP_DESKTOP>",,
                     "workarea=yes;objectid=<MY_FOLDER>",,
                     "f"
```

Create a new LaunchPad object in the folder created in the previous example and fill in some buttons:

```
call SysCreateObject "WPLaunchPad",,
                     "My LaunchPad",,
                     "<MY_FOLDER>",,
                     "fpobjects=<WP_OS2FS>,<WP_OS2WIN>;",
                     "objectid=<MY_LAUNCHPAD>",,
                     "r"
```

CD-ROM

The CD-ROM included with this book contains a set of REXX scripts that can be used to create any standard Workplace Shell object that you accidentally deleted (or deleted and now want to have back). These REXX scripts all call the SysCreateObject function to recreate these objects.

SysCreateShadow

The SysCreateShadow function is used to create a shadow of a Workplace Shell object in a specified folder. This has the same effect as holding the Ctrl and Shift keys while dragging the object to another folder, or using the **Create shadow...** pop-up menu item for the object.

Syntax

```
RC = SysCreateShadow(name,folder)
call SysCreateShadow name,folder
```

Parameters

- **name** — The name of the object for which you want to create a shadow. This may be specified as an Object Identifier or, if the object is a Folder or Data File, the fully qualified path name of the object.
- **folder** — The name of the folder that the shadow object is to be created in. This may be specified as an Object Identifier or the fully qualified path name of the folder.

Result

The result is an indication of whether or not the shadow object was created successfully, as follows:

- 0 — The shadow object could not be created.
- 1 — The shadow object was created successfully.

Example

Create a shadow of the program reference object for an OS/2 Window on the Desktop:

```
call SysCreateShadow "<WP_OS2WIN>","<WP_DESKTOP>"
```

There is a subtle difference between using the SysCreateShadow function to create a shadow object and using SysCreateObject(**wpshadow,...**) to create the same shadow object. The difference is that if you use the SysCreateShadow function, the shadow object has all default properties, whereas if you use the SysCreateObject function, you can pass a setup string to the new shadow object and assign an Object Identifier to the new object.

SysDeregisterObjectClass

The SysDeregisterObjectClass function is used to remove an object class definition from the system.

Syntax

```
RC = SysDeregisterObjectClass(classname)
call SysDeregisterObjectClass classname
```

Parameters

- **classname** — The name of the object class to be removed from the system.

Result

The result is an indication of whether or not the object class was deregistered successfully, as follows:

- 0 — The object class could not be deregistered.
- 1 — The object was deregistered successfully.

Example

Remove the definition of the "MyClass" object class from the system:

```
call SysDeregisterObjectClass "MyClass"
```

SysDestroyObject

The SysDestroyObject function is used to delete a Workplace Shell object from a folder. This has the same effect as dragging the object to the shredder or using the Delete... pop-up menu item for the object.

Syntax

```
RC = SysDestroyObject(name)
call SysDestroyObject name
```

Parameters

- **name** — The name of the object to be deleted. This may be specified as an Object Identifier or, if the object is a Folder or Data File, the fully qualified path name of the object.

Result

The result is an indication of whether or not the object was deleted successfully, as follows:

- **0** — The object could not be deleted.
- **1** — The object was deleted successfully.

Example

Delete the folder created in the first example for SysCreateObject:

```
call SysDeleteObject "<MY_FOLDER>"
```

Note

If you delete a folder, you also delete the contents of the folder. In this example, deleting the folder created in the first example for SysCreateObject also deletes the LaunchPad created in the second example.

SysMoveObject

The SysMoveObject function is used to move a Workplace Shell object from one folder to another. This has the same effect as dragging the object to another folder, or using the **Move...** pop-up menu item for the object.

Syntax

```
RC = SysMoveObject(name,folder)
call SysMoveObject name,folder
```

Parameters

- **name** — The name of the object to be moved. This may be specified as an Object Identifier or, if the object is a Folder or Data File, the fully qualified path name of the object.
- **folder** — The name of the folder that the object is to be moved to. This may be specified as an Object Identifier or the fully qualified path name of the folder.

Result

The result is an indication of whether or not the object was moved successfully, as follows:

- ✗ 0 — The object could not be moved.
- ✗ 1 — The object was moved successfully.

Example

Move the program reference object for an OS/2 Window from the Command Prompts folder to the Desktop:

```
call SysMoveObject "<WP_OS2WIN>","<WP_DESKTOP>"
```

SysOpenObject

The SysOpenObject function is used to open a view of a Workplace Shell object. This has the same effect as double-clicking on an object or using the **Open** pop-up menu item for the object.

Syntax

```
RC = SysOpenObject(name,view,flag)
call SysOpenObject name,view,flag
```

Parameters

- ✗ **name** — The name of the object to be opened. This may be specified as an Object Identifier or, if the object is a Folder or Data File, the fully qualified path name of the object.
- ✗ **view** — The name of the view to be opened. For a standard Workplace Shell object this may be one of the following values:
  - DEFAULT — open the default view of the object
  - ICON — open the icon view (only applies to folders)
  - PALETTE — open the palette view (only applies to palettes or LaunchPad)
  - RUNNING — start an executable program (only applies to program references or executable program files)
  - SETTINGS — open the Settings Notebook for the object

- flag — Indicates how the view is to be opened. This parameter must have one of the following values:
  - 0 — Unconditionally open the specified view of the object
  - 1 — Open the specified view of the object if it is not already open, or surface the existing view if it is already open

Result

The result is an indication of whether or not the view of the object was opened successfully, as follows:

- 0 — The view could not be opened.
- 1 — The view was opened successfully.

Example

Open the icon view of the OS/2 System Folder:

```
call SysOpenObject "<WP_OS2SYS>","icon",1
```

Note

Calling the SysOpenObject function is similar to calling SysSetObjectData and passing a setup string that includes the keyword "OPEN=view". The only difference is that the SysOpenObject function only understands specific views, whereas calling the SysSetObjectData method allows you to specify any view that the object understands using the "OPEN=view" setup string.

SysQueryClassList

The SysQueryClassList function is used to obtain the complete list of object classes that are currently registered with the Workplace Shell.

Syntax

```
call SysQueryClassList name
```

Parameters

- name — The name of a REXX stem variable. This may be specified as the name of the stem variable, the name of the variable followed by a period, or a quoted string containing the name of the stem variable (with or without a trailing period).

Result

The result is a set of compound symbols starting with the specified stem. The symbols returned are:

- **name.0** — The number of registered object classes
- **name.1** through name.n (where n = name.0) — The name of an object class and the name of the dynamic link library containing that class. These two names can be separated using the Parse function.

Example

Display a list of all Workplace Shell object classes:

```
call SysQueryClassList classlist
say "There are "classlist.0" classes registered"
do i = 1 to classlist.0
  parse value classlist.i with class dll
  say "Class "i" is "class", which is in "dll".DLL"
end
```

SysRegisterObjectClass

The SysRegisterObjectClass function is used to register a new object class definition with the system.

Syntax

```
RC = SysRegisterObjectClass(classname,modulename)
call SysRegisterObjectClass classname,modulename
```

Parameters

- **classname** — The name of the object class to be registered with the system.
- **modulename** — The name of the dynamic link library that contains the class to be registered. This is specified as just the one- to eight-character library name — the system assumes a file extension of ".DLL".

Warning

OS/2 Warp requires that all object classes be contained within dynamic link libraries whose names are at most eight characters long.

Result

The result is an indication of whether or not the object class was registered successfully, as follows:

- **0** — The object class could not be registered. Some possible causes of this failure are:
  - The specified dynamic link library could not be found in any directory listed on the LIBPATH= statement in the CONFIG.SYS file.
  - The specified dynamic link library is dependent on another dynamic link library that cannot be located.
  - The specified object class is not contained within the dynamic link library.
  - The specified object class is already registered with the system.
- **1** — The object was registered successfully.

Example

Register the "MyClass" object class, which is contained within the file MYCLASS.DLL, with the system:

```
call SysRegisterObjectClass "MyClass","MYCLASS"
```

SysSaveObject

The SysSaveObject function is used to force a Workplace Shell object to save its properties in permanent storage. For an Abstract object, this is normally in the OS2.INI file. For a File System object, this is normally in the extended attributes of the file or directory.

Normally, Workplace Shell objects automatically save their data whenever it is changed. However, most of the time, the Workplace Shell objects defer saving their data for a period of time. You might call this function to ensure that the changes you made to an object are stored immediately.

Syntax

```
RC = SysSaveObject(name,mode)
call SysSaveObject name,mode
```

Parameters

- **name** — The name of the object whose data is to be saved. This may be specified as an Object Identifier or, if the object is a Folder or Data File, the fully qualified path name of the object.
- **mode** — An indication of how to save the object's data. This parameter may have one of the following values:

- 0 — Save the data immediately.
- 1 — Save the data after the normal delay time.

Result

The result is an indication of whether or not the object's data was saved successfully, as follows:

✗ 0 — The object could not be saved.

✗ 1 — The object was saved successfully.

Example

Save the data for the OS/2 System Folder immediately:

```
call SysSaveObject "<WP_OS2SYS>",0
```

SysSetObjectData

The SysSetObjectData function is used to change the properties of a Workplace Shell object. The SysSetObjectData function can also be used to open a view of an object.

Syntax

```
RC = SysSetObjectData(name,setupstring)
call SysSetObjectData name,setupstring
```

Parameters

✗ **name** — The name of the object whose properties are to be changed. This may be specified as an Object Identifier or, if the object is a Folder or Data File, the fully qualified path name of the object.

✗ **setupstring** — The setup string containing the changes you want to make to the object's properties. See the Setup Strings section earlier in this chapter.

Result

The result is an indication of whether or not the object's properties were changed successfully, as follows:

✗ 0 — The object's properties could not be changed.

✗ 1 — The object's properties were changed successfully.

Examples

Change the Solid Color Palette to be deleteable, then delete the object from the System Setup Folder:

```
call SysSetObjectData "<WP_LORESCLRPAL>","NODELETE=NO"
call SysDestroyObject "<WP_LORESCLRPAL>"
```

Open a tree view of the Desktop Folder:

```
call SysSetObjectData "<WP_DESKTOP>","open=tree"
```

File Access Functions

This group of REXX utility functions can be used to access files and other file system objects.

SysDriveInfo

The SysDriveInfo function is used to retrieve information about a file system device.

Syntax

```
information = SysDriveInfo(driveletter)
```

Parameters

- ✘ **driveletter** — The letter assigned to the drive for which you want to get information.

Result

The result is a string containing the information about the drive. This string contains the following information:

- ✘ **drive letter** — the drive letter (followed by a colon) of the drive whose information is being returned.
- ✘ **free space** — The number of bytes of free space on the drive. If the drive is a read-only device, such as a CD-ROM, the free space is returned as 0.
- ✘ **total size** — The size of the drive, in bytes.
- ✘ **label** — The drive label.

If the specified drive letter is invalid, the drive cannot be accessed, or the drive is a removable medium device (like a diskette or read-write optical drive) and there is no medium in the drive, the returned value is a null string.

Example

Display the information about the D: drive:

```
Drive = "D"
info = SysDriveInfo(Drive)
if Length(info) = 0 Then
  Say "Drive "Drive" is not accessible"
Else Do
  Parse value info with DriveLetter Free Total Label
  say "Drive "DriveLetter" ["Label"]"
  say "Total Space = "Total" Free Space = "Free
End
```

```
The following lines are displayed:
Drive D: [OS/2 3-0]
Total Space = 104617984 Free Space = 20969472
```

SysDriveMap

The SysDriveMap function is used to get a selected list of file system devices in your system.

Syntax

```
map = SysDriveMap([startingdriveletter],[option])
```

Parameters

- **startingdriveletter** — The letter for the first drive you want to appear in the map. If this parameter is not specified, the map starts with the C: drive.
- **option** — The type of map you want. This parameter must have one of the following values:
  - USED — List all drive letters that are assigned. This includes local and remote devices.
  - FREE — List all drive letters that are not assigned.
  - LOCAL — List all drive letters that are assigned to local devices.
  - REMOTE — List all drive letters that are assigned to remote devices.
  - DETACHED — List all drive letters that are assigned to remote devices that have been detached by the network software.

  If this parameter is not specified, USED is assumed.

Result

The result is a string containing the requested list of drive letters. Each drive letter in the list is followed by a colon. The drive letters are separated by spaces.

Examples

Display the list of drive letters that are currently assigned:

```
say SysDriveMap()

The following line is displayed:
C: D: E: F: G:
```

Display the list of drive letters higher than P: that are currently unassigned:

```
say SysDriveMap("Q","free")

The following line is displayed:
Q: R: S: T: U: V: W: X: Y: Z:
```

SysFileDelete

The SysFileDelete function is used to delete a file.

Syntax

```
RC = SysFileDelete(filename)
call SysFileDelete filename
```

Parameters

- ✘ **filename** — The name of the file to be deleted. This may be a fully qualified path specification (including the drive letter) or a partially qualified filename (based on the current directory). The filename cannot include any wildcard characters.

Result

The result is a return code indicating the status of the operation. The result code may be:

- ✘ 0 — The file was deleted successfully.
- ✘ 2 — The specified file does not exist.
- ✘ 3 — The specified path does not exist.
- ✘ 5 — You cannot delete the specified file. This normally means the file is marked as read-only or is on a read-only medium, such as a CD-ROM or a read-only network attached drive.

- ✗ 19 — The specified drive is write protected. You may have specified to delete a file from a write-protected diskette.
- ✗ 26 — The specified drive is not a DOS or OS/2 disk.
- ✗ 32 — You cannot delete the specified file because it is being used by another program.
- ✗ 36 — You cannot delete the specified file because the sharing buffer is exceeded.
- ✗ 87 — You specified an invalid parameter.
- ✗ 206 — The specified filename is invalid.

Examples

Delete the file named "test" in the current directory:

```
call SysFileDelete("test")
```

Delete the file named "d:\sample\test.fil":

```
call SysFileDelete("d:\sample\test.fil")
```

SysFileSearch

The SysFileSearch function is used to search a file for a specified string.

Syntax

```
call SysFileSearch string,filename,line[,options]
```

Parameters

- ✗ **string** — The string to search for.
- ✗ **filename** — The name of the file to be searched. This may be a fully qualified path specification (including the drive letter) or a partially qualified filename (based on the current directory). The filename cannot include any wildcard characters.
- ✗ **line** — The name of a REXX stem variable. This may be specified as the name of the stem variable, the name of the variable followed by a period, or a quoted string containing the name of the stem variable (with or without a trailing period).
- ✗ **options** — A string containing one or more of the following options (in any order):

- C — Only find lines in the file that contain a sequence of characters that exactly match the search string, including the case of the characters. If this option is not specified, the case of the characters in the search string is ignored.
- N — Put the number of the matching line into the resulting variables, in front of the text of the line. If this option is not specified, only the text of the line appears in the result variables.

Result

The result is a set of compound symbols starting with the specified stem. The symbols returned are:

- **line.0** — The number of lines that contain the search string.
- **line.1** through **line.n** (where n = line.0) — The text of a line containing the specified search string. If the N option is specified, these variables contain the line number and the text of the line.

Examples

Find all lines in the CONFIG.SYS file that contain the keyword RUN=:

```
string = "run="
call SysFileSearch string,'d:\config.sys',found
say "There were "found.0" instances of the string "string
do i = 1 to found.0
  say found.i
end

The following lines are displayed:
There were 3 instances of the string run=
rem RUN=C:\STACKER\OS2\FATMGR.EXE
RUN=c:\tcpip\bin\cntrl.exe
RUN=c:\tcpip\bin\vdosctl.exe
```

Find all lines in the CONFIG.SYS file that contain the string “tcpip”. Make the search case sensitive and display the numbers of the lines containing this string:

```
string = "tcpip"
call SysFileSearch string,'d:\config.sys',found,cn
say "There were "found.0" instances of the string "string
do i = 1 to found.0
  say found.i
end
```

```
The following lines are displayed:
There were 10 instances of the string tcpip
9 LIBPATH=.;D:\OS2\DLL;D:\OS2\MDOS;D:\;D:\OS2\APPS\DLL;
D:\MMOS2\DLL;f:\TOOLKT30\
SOM\COMMON\DLL;f:\TOOLKT30\SOM\LIB;f:\TOOLKT30\DLL;
F:\IBMCPP\DLL;c:\dll;c:\tcpip
\dll;c:\TCPIP\UMAIL;D:\VIEWER\DLL;f:\IBMWORKS
10 SET PATH=C:\CMD;D:\OS2;D:\OS2\SYSTEM;D:\OS2\INSTALL;D:\;
D:\OS2\MDOS;D:\OS2\AP
PS;C:\WINDOWS;D:\MMOS2;c:\os2pgm;f:\TOOLKT30\SOM\BIN;
f:\TOOLKT30\BIN;F:\IBMCPP\B
IN;c:\tcpip\bin;c:\TCPIP\UMAIL;D:\VIEWER\BIN;f:\IBMWORKS;
13 SET HELP=D:\OS2\HELP;D:\OS2\HELP\TUTORIAL;D:\MMOS2\HELP;
f:\TOOLKT30\HELP;F:\I
BMCPP\HELP;c:\tcpip\help;c:\TCPIP\UMAIL;D:\VIEWER\HELP;
C:\BA2PRO;f:\IBMWORKS
130 SET ETC=c:\tcpip\etc
131 DEVICE=c:\tcpip\bin\inet.sys
132 DEVICE=c:\tcpip\bin\ifndisnl.sys
133 DEVICE=c:\tcpip\bin\vdostcp.vdd
134 rem DEVICE=c:\tcpip\bin\vdostcp.sys
135 RUN=c:\tcpip\bin\cntrl.exe
136 RUN=c:\tcpip\bin\vdosctl.exe
```

SysFileTree

The SysFileTree function is used to find all files that match the specified search criteria. The SysFileTree function can also be used to change the attributes of a file.

Syntax

```
RC = SysFileTree(filespec,name,[options],[attr],[newattr])
call SysFileTree filespec,name,[options],[attr],[newattr]
```

Parameters

- **filespec** — The file specification to be searched for. The file specification may contain wildcard characters, where "?" indicates any single character and "*" indicates zero or more characters.
- **name** — The name of a REXX stem variable. This may be specified as the name of the stem variable, the name of the variable followed by a period, or a quoted string containing the name of the stem variable (with or without a trailing period).
- **options** — A string containing one or more of the following options (in any order):

- F — Search only for files, not directories, that match the file specification.
- D — Search only for directories, not files, that match the file specification.
- B — Search for both files and directories that match the file specification. If you specify any combination of the F, D, and B options, the last one is used as the search criterion and all of the others are ignored.
- S — Search all subdirectories for matching files or directories also.
- T — Return the last write time and date in the form "yy/mm/dd/hh/mm" instead of the normal format.
- O — Return only the file specification of matched files. Normally, SysFileTree returns the last write date and time, file size, file attributes, and file specification for each matching file. If you specify both the O and the T options, the T option is ignored.

✗ **attrib** — The attribute mask used as one of the search criteria. In order for a file or directory to be returned in the result, it has to match the file specification and its attributes must match this mask, if specified. The attribute mask is a five-character string, where each character represents one of the following file attributes:

- **Archive** (1st character) — The file was changed after it was last backed up.
- **Directory** (2nd character) — This is a directory, not a file.
- **Hidden** (3rd character) — The file or directory is not visible.
- **Read-Only** (4th character) — The file or directory cannot be written to or deleted.
- **System** (5th character) — The file or directory belongs to the operating system.

Each position in the attribute mask may be one of the following characters:

- * — The attribute is ignored.
- + — The file attribute must be set in order for this file or directory to meet the search criteria.
- – — The file attribute must be clear in order for this file or directory to match the search criteria.

- **newattr** — The attribute mask used to change the attributes of the matched files. This mask has the same format as the **attr** mask. For every file that matches the search criteria, this mask specifies how the file's attributes are to be modified, as follows:
  - * — The attribute is not changed.
  - + — The file attribute is set.
  - – — The file attribute is cleared.

Result

The result code (RC) indicates whether or not the operation succeeded. The result code may be the following values:

- 0 — The search completed successfully. The **name** compound variables contain the results of the search.
- 2 — The search failed because of insufficient memory.

The result is a set of compound symbols starting with the specified stem. The symbols returned are:

- **name.0** — The number of files that matched the search criteria.
- **name.1** through **name.n** (where n = name.0) — Information about a file that matched the search criteria. This string may have one of the following formats:
  - **date time size attributes file-specification** — This is the normal output format. The date and time indicate the last time the file was updated, in standard format. The size is the number of bytes contained in the file. The attributes is a five-character string indicating the attributes of the file, in the order ADHRS, where each character may be the letter representing the attribute (if the attribute is set) or a minus sign ("–") if the attribute is clear. The file specification is the fully qualified file specification for the file.
  - **yy/mm/dd/hh/mm size attributes file-specification** — This is the output format you get if you specify the T option. In this case, the time stamp is in a format that is easily sortable.
  - **file-specification** — This is the output format you get if you specify the O option.

Examples

Find all command files anywhere in the subtree that starts in the current directory:

```
filespec = "*.cmd"
RC = SysFileTree(filespec,found)
say "RC = "RC
say "There were "found.0" files that match "filespec
do i = 1 to found.0
  say found.i
end

The following lines are displayed:
RC = 0
There were 3 files that match *.cmd
  5/13/95 10:26a  501 A---- F:\test\box.cmd
  5/10/95  8:18p  288 ----- F:\test\fakeblob.cmd
  5/10/95  8:30p 1112 ----- F:\test\fakerec.cmd
```

Turn off the archive attribute in all *.cmd files in the current directory:

```
filespec = "*.cmd"
call SysFileTree filespec,found,,"+****","-****"
say "There were "found.0" files that match "filespec
do i = 1 to found.0
  say found.i
end

The following lines are displayed:
There were 1 files that match *.cmd
  5/13/95 10:29a  501 ----- F:\test\box.cmd
```

Notice that the archive attribute is now off for the file “box.cmd”.

List all files and directories starting with the current directory:

```
filespec = "*"
call SysFileTree filespec,found,"bso"
say "There were "found.0" files that match "filespec
do i = 1 to found.0
  say found.i
end

The following lines are displayed:
There were 7 files that match *
F:\test\box.cmd
F:\test\fakeblob.cmd
F:\test\fakerec.cmd
F:\test\sample.dir
F:\test\sample.dir\amarks.not
F:\test\sample.dir\amarks1.not
F:\test\sample.dir\amarks2.not
```

SysGetEA

The SysGetEA function is used to retrieve an extended attribute for a file or directory.

Syntax

```
RC = SysGetEA(filename,eaname,eavalue)
call SysGetEA filename,eaname,eavalue
```

Parameters

- ✗ **filename** — The name of the file or directory for which the extended attribute is to be retrieved. This may be a fully qualified path specification (including the drive letter) or a partially qualified filename (based on the current directory). The filename cannot include any wildcard characters.
- ✗ **eaname** — The name of the extended attribute to be retrieved. This name is not case sensitive. Most system-defined extended attribute names start with a period. Some interesting extended attributes are:
  - **.comments** — The information you entered in the Comments entry field on the third File page of the file's Settings Notebook.
  - **.history** — The information maintained by the system and displayed in the History field on the third File page of the file's Settings Notebook.
  - **.icon** — The icon assigned to this file. The information returned by the .icon extended attribute is the binary image of the icon — do not try to display it. You can copy an icon from one file to another by retrieving the .icon EA from one file and writing it to the second file.
  - **.icon1** — The animation icon for a folder.
  - **.keyphrases** — The information you entered in the Key phrases entry field on the third File page of the file's Settings Notebook.
  - **.longname** — The name displayed by the Workplace Shell as the title of the object when the title doesn't fit in the file system's naming convention.
  - **.subject** — The information you entered in the Subject entry field on the first File page of the file's Settings Notebook.
  - **.type** — The data type of the file.

Most extended attributes contain some binary data. Be very careful what you do with the information returned by the SysGetEA function. If you try to display the data, you have to bypass the binary information, or what you display will be unpredictable.

- **eavalue** — The name of a REXX variable to receive the extended attribute information.

Result

The result code (RC) indicates whether or not the operation succeeded. The result code may be the following values:

- 0 — The extended attribute was retrieved successfully and its value placed in the eavalue variable.
- **other** — The extended attribute could not be retrieved. The result code is the OS/2 system return code.

Warning

I could not get the "RC=SysGetEA(...)" form of this function to work. REXX kept giving me a syntax error on the function call, even though it looked good to me. I would suggest using the "call SysGetEA ..." form of this function, and checking the length of the returned string to determine whether or not the extended attribute was retrieved.

Examples

Retrieve and display some extended attributes:

```
filename = "d:\desktop\sample_f\normal_i"
say "Here are some EAs for file "filename":"
eaname = ".longname"
call SysGetEA filename,eaname,"eavalue"
parse var eavalue 5 value
say "The value of the "eaname" EA is '"value"'"
eaname = ".subject"
call SysGetEA filename,eaname,"eavalue"
parse var eavalue 5 value
say "The value of the "eaname" EA is '"value"'"
eaname = ".type"
call SysGetEA filename,eaname,"eavalue"
parse var eavalue 11 value
say "The value of the "eaname" EA is '"value"'"
```

```
The following lines are displayed:
Here are some EAs for file d:\desktop\sample_f\normal_i:
The value of the .longname EA is 'Normal Icon'
The value of the .subject EA is 'This is a normal icon'
The value of the .type EA is 'Plain Text'
```

Note

The .longname and .subject extended attributes contain 4 bytes of binary data before the text, so I had to skip over those 4 bytes using the Parse command before displaying the value. Similarly, the .type extended attribute contains 10 bytes of binary data preceding the actual type code. If you are going to display extended attributes using the SysGetEA function, you will have to experiment a little to determine how many bytes of binary data precede the value you want to display.

Copy the icons from the OS/2 System folder to my Sample folder:

```
filename1 = "d:\ira's_de\os!2_sys"
filename2 = "d:\desktop\sample_f"
eaname = ".icon"
call SysGetEA filename1,eaname,"eavalue"
call SysPutEA filename2,eaname,eavalue
eaname = ".icon1"
call SysGetEA filename1,eaname,"eavalue"
call SysPutEA filename2,eaname,eavalue
```

Tip

You can copy extended attributes from one file or directory to another using the SysGetEA and SysPutEA functions. However, the Workplace Shell is not informed of the changes, and you may not see the changes to the properties that you affect until the next time you boot your system. For example, you can change the .icon extended attribute for a file or directory. But, you won't see the change unless you open the Settings Notebook for the data file object, at which time the Workplace Shell updates its properties from the extended attributes.

Warning

I noticed that when I set the .icon2 extended attribute for a folder the new icon didn't actually show up. I opened the Settings Notebook for the folder, and it still wasn't updated. I looked on General Page 2, and it showed the old icon. When I selected the Edit... push button, the icon it displayed in the Icon Editor was the new icon! If you change the .icon2 extended attribute, it won't actually take effect until the folder object goes dormant and gets reawakened. This might not happen until the system is shut down and restarted.

Modifying Configuration Files

Warning

Take great care when modifying the OS/2 configuration files. The Workplace Shell, the Presentation Manager, other components of OS/2, and application programs maintain important information in these files. If you delete or change any information necessary for the operation of OS/2, you might make the system unusable.

SysIni

The SysIni function is used to retrieve data from the OS/2 configuration files (OS2SYS.INI and OS2.INI) or change data in the configuration files.

Syntax

The SysIni function actually has six different modes of operation, each with its own syntax. The parameters used in the different function calls are listed below.

Here is the syntax for each mode:

1 Retrieve a single key value.

```
keyvalue = SysIni([filename],application,key)
```

2 Set a single key value.

```
RC = SysIni([filename],application,key,value)
call SysIni [filename],application,key,value
```

3 Delete a single key.

```
RC = SysIni([filename],application,key,"DELETE:")
call SysIni [filename],application,key,"DELETE:"
```

4 Delete an application and all associated keys.

```
RC = SysIni([filename],application,["DELETE:"])
call SysIni [filename],application,["DELETE:"]
```

5 Retrieve all keys for an application.

```
RC = SysIni([filename],application,"ALL:",information)
call SysIni [filename],application,"ALL:",information
```

6 Retrieve the names of all applications.

```
RC = SysIni([filename],"ALL:",information)
call SysIni [filename],"ALL:",information
```

Parameters

- ✘ **filename** — The name of the configuration file to be accessed. This parameter may be the name of a private configuration file or must be one of the following values:

  - **USER** — Access the user configuration file, normally OS2.INI.
  - **SYSTEM** — Access the system configuration file, normally OS2SYS.INI.
  - **BOTH** — Access both the user and system configuration files when retrieving data. Access the user configuration file when setting data.

  If this parameter is not specified, the user configuration file is accessed.

- ✘ **application** — The application name used to store information in the configuration file.
- ✘ **key** — The key name used to store information in the configuration file.
- ✘ **value** — The value to be stored in the configuration file entry.
- ✘ **information** — The name of a REXX stem variable. This may be specified as the name of the stem variable, the name of the variable followed by a period, or a quoted string containing the name of the stem variable (with or without a trailing period).

Result

Each mode has its own result, as follows:

1. Retrieve a single key value: The result (key value) is the returned key word value or "ERROR:" if the value could not be retrieved.
2. Set a single key value: The result (RC) is an empty string if the value was set successfully, or "ERROR:" if the value could not be set.
3. Delete a single key: The result (RC) is an empty string if the key was deleted successfully, or "ERROR:" if the key could not be deleted.
4. Delete an application and all associated keys: The result (RC) is an empty string if the application and its keys were deleted successfully, or "ERROR:" if the keys could not be deleted.

5 Retrieve all keys for an application: The result (RC) is set to an empty string, information.0 is set to the number of keys for this application, and information.1 through information.n (where n is information.0) are set to the names of the keys for this application. If the keys cannot be retrieved, the result (RC) is set to "ERROR:".

6 Retrieve the names of all applications: The result (RC) is set to an empty string, information.0 is set to the number of applications, and information.1 through information.n (where n is information.0) are set to the names of the applications. If the names of the applications cannot be retrieved, the result (RC) is set to "ERROR:".

Examples

Retrieve the value of the key "CometCursorEnabled" for application "PM_ControlPanel" (this indicates whether or not you enabled the Comet Cursor feature):

```
say "Comet Cursor = "||,
     SysIni("user","PM_ControlPanel","CometCursorEnabled")
```

```
The following line is displayed:
Comet Cursor = 1
```

Set the value of the key "CometCursorEnabled" for application "PM_ControlPanel" to 0 (this disables the Comet Cursor feature the next time you start your system):

```
call SysIni "user","PM_ControlPanel","CometCursorEnabled","0"
```

Delete the key "CometCursorEnabled" from application "PM_ControlPanel":

```
call SysIni "user","PM_ControlPanel","CometCursorEnabled",,
            "DELETE:"
```

Delete all keys for application "PM_ColorCircle":

```
call SysIni "user","PM_ColorCircle","DELETE:"
```

Retrieve all keys for application "PM_Colors" (PM_Colors is the set of colors used by Presentation Manager):

```
call SysIni "user","PM_Colors","ALL:","information"
say information.0" keys returned"
do i=1 to information.0
  say information.i
end
```

```
The following lines are displayed:
45 keys returned
Display
ShadowHiliteBgnd
ShadowHiliteFgnd
ShadowText
EntryField
       ...
FolderIconTextBackground
DesktopIconTextBackground
DesktopIconText
```

Retrieve the names of all for applications:

```
call SysIni "user","ALL:","information"
say information.0" applications returned"
do i=1 to information.0
  say information.i
end

The following lines are displayed:
114 applications returned
Passport/2
IBMWORKS
PM_Default_National
PM_ED_HOOKS
DLP_GENERAL
IconEdit 2.0
       ...
PicView
DlgEdit
BackAgain_V2
PMJPEG
PMREXX_Interface
```

SysMkDir

The SysMkDir function is used to create a directory.

Syntax

```
RC = SysMkDir(directory)
call SysMkDir directory
```

Parameters

✗ **directory** — The name of the directory to be created. This may be a fully qualified path specification (including the drive letter) or a partially qualified filename (based on the current directory).

You can only create a single level of directory with a call to SysMkDir. For example, if you want to create **D:\FIRST\SECOND,** the directory **D:\FIRST** must already exist before you try to create SECOND.

Result

The result is a return code indicating the status of the operation. The result code may be:

- ✗ 0 — The directory was created successfully.
- ✗ 3 — The specified path does not exist. This means that you tried to create a directory based on a nonexistent directory.
- ✗ 5 — You do not have access to the specified directory. You may have specified to create a directory on a network attached drive for which you do not have write access to the required directory or you may have tried to create a directory that already exists.
- ✗ 19 — The specified drive is write protected. You may have specified to create a directory on a write-protected diskette, a CD-ROM, or a read-only network attached drive.
- ✗ 26 — The specified drive is not a DOS or OS/2 disk.
- ✗ 87 — You specified an invalid parameter.
- ✗ 108 — The specified drive is locked.
- ✗ 206 — The specified directory name is invalid.

Examples

Create the "test" directory on the D: drive:

```
call SysMkDir("d:\test")
```

Create the "first/second" directory on the C: drive::

```
call SysMkDir("c:\first")
call SysMkDir("c:\first\second")
```

SysPutEA

The SysPutEA function is used to write an extended attribute for a file.

Syntax

```
RC = SysPutEA(filename,eaname,eavalue)
call SysPutEA filename,eaname,eavalue
```

Parameters

- **filename** — The name of the file for which the extended attribute is to be written. This may be a fully qualified path specification (including the drive letter) or a partially qualified filename (based on the current directory). The filename cannot include any wildcard characters.
- **eaname** — The name of the extended attribute to be written. This name is not case sensitive. Most system-defined extended attribute names start with a period. See the description of the SysGetEA function for a list of some of the system-defined extended attributes.
- **eavalue** — The name of a REXX variable containing the information to be written as the extended attribute.

Warning

Most extended attributes contain some binary data. Be very careful how you format the information to be written by the SysPutEA function. If you write invalid information for a critical system-defined extended attribute (such as the Workplace Shell information for the Desktop Folder), you might make your system unuseable.

Result

The result code (RC) indicates whether or not the operation succeeded. The result code may be the following values:

- **0** — The extended attribute was written successfully.
- **other** — The extended attribute could not be written. The result code is the OS/2 system return code.

Example

Copy the icons from the OS/2 System folder to my Sample folder:

```
filename1 = "d:\ira's_de\os!2_sys"
filename2 = "d:\desktop\sample_f"
eaname = ".icon"
call SysGetEA filename1,eaname,"eavalue"
call SysPutEA filename2,eaname,eavalue
eaname = ".icon1"
call SysGetEA filename1,eaname,"eavalue"
call SysPutEA filename2,eaname,eavalue
```

SysRmDir

The SysRmDir function is used to delete a directory.

The directory to be deleted must be empty. It must not contain any data files or any other directories.

Syntax

```
RC = SysRmDir(directory)
call SysRmDir directory
```

Parameters

- **directory** — The name of the directory to be deleted. This may be a fully qualified path specification (including the drive letter) or a partially qualified filename (based on the current directory).

Result

The result is a return code indicating the status of the operation. The result code may be:

- 0 — The directory was deleted successfully.
- 2 — The specified directory does not exist.

  I got this return code when I tried deleting a non-existent directory from a CD-ROM.

- 3 — The specified directory does not exist.

  I got this return code when I tried deleting a non-existent directory from a diskette or a fixed disk.

- 5 — You do not have access to the specified directory. You may have specified to delete a directory that is not empty or is on a network attached drive for which you do not have write access.
- 16 — The specified directory is the current directory for an OS/2 session.
- 19 — The specified drive is write protected. You may have specified to delete a directory from a write-protected diskette, a CD-ROM, or a read-only network attached drive.
- 26 — The specified drive is not a DOS or OS/2 disk.
- 87 — You specified an invalid parameter.

- 108 — The specified drive is locked.
- 206 — The specified directory name is invalid.

Example

Remove the "test" directory from the D: drive:

```
call SysRmDir("d:\test")
```

SysSearchPath

The SysSearchPath function is used to find a file in the set of directories specified by an environment variable.

Syntax

```
filespec = SysSearchPath(path,filename)
```

Parameters

- **path** — The name of an environment variable containing the list of directories to be searched for the specified file. This variable must be formatted similarly to the PATH or DPATH variables, with the directory names separated by semicolons.

  Warning

  This parameter must be specified as an uppercase string.
- **filename** — The name of the file to be located. This must be an unqualified filename (that is, do not specify any path information).

Result

The result is the fully qualified name of the file, if it is found. If the file is not found in any directory specified by the environment variable, the result is an empty string.

Example

Find the "cmdph.hlp" file in any directory listed in the "HELP" environment variable:

```
say SysSearchPath("HELP","cmdph.hlp")

The following line is displayed:
D:\MMOS2\HELP\cmdph.hlp
```

SysSetIcon

The SysSetIcon function is used to associate an icon with a file.

Note

The icon information is extracted from the specified icon file and written to the destination file's .icon extended attribute.

Uncensored

The Workplace Shell is not informed about the change to the file's .icon extended attribute. If the destination file is in an open folder, you will not see the icon change until you close the folder, let it go dormant, and reopen the file. Alternatively, you could open the Settings Notebook for the destination file — this makes the Workplace Shell refresh its properties, including the icon, from the file system.

Syntax

```
RC = SysSetIcon(filename,iconfilename)
call SysSetIcon filename,iconfilename
```

Parameters

- **filename** — The name of the file for which the icon is to be associated. This may be a fully qualified path specification (including the drive letter) or a partially qualified filename (based on the current directory). The filename cannot include any wildcard characters.
- **iconfilename** — The name of the .ICO file containing the icon. This may be a fully qualified path specification (including the drive letter) or a partially qualified filename (based on the current directory). The file name cannot include any wildcard characters.

Result

The result is an indication of whether or not the icon file was associated with the destination file, as follows:

- 0 — The icon could not be associated with the destination file. Some possible causes of a failure are:
  - The file specified by **filename** does not exist.
  - The file specified by **iconfilename** does not exist.
  - The file specified by **filename** cannot be written to because it is on a read-only drive or the file system doesn't support extended attributes.
- 1 — The object was associated with the destination file successfully.

Example

Associate the "myicon.ico" file with "test.exe" (both files are in the current directory):

```
call SysSetIcon("test.ext","myicon.ico")
```

SysTempFileName

The SysTempFileName function is used to create a unique filename that can be used for a temporary file.

Syntax

```
filespec = SysTempFileName(template[,filter])
```

Parameters

- **template** — The template for the filename to be created. This may be a fully qualified path specification (including the drive letter) or a partially qualified filename (based on the current directory). The filename portion of the template must contain from one to five filter characters. The path portion of the template (if specified) cannot contain any filter characters.
- **filter** — The filter character used in the template. This can be any single character. If this parameter is not specified, the filter character is the question mark ("?") character.

Note

The SysTempFileName function generates a five-digit random number and replaces the filter characters in the template with these digits. If the resulting filename already exists, the SysTempFileName function generates a new random number and tries again.

Result

The result is the fully qualified name of a file that matches the template specification and doesn't already exist. If the parameters are invalid or if SysTempFileName cannot generate a unique filename that doesn't already exist, the result is an empty string.

Example

Generate two temporary filenames:

```
tempname1 = SysTempFileName("f:\test\temp????")
tempname2 = SysTempFileName("new@@@@@","@")
say "First temporary file name is '"tempname1"'"
say "Second temporary file name is '"tempname2"'"

The following lines are displayed:
First temporary file name is 'f:\test\temp2842'
Second temporary file name is 'd:\new82038'
```

SysWaitNamedPipe

The SysWaitNamedPipe function is used to wait for an instance of a named pipe to become available. This function is called only after trying to open a named pipe and receiving a return code 231, indicating "pipe busy."

Syntax

```
result = SysWaitNamedPipe(pipename[,timeout])
```

Parameters

- **pipename** — The name of the named pipe to be waited for. The name must be of the form \PIPE\name.
- **timeout** — The maximum time, in milliseconds, to wait for an instance of the named pipe to become available. If this parameter is not specified or is zero, the SysWaitNamedPipe function will wait up to the number of milliseconds specified when the named pipe was created. If this paraemter is specified as -1, the SysWaitNamedPipe function will wait indefinitely.

Result

The result is a return code indicating the status of the operation. The result code may be:

- 0 — The named pipe instance is available.

  Warning

  Even though the DosWaitNamedPipe function indicates that the named pipe instance is available, by the time you try to open the named pipe, the pipe may become busy again.

- 2 — The specified named pipe does not exist.
- 95 — An interrupt occurred while waiting for the named pipe instance. This probably means that you pressed Ctrl-Break or Ctrl-C while the SysWaitNamedPipe function was waiting for the named pipe to become available.
- 231 — The named pipe instance did not become available during the time specified by the time-out parameter.

Example

Open a named pipe. If the named pipe instance is not available, wait for it, then open the named pipe:

```
PipeName = "\PIPE\MYPIPE"
Do Until RC<>231
  Parse Value Stream(PipeName,"C","OPEN") With State ":" RC
  If RC=231 Then Call SysWaitNamedPipe(PipeName,-1)
End
```

Miscellaneous Functions

These are the remaining REXX utility functions that just didn't fit into any of the previous categories.

SysDropFuncs

The SysDropFuncs function is used to delete all loaded REXX functions.

Warning

If you invoke this function, all previously loaded REXX functions are unavailable to any REXX script until you load them again. This includes the REXX utility functions.

Syntax

```
call SysDropFuncs
```

Parameters

None

Result

All previously loaded REXX functions are unloaded and are unavailable until you invoke the SysAddFuncs function to load them again.

Example

Unload all REXX functions:

```
call SysDropFuncs
```

SysOS2Ver

The SysOS2Ver function is used to retrieve the version of OS/2 that is running.

Syntax

```
version = SysOS2VER()
```

Parameters

None

Result

The result is a string containing the current OS/2 version in the form "n.nn". For OS/2 Warp, the version returned is "2.30".

Example

Display the current OS/2 version:

```
say "OS/2 Version "SysOS2Ver" is running."
```

```
The following line is displayed:
OS/2 Version 2.30 is running.
```

Summary

There you have it. If you followed my descriptions, you are now an expert in the use of the REXX utility functions. But, what can you use them for? Each function is interesting, but there must be some trick to string them together and make something useful.

CD-ROM

My suggestion is that you look at some of the REXX scripts we have provided on the CD-ROM included with this book. Some of them are pretty simple; others are complex and may give you some good ideas.

Developer's Desktop

Here is an example of one of the REXX scripts I use on my systems. I wanted to be able to play a random sound clip when my system starts and another random clip when I shut the system down. One obvious way would be to copy a random .WAV file into files named STARTUP.WAV and SHUTDOWN.WAV, but that is boring. I wanted something better.

One of the annoying things I noticed about the system was that when one system sound is playing and another system event occurs, the first sound stops. My start-up sound was being interrupted by the opening of the Desktop folder. I wanted to fix that at the same time.

I invoke the REXX script shown in this example from my STARTUP.CMD. I set the System Startup sound to "none" so that the system doesn't play anything. My script plays the start-up sound for me and sets up the shutdown sound effect. When the Desktop Folder opens, the system sound interrups my start-up sound, then resumes playing my sound. That is what I wanted.

```
/* Select Startup/Shutdown Sound Effects */
call RxFuncAdd 'SysLoadFuncs','RexxUtil','SysLoadFuncs'
call SysLoadFuncs
'@echo off'

/* Disable Startup Sound Effect */
call SysIni 'd:\mmos2\mmpm.ini','MMPM2_AlarmSounds',,
            '7','#System startup#60'X2C('00')

/* Select Shutdown Sound Effect */
call SysFileTree 'c:\mmos2\sounds\shutdown\*.wav',,
                  shutdown,'o'
ix=Random(1,shutdown.0)
call SysIni 'd:\mmos2\mmpm.ini','MMPM2_AlarmSounds',,
            '8',shutdown.ix'#System shutdown#60'X2C('00')

/* Select Startup Sound Effect */
call SysFileTree 'c:\mmos2\sounds\startup\*.wav',,
                  startup,'o'
ix=Random(1,startup.0)

/* Play Startup Sound Effect */
'call play file='startup.ix
```

Note

There are a few things to note about this example:

- ✘ I found (experimentally) that the MMPM.INI file is a standard configuration file and that the "MMPM2_AlarmSounds" application is where the System Sound definitions are stored. Key "7" keeps the definition for the System Startup sound effect, while key "8" is where the System Shutdown sound effect is stored.
- ✘ The format of the information for a system sound consists of the following fields:
    - The name of the .WAV file to be played (if any)
    - A pound sign character ("#")
    - The name of the system sound
    - A pound sign character ("#")
    - The volume that the sound is to be played at as a decimal number between 1 and 100 (a percentage of the full volume)
    - A null byte

- ✗ I put my selection of start-up sounds into the directory "c:\mmos2\sounds\startup" and my shutdown sounds into "c:\mmos2\sounds\shutdown".
- ✗ The SysFileTree function builds a list of files contained in the selected directory. I select a random file to be played by using the Random function, specifying that I want a random number between 1 and the number of files returned by SysFileTree.

You may not want to perform these exact functions, but this example illustrates how to select a random file out of a directory. It also demonstrates how to use the SysIni function to write information to a configuration file, and how to mix binary information with simple text when constructing the data to be written.

Ira H. Schneider is an Advisory Programmer on the Workplace Shell Development Team. He has been instrumental in developing OS/2 Warp, including performance improvements and functional enhancements to the Workplace Shell. He has also worked on the OS/2 Swapper and on OS/2 System Initialization, and he was Lead Architect for OS/2 1.2. Ira has been with IBM since 1969.

Chapter 9

OS/2 System Objects

Still More Objects at Your Disposal

Many standard objects are installed in OS/2 Warp to assist you in running your system. These objects are stored in standard folders installed on your Desktop. In this chapter, I will describe these objects so that you know where to look for their key functions.

Objects in the OS/2 System Folder

The OS/2 System folder (see Figure 9-1) contains the main folders and objects necessary for controlling OS/2 Warp. You will find most of the important functions in these folders.

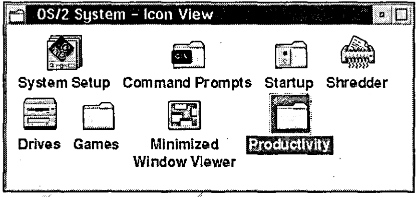


Figure 9-1. The OS/2 System Folder

System Setup folder

The System Setup folder contains objects used to configure your system the way you want it and to customize the Workplace Shell to work for you. These objects have all been described in other chapters of this book. Table 9-1 shows you where to look for information about each of the objects in the System Setup folder:

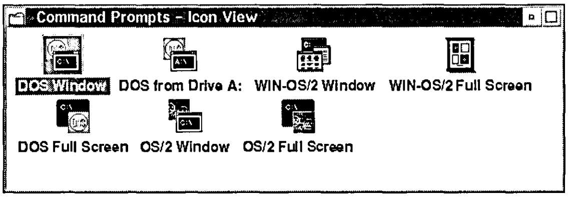
Table 9-1. Index to Coverage of System Setup Folder Objects

Object	Function	Where Described
Add Programs	Migrate OS/2, DOS, and Windows programs to the Workplace Shell	Chapter 1
Country	Customize OS/2 Warp for your country	Chapter 7
Create Utility Diskettes	Create bootable OS/2 Warp diskettes	Chapter 1
Device Driver Install	Install drivers for display devices	Chapter 1
Font Palette	Customize the fonts used by windows and add or remove fonts from OS/2 Warp	Chapter 6
Keyboard	Customize your keyboard	Chapter 7
Mixed Color Palette (256 colors)	Customize the colors used by windows	Chapter 6
Mouse	Customize your mouse	Chapter 7
Scheme Palette	Customize the colors and fonts for windows and folders and the backgrounds for folders.	Chapter 6
Selective Install	Install portions of OS/2 Warp	Chapter 1
Selective Uninstall	Remove portions of OS/2 Warp	Chapter 1
Solid Color Palette (16 colors)	Customize the colors used by windows	Chapter 6
Sound	Customize system sounds	Chapter 7
Spooler	Customize the Print Spooler	Chapter 7
System	Customize your display and other Workplace Shell options	Chapter 7
System Clock	Customize and display the System Clock	Chapter 7
WIN-OS/2 Setup	Customize WIN-OS/2 sessions	Chapter 7

Command Prompts folder

As shown in Figure 9-2, the Command Prompts folder contains objects that you can use to start various types of programs, including OS/2, DOS, and WIN-OS/2 full-screen or windowed command prompts. Each object opens a different type of window.

Figure 9-2. Command Prompts Folder



OS/2 command prompts

There are two types of OS/2 command prompts: full-screen and windowed. The two types of command prompts share certain characteristics.

The SET OS2_SHELL= statement in the CONFIG.SYS file specifies the program that is started when you invoke an OS/2 command prompt. Normally, this is CMD.EXE. However, you can replace the OS/2 command processor with any program you choose by changing this statement in the CONFIG.SYS file.

Tip

When OS/2 Warp is installed, the SET OS2_SHELL= statement specifies to start CMD.EXE without any parameters. You can have CMD.EXE execute a command, or even a batch file, including a REXX script, before presenting you with the command prompt.

Developer's Desktop

For example, I like to have my command prompt windows have 32 lines instead of the standard 24 lines. I accomplish this by having CMD.EXE automatically execute a MODE command to set the display to 32 line mode. To do so, I put the following statement into the CONFIG.SYS file:

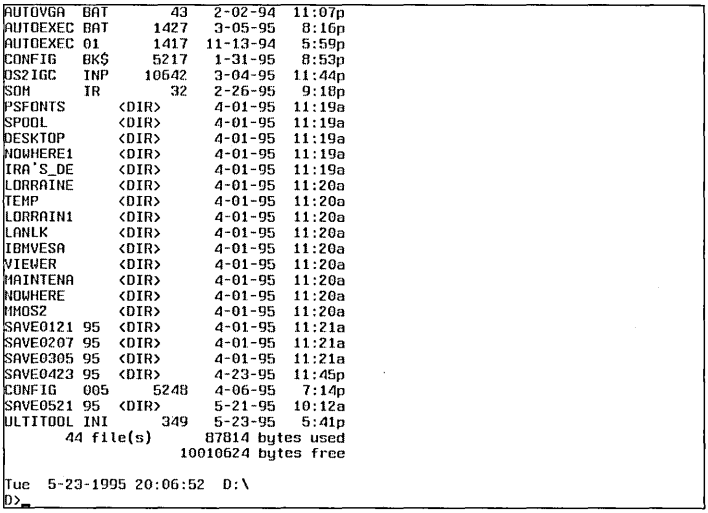
```
SET OS2_SHELL=D:\OS2\CMD.EXE /k mode 80,32
```

OS/2 Full Screen. The OS/2 Full Screen object opens a full-screen window containing an OS/2 command prompt. You may invoke any OS/2, DOS, or Windows program from this command prompt. As shown in Figure 9-3, this window takes up the entire screen and doesn't have a title bar or sizing borders.

You can return to the Workplace Shell at any time by pressing Ctrl-Esc. The window will be minimized and your Desktop will be visible again, with the Window List displayed.

To return to the full-screen command prompt at any time, double-click on its minimized icon or on its title in the Window List.

When you are finished using the full-screen command prompt window, type **exit** at the prompt and the window will close.



```
AUTOVGA  BAT        43   2-02-94  11:07p
AUTOEXEC BAT      1427   3-05-95   8:16p
AUTOEXEC 01       1417  11-13-94   5:59p
CONFIG   BK$      5217   1-31-95   8:53p
OS2IGC   INP     10642   3-04-95  11:44p
SOM      IR         32   2-26-95   9:18p
PSFONTS      <DIR>       4-01-95  11:19a
SPOOL        <DIR>       4-01-95  11:19a
DESKTOP      <DIR>       4-01-95  11:19a
NOWHERE1     <DIR>       4-01-95  11:19a
IRA'S_DE     <DIR>       4-01-95  11:19a
LORRAINE     <DIR>       4-01-95  11:20a
TEMP         <DIR>       4-01-95  11:20a
LORRAIN1     <DIR>       4-01-95  11:20a
LANLK        <DIR>       4-01-95  11:20a
IBMVESA      <DIR>       4-01-95  11:20a
VIEWER       <DIR>       4-01-95  11:20a
MAINTENA     <DIR>       4-01-95  11:20a
NOWHERE      <DIR>       4-01-95  11:20a
MMOS2        <DIR>       4-01-95  11:20a
SAVE0121 95  <DIR>       4-01-95  11:21a
SAVE0207 95  <DIR>       4-01-95  11:21a
SAVE0305 95  <DIR>       4-01-95  11:21a
SAVE0423 95  <DIR>       4-23-95  11:45p
CONFIG   005      5248   4-06-95   7:14p
SAVE0521 95  <DIR>       5-21-95  10:12a
ULTITOOL INI       349   5-23-95   5:41p
       44 file(s)      87814 bytes used
                    10010624 bytes free

Tue  5-23-1995 20:06:52  D:\
D>_
```

Figure 9-3.
OS/2 Full Screen Command Prompt

OS/2 Window. The OS/2 Window object opens a window containing an OS/2 command prompt. From this command prompt, you may invoke any OS/2 program that can run in a window or any DOS or Windows program that can run in a display mode compatible with your Desktop. For example, with many display adapters, you cannot invoke a DOS program that uses Super VGA graphics (that is, a resolution higher than 640 × 480 pixels) from a windowed command prompt. This is a limitation of the display hardware and the display device drivers. However, you can run these programs from a full-screen command prompt. See the *DOS Window* section below for the limitations on DOS programs running in a window.

When you are finished using the command prompt window, type **exit** at the prompt and the window will close.

Tip

You can set the physical characteristics for all OS/2 and DOS windowed sessions by holding the **Shift** key while manipulating the window. You can save the following characteristics:

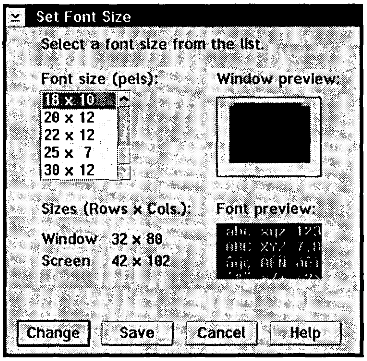
- Open window maximized or restored: Hold the **Shift** key while double-clicking on the title bar for the window. The window will alternate between maximized and restored. The last state of the window that you set while holding the Shift key will be used for all other OS/2 or DOS Windows that you open from the OS/2 Window or DOS Window objects, or by starting a program in an OS/2 or DOS Window.

- Position of the window: Hold the **Shift** key while moving the window around on the screen. The last position that you set while holding the Shift key will be used for all other OS/2 or DOS Windows.

- Size of the window: If the window is not maximized, you can hold the **Shift** key while resizing it. The last size that you set this way will be used for all other OS/2 or DOS Windows.

An OS/2 Window has several useful items in its pop-up menu:

- **Font Size** — Displays the Set Font Size dialog, as shown in Figure 9-4. This dialog allows you to set the font for this window or for all OS/2 and DOS text windows.



Set Font Size

Select a font size from the list.

Font size (pels):

18 x 10
20 x 12
22 x 12
25 x 7
30 x 12

Sizes (Rows x Cols.):

Window 32 x 80
Screen 42 x 102

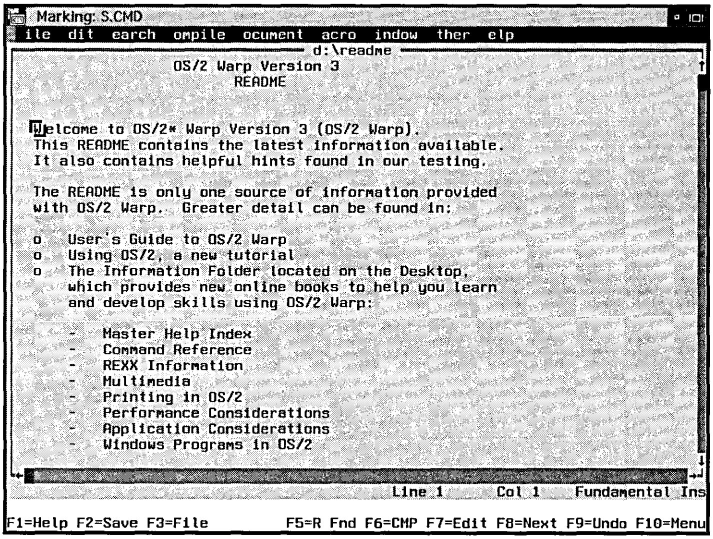
Figure 9-4. Set Font Size Dialog

- **Full-screen** — Changes this window to a full-screen window. Once you select this option, you can no longer request the pop-up menu for the window to make other changes.

- **Scroll** — Toggles between a scrolling and nonscrolling window. When the window is a scrolling window, the title of the window changes from OS/2 Window to Scrolling OS/2 Window. In a nonscrolling window, the title changes back to OS/2 Window. In a scrolling window, you can use the cursor control keys to scroll up, down, left, or right. In a nonscrolling window, the cursor control keys have their normal effect and you have to click on the scroll bars to scroll the window.

- ✘ Mark — Allows you to mark text or graphics to be copied to the clipboard. The cursor appears as a partial box, as shown in Figure 9-5. To mark a block of text or graphics in the window, position the cursor to one corner of the block of text to be marked, press and hold mouse button 1, move the cursor to the diagonally opposite corner of the block to be marked, and release the mouse button. The marked area is now highlighted (as shown in Figure 9-6), and you can use the Copy option to copy this block of text or graphics to the clipboard.

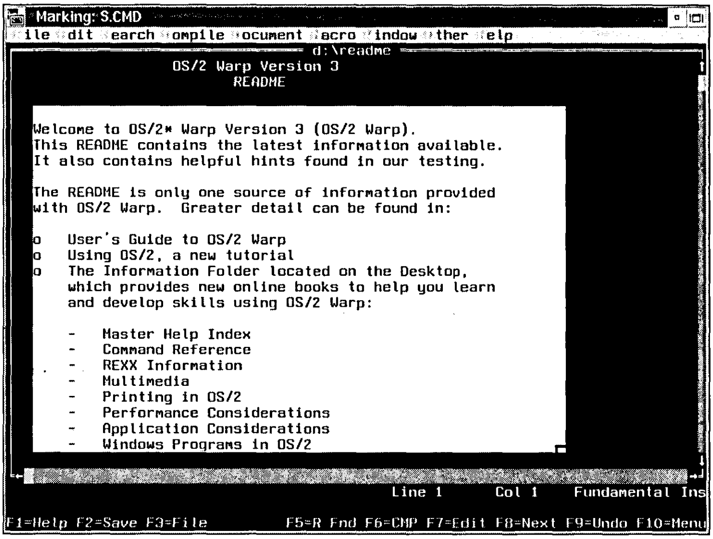
Figure 9-5. Before Marking Text



```
                    OS/2 Warp Version 3
                           README

Welcome to OS/2* Warp Version 3 (OS/2 Warp).
This README contains the latest information available.
It also contains helpful hints found in our testing.

The README is only one source of information provided
with OS/2 Warp.  Greater detail can be found in:

o   User's Guide to OS/2 Warp
o   Using OS/2, a new tutorial
o   The Information Folder located on the Desktop,
    which provides new online books to help you learn
    and develop skills using OS/2 Warp:

    -   Master Help Index
    -   Command Reference
    -   REXX Information
    -   Multimedia
    -   Printing in OS/2
    -   Performance Considerations
    -   Application Considerations
    -   Windows Programs in OS/2

                                          Line 1     Col 1    Fundamental Ins

F1=Help F2=Save F3=File         F5=R Fnd F6=CMP F7=Edit F8=Next F9=Undo F10=Menu
```

- ✘ Copy — Copies the marked text or graphics to the clipboard. If you have not marked an area, this item is not selectable.
- ✘ Copy All — Copies all of the text or graphics in the window to the clipboard.
- ✘ Paste — Pastes the text or graphics from the clipboard into the window. If the window is a command prompt window, you can paste in text from the clipboard. The text is handled the same as commands that you type after the prompt.

Figure 9-6.
After Marking Text



```
                    d:\readme
                     OS/2 Warp Version 3
                           README

Welcome to OS/2* Warp Version 3 (OS/2 Warp).
This README contains the latest information available.
It also contains helpful hints found in our testing.

The README is only one source of information provided
with OS/2 Warp.  Greater detail can be found in:

o   User's Guide to OS/2 Warp
o   Using OS/2, a new tutorial
o   The Information Folder located on the Desktop,
    which provides new online books to help you learn
    and develop skills using OS/2 Warp:

    -   Master Help Index
    -   Command Reference
    -   REXX Information
    -   Multimedia
    -   Printing in OS/2
    -   Performance Considerations
    -   Application Considerations
    -   Windows Programs in OS/2

                                        Line 1     Col 1    Fundamental Ins

F1=Help F2=Save F3=File        F5=R Fnd F6=CMP F7=Edit F8=Next F9=Undo F10=Menu
```

The Set Font Size dialog (see Figure 9-4) enables you to select the size of the font for the text displayed in an OS/2 or DOS Window.

To select a new font, choose it from the Font size list box. The fonts are listed as vertical size × horizontal size, where the vertical and horizontal sizes are the height and width of each character in pels (or pixels).

The Window preview window shows you how a maximized window will look using this font. The size of the window is calculated based on the current video mode (that is, the number of rows and columns you chose using the MODE command). The dark area in the center represents the maximized window, the colored area surrounding it represents the Desktop. From this picture you can get an idea how big a window using this font will look on the Desktop.

The information below the Sizes heading tells you how many rows and columns the window is currently set to and the maximum number of rows and columns that the window could be set to and still fit on the Desktop, using the currently selected font. As you choose different fonts, the values for the screen rows and columns will change, reflecting the size of the font. The Font preview window shows you what the characters in the selected font look like.

Using the information presented by this dialog, you can get a good idea what effect the font you choose will have on the window.

If you want to use the font you have selected, click on the **Change** push button. This will close the Set Font Size dialog and return you to the window, with the new font selected. This font change will be effective only until you close the window. If you want this font to be used for all OS/2 and DOS Windows, click on the **Save** push button. This will close the Set Font Size dialog and return you to the window, with the new font selected. This font change will now be effective for all new OS/2 and DOS Windows.

If you don't want to change the font, click on the **Cancel** push button. This will close the Set Font Size dialog and return you to the window with the original font still selected.

From now on, any time you open an OS/2 Window, it will have the number of rows and columns specified by the MODE command, it will be positioned where you left it, and it will be maximized (so it will automatically adjust its size to match the number of rows and columns being displayed). Also, if you open a DOS Window, it will be maximized and positioned in the same place. However, the number of rows and columns for the DOS Window is not controlled by the MODE command on the OS2_SHELL statement.

Customizing OS/2 Windows

You can combine all of the techniques I have shown you to completely customize your OS/2 Windows, as follows:

1. Add the parameter **/k mode columns rows** to the end of the SET OS2_SHELL= statement in the CONFIG.SYS file, where columns is the width of the window in characters and rows is the height of the window in lines. The default is 25 lines of 80 characters. Restart your system after changing the CONFIG.SYS file.
2. Open an OS/2 Window.
3. Hold the **Shift** key and double-click on the title bar of the window or click on the **Maximize** push button.
4. Hold the **Shift** key and position the window where you want it on the screen.
5. Select the **Font size...** item from the window's pop-up menu.
6. Select the font that makes the window look the best to you and click on the **Save** push button.

DOS command prompts

There are two types of DOS command prompts: full-screen and windowed. The two types of command prompts share some common characteristics.

The program starts when you invoke a DOS command prompt specified by the SET SHELL= statement in the CONFIG.SYS file. Normally, this is COMMAND.COM. However, by changing this statement in the CONFIG.SYS file, you can replace the DOS command processor with any program you choose.

When OS/2 Warp is installed, the SET SHELL= statement specifies to start COMMAND.COM with a default environment size of 256 bytes. If you run out of space while defining environment variables using the SET command, you can change the size of the DOS environment by adding the parameter **/e:size** to the end of the SET SHELL= statement in the CONFIG.SYS file, where size is the number of bytes to allocate for the DOS environment.

When a DOS command processor is started, the AUTOEXEC.BAT file is executed automatically before presenting you with a command prompt. You may put any statements you want into the AUTOEXEC.BAT file, including statements to run a virus protection program, set DOS environment variables, load required Terminate and Stay Resident (TSR) DOS programs, set the video display mode, and so on.

For most sound cards, if you start a DOS program that requires use of your sound card, no other program may use the sound card while that program is running. This means that the OS/2 system sounds are disabled and you may not be able to start another DOS or Windows program that requires sound until you end this program. This is a limitation of the sound device drivers provided by the manufacturers of the sound cards.

DOS Full Screen. The DOS Full Screen object opens a full-screen window containing a DOS command prompt. You may invoke any DOS program from this command prompt. This window takes up the entire screen and does not have a title bar or sizing borders.

You can return to the Workplace Shell at any time by pressing **Ctrl-Esc**. The window will be minimized and your Desktop will be visible again, with the Window List displayed.

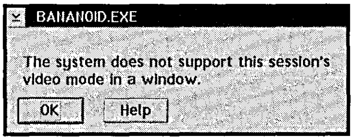
You can return to the full-screen command prompt at any time by double-clicking on its minimized icon or by double-clicking on its title in the Window List.

When you are finished using the full-screen command prompt window, type **exit** at the prompt and the window will close.

DOS Window. The DOS Window object opens a window containing a DOS command prompt. From this command prompt, you may invoke any DOS program that can run in a display mode compatible with your Desktop. For example, with many display adapters, you cannot invoke a DOS program that uses Super VGA graphics (that is, a resolution higher than 640 × 480 pixels) from a windowed command prompt, even though your Desktop is running at a higher resolution. This is a limitation of the display hardware and the display device drivers. However, you can run these programs from a full-screen command prompt.

Shhh! Tip

If you try to run a DOS program from a windowed command prompt that is not supported by the Desktop display mode, you will see the message box shown in Figure 9-7.



BANANOID.EXE

The system does not support this session's video mode in a window.

Figure 9-7.
Video Mode Not Supported Message Box

When you get this message, you must press the **OK** push button to remove the message box from the screen. You then have two choices:

1. You can terminate the program by double-clicking on the system icon at the upper-left corner of the program's window.
2. You can switch the program to a full-screen session by clicking on the system icon at the upper-left corner of the program's window and selecting the **Full-screen** option from the pop-up menu.

When you are finished using the windowed command prompt window, type **exit** at the prompt and the window will be closed.

Tip

You can set the physical characteristics for all OS/2 and DOS windowed sessions by holding the **Shift** key while manipulating the window. You can open all windows maximized or restored, set the window's position, and set the window's size just as you can for an OS/2 Window.

A DOS Window has the same items in its pop-up menu as an OS/2 Window. These items perform the same functions for DOS Windows as for OS/2 Windows. In addition, a DOS Window has the following items in its pop-up menu:

- ✘ **Scaled image** — Toggles between two different window sizes when displaying in graphics mode. When this option is not selected, a graphics mode window may be rather small. If you select this option, the window is drawn larger, and the graphics image is scaled to fit. This item appears only if the program running in the window is using a graphics video mode.
- ✘ **DOS Settings...** — Displays the DOS Settings - Categories dialog to allow you to modify some of the DOS settings for the program. See Chapter 16 for a description of the DOS settings.

From now on, any time you open a DOS Window, it will have the number of rows and columns specified by the MODE command, it will be positioned where you left it, and it will be maximized (so it will automatically adjust its size to match the number of rows and columns being displayed).

Customizing DOS Windows

Tip

You can customize your DOS Windows as follows:

1. Add the statement **mode columns rows** to the AUTOEXEC.BAT file, where columns is the width of the window in characters and rows is the height of the window in lines. The value for columns must be 40 or 80, and the value for rows must be 25, 43, or 50. The standard default is 25 lines of 80 characters.
2. Open a DOS Window.
3. Configure the DOS Window just as you configured an OS/2 Window. Any changes to the window size, position, or maximized versus restored state that you make while holding the **Shift** key apply to all DOS and OS/2 Windows.

DOS from Drive A. The DOS from Drive A: object opens a full-screen DOS window and boots the version of DOS that is contained on the DOS Startup Diskette in the A: drive. Normally, you would configure this diskette to present you with a DOS command prompt, but you could invoke any DOS program from the diskette. See *Starting a Specific Version of DOS* in the OS/2 Command Reference in the Information folder for instructions on how to build a DOS Startup Diskette.

You can return to the Workplace Shell at any time by pressing **Ctrl-Esc.** The window will be minimized, and your Desktop will be visible again, with the Window List displayed.

You can return to the full-screen command prompt by double-clicking on its minimized icon or by double-clicking on its title in the Window List.

When you are finished using the full-screen command prompt window, type **exit_vdm** at the prompt and the window will be closed.

WIN-OS/2 command prompts

There are two types of WIN-OS/2 command prompts: full-screen and windowed. The two types of command prompts share some common characteristics.

Both WIN-OS/2 command prompt objects start WIN-OS/2 and invoke the Windows Program Manager. The windows hardware configuration is described in the SYSTEM.INI file, and the software configuration is in the WIN.INI file.

For most sound cards, if you configure Windows for sound support and start a WIN-OS/2 session, no other program may use the sound card while that session is active. This means that the OS/2 system sounds are disabled and you may not be able to start another DOS program or Windows session that requires sound until you end this session. This is a limitation of the sound device drivers provided by the manufacturers of the sound cards.

Note

Pro Audio sound cards by MediaVision have both Sound Blaster–compatible hardware and Pro Audio hardware on the same card. The MediaVision device drivers allow you to configure one section of the card for use by OS/2 programs and the other section of the card for use by Windows or DOS programs. For example, you could install the Sound Blaster device drivers into Windows while having the Pro Audio sound drivers installed for OS/2. With the device drivers installed this way, an OS/2 program can use the sound card at the same time as a Windows session is using it. However, you still can't share the sound card between multiple Windows sessions or between a Windows session and a DOS program.

WIN-OS/2 Full Screen. The WIN-OS/2 Full Screen object starts the Windows Program Manager in a full-screen window. You may invoke any Windows program from the Program Manager.

To return to the Workplace Shell, press **Ctrl-Esc** at any time. The window will be minimized, and your Desktop will be visible again, with the Window List displayed.

You can return to the full-screen command WIN-OS/2 session by double-clicking on its minimized icon or by double-clicking on its title in the Window List.

When you are finished using the full-screen WIN-OS/2 session, double-click on the Program Manager's system icon (in the upper-left corner of the Program Manager's window). This will close the Program Manager and the full-screen WIN-OS/2 session.

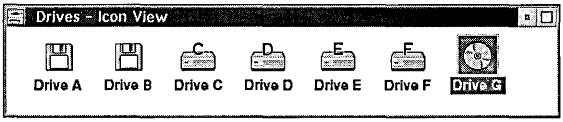
WIN-OS/2 Window. The WIN-OS/2 Window object starts the Windows Program Manager in a window. You may invoke any Windows program from the Program Manager.

When you are finished using the WIN-OS/2 session, double-click on the Program Manager's system icon (in the upper-left corner of the Program Manager's window). This will close the Program Manager and the WIN-OS/2 session.

Drives folder

The Drives folder (see Figure 9-8) contains Disk objects for each file system device connected to your system. This includes diskette drives, fixed disk drives, CD-ROM drives, read-write optical drives, and attached network drives. See Chapter 5 for a description of the Drive object.

Figure 9-8.
Drives Folder



Productivity folder

The Productivity folder (see Figure 9-9) contains objects that perform useful functions for you. These objects are optionally installed if you choose to install tools and games during phase 2 of System Installation or from Selective Install.

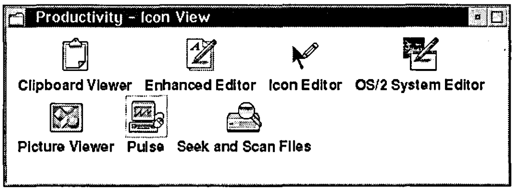
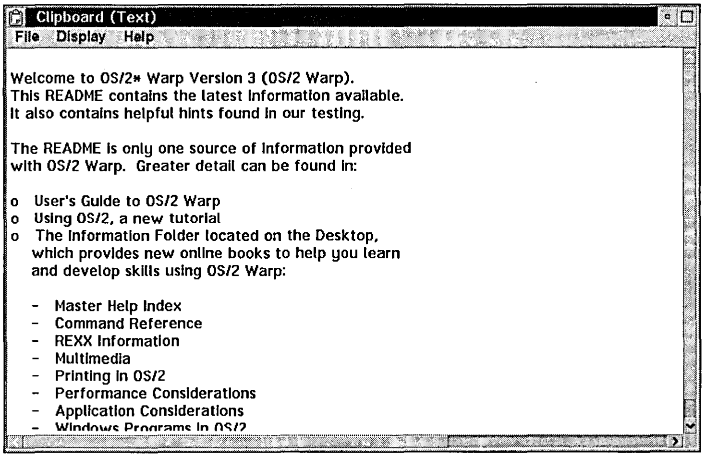


Figure 9-9.
Productivity Folder

Clipboard Viewer

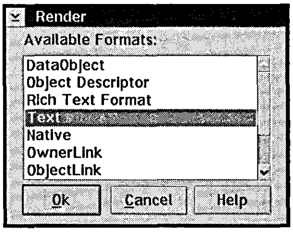
The Clipboard Viewer (Figure 9-10) shows you the current contents of the OS/2 clipboard. You can use the clipboard to exchange information among OS/2, DOS, and Windows programs.



```
Welcome to OS/2* Warp Version 3 (OS/2 Warp).
This README contains the latest information available.
It also contains helpful hints found in our testing.

The README is only one source of information provided
with OS/2 Warp.  Greater detail can be found in:

o  User's Guide to OS/2 Warp
o  Using OS/2, a new tutorial
o  The Information Folder located on the Desktop,
   which provides new online books to help you learn
   and develop skills using OS/2 Warp:

   -  Master Help Index
   -  Command Reference
   -  REXX Information
   -  Multimedia
   -  Printing in OS/2
   -  Performance Considerations
   -  Application Considerations
   -  Windows Programs in OS/2
```

Figure 9-10.
Clipboard Viewer

The clipboard can normally hold one item at a time. However, that item can be represented in different formats, and the clipboard can hold multiple formats of data. You can choose which format of data you want to see by selecting the **Display** menu item and then selecting the **Render...** choice from the menu. This displays the Render dialog, as shown in Figure 9-11.



Available Formats:

DataObject
Object Descriptor
Rich Text Format
Text
Native
OwnerLink
ObjectLink

Figure 9-11.
Clipboard Viewer:
Render Dialog

From the Available Formats list box, select the format of data that you want to see. The selected data format will be displayed in the Clipboard Viewer's window. If the format you selected is not available, the Clipboard Viewer displays a message box indicating "Format not rendered." You will then have to choose another format to see the data.

Enhanced Editor

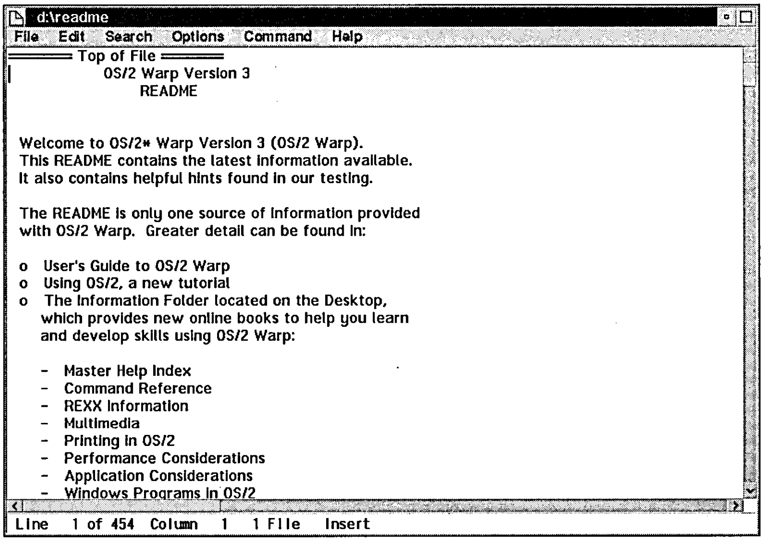
The Enhanced Editor is a full-function text editor, as shown in Figure 9-12. It is capable of editing multiple files simultaneously and can even perform some minimal word processing functions.

The best way to learn how to use the Enhanced Editor is to try it out. You can invoke it from the Productivity folder by double-clicking on its icon, or from a command prompt by typing **EPM** or **EPM filename,** where filename is the name of a file to be edited. You can request many of the useful functions directly from the Enhanced Editor's menus.

The File menu contains the following functions:

- ✗ New — Replaces the file in the current edit window with an empty file.
- ✗ **Open.Untitled** — Opens a new edit window with an empty file.
- ✗ **Open...** — Opens an existing file into a new edit window.

Figure 9-12.
The Enhanced Editor



```
═════ Top of File ═════
          OS/2 Warp Version 3
                README


Welcome to OS/2* Warp Version 3 (OS/2 Warp).
This README contains the latest information available.
It also contains helpful hints found in our testing.

The README is only one source of information provided
with OS/2 Warp.  Greater detail can be found in:

o  User's Guide to OS/2 Warp
o  Using OS/2, a new tutorial
o  The Information Folder located on the Desktop,
   which provides new online books to help you learn
   and develop skills using OS/2 Warp:

   -  Master Help Index
   -  Command Reference
   -  REXX Information
   -  Multimedia
   -  Printing in OS/2
   -  Performance Considerations
   -  Application Considerations
   -  Windows Programs in OS/2
```

```
Line   1 of 454   Column   1   1 File   Insert
```

- ✘ **Import text file...** — Copies an existing text file into the file you are currently editing.
- ✘ **Rename...** — Changes the name of the current file.
- ✘ **Save** — Saves the current file.
- ✘ **Save as...** — Saves the current edit window into a new file.
- ✘ **Save and close** — Saves the current file and closes the file and the edit window.
- ✘ **Quit** — Closes the current edit window without saving the changes to the file.
- ✘ **Print file...** — Prints the current edit window.

The Edit menu performs the following editing operations on the current edit window:

- ✘ **Undo line** — Removes the last changes done to the current line.
- ✘ **Undo...** — Removes the changes done anywhere in the edit window. The Undo/Redo dialog allows you to cycle through the changes you made and either remove the changes or redo them (if you previously removed them).
- ✘ **Copy** — Copies the highlighted text to the clipboard.
- ✘ **Cut** — Copies the highlighted text to the clipboard and removes it from the edit window.
- ✘ **Paste** — Copies the text from the clipboard to the position in the edit window marked by the cursor.
- ✘ **Paste lines** — Copies the text from the clipboard as new lines in the edit window.
- ✘ **Paste block** — Copies the text from the clipboard as a rectangular block in the edit window.
- ✘ **Style...** — Changes the style of marked text or registers a new style. A style consists of a font, the font size, special font attributes (such as bold, italics, underline, strikethrough, or outline), and the foreground and background colors for the text.
- ✘ **Copy mark** — Copies the marked text to the cursor.
- ✘ **Move mark** — Moves the marked text to the cursor.
- ✘ **Overlay mark** — Overlays the marked text on the text following the cursor.

- ✗ **Adjust mark** — Overlays the marked text at the cursor and then blanks the source.
- ✗ **Unmark** — Removes the text marks.
- ✗ **Delete mark** — Deletes the marked text.
- ✗ **Print mark...** — Prints the marked text.

The Search menu enables you to find or replace text in the edit window as follows:

- ✗ **Search...** — Displays the Search dialog, which allows you to search for and replace text anywhere in the edit window.
- ✗ **Find next** — Finds the next occurrence of the text specified by the last Search operation.
- ✗ **Change next** — Changes the next occurrence of the text as specified by the last Search and Replace operation.
- ✗ **Bookmarks** — Allows you to define a bookmark or position the cursor on a selected bookmark.

The Options menu allows you to specify various options that control the operation of the program:

- ✗ **Preferences** — Allows you to customize the editor.
- ✗ **Autosave...** — Displays the current autosave value and the autosave directory.
- ✗ **Messages...** — Displays previously displayed messages.
- ✗ **Frame controls** — Allows you to specify options controlling the location and display of various status areas for the editor.
- ✗ **Save options** — Saves the options you have selected so that they are active the next time you invoke the editor.

The Command menu allows you to execute OS/2 or editor commands:

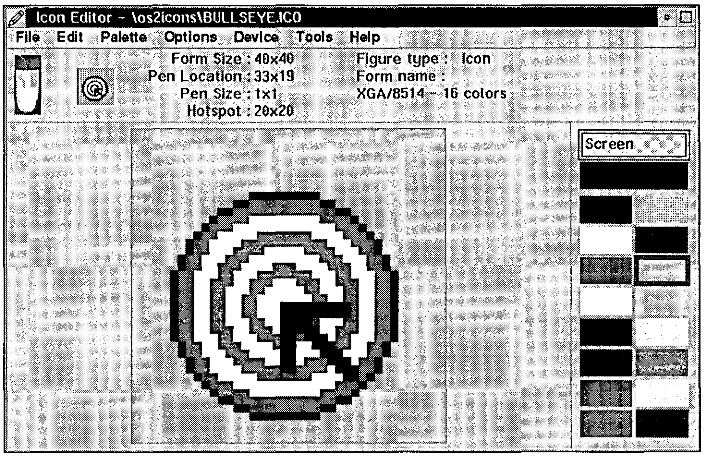
- ✗ **Command dialog...** — Displays a dialog that allows you to specify a command to be executed.
- ✗ **Halt command** — Stops the currently executing command.

You can see from this brief introduction to the menu structure that the Enhanced Editor has a lot of very powerful functions. The Help information can guide you through the rest of its functions.

Icon Editor

The Icon Editor (Figure 9-13) allows you to edit icons, pointers, and small bitmaps. You can invoke the Icon Editor by double-clicking on its icon in the Productivity folder, by invoking **ICONEDIT** from a command prompt, by selecting the **Edit...** push button on the General Page of the Settings Notebook of any object, or by double-clicking on an Icon, Pointer, or Bitmap data file object.

Figure 9-13.
Icon Editor



Form Size : 40x40
Pen Location : 33x19
Pen Size : 1x1
Hotspot : 20x20
Figure type : Icon
Form name :
XGA/8514 - 16 colors

The main portion of the Icon Editor's window is taken up by a representation of the icon, pointer, or bitmap you are editing. Each square within this area represents one pixel of the final image. The area to the right contains the set of colors that you can use.

For an icon or pointer, two of the colors are Screen and Inverse. If you set a pixel to the Screen color, that pixel is transparent. (The background shows through.) If you set a pixel to the Inverse color, that pixel allows the inverse of the background color to show through. This allows an icon or a pointer to have an irregular shape; the area outside the actual image is set to the Screen color.

The Icon Editor is a fairly basic editor. You can color individual pixels and draw horizontal and vertical lines, circles, and ellipses. You can also use cut, copy, and paste to move image information from one icon to another, or from another graphics program to the Icon Editor. For an icon or pointer, you can designate the hot spot and make certain pixels transparent. You cannot enter text directly with the Icon Editor. However, if you are fairly creative and have a steady hand, you can draw anything you want.

If you just invoke the Icon Editor without specifying a file to be edited, you start out with a clean slate and a default color table. You should choose the correct object you want to create before you start painting. (See the description of the File/New option, below.)

If you invoke the Icon Editor and specify the file to be edited, the editing area is filled in with the current image and the color table is set to the color table from the file.

Warning

If your system is Palette managed (see Chapter 6), the colors in the editing window may not be the same as the colors in the color table. For example, on my home system, when the Palette Manager is active, the Screen Color, which displays as a light green in the color table, displays as a medium gray in the image area.

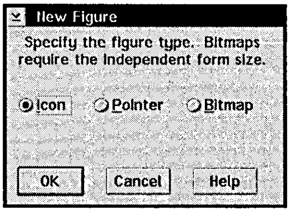
Shhh! Tip

When I am editing an icon with the Palette Manager active, I often can't see the cursor because the screen color displays as the same color as the cursor, medium gray. To get around that problem, I temporarily fill the outside of the icon with white, edit the icon, and then fill the outside with the screen color just before I save the file.

You can select two colors from the color table to be used when drawing. Move the cursor to a color in the color table and click mouse button 1. This color is now assigned to that mouse button. Move the cursor to a different color and click mouse button 2 to assign a different color to it. You can then move the cursor to within the image area and press either mouse button to paint with the selected colors. You can continue to paint pixels by holding the mouse button and moving the mouse pointer around within the image area. If you want to fill an area with either of your selected colors, select the **Tools/Color fill** option and then click either button within the image area to fill the area with the color.

The menus of the Icon Editor allow you to control its subsidiary functions. With the File menu, you can control what file you are editing, as follows:

- ✗ New — Creates a new image file. This displays the New Figure dialog, as shown in Figure 9-14. You must specify the type of image you are creating by selecting the **Icon**, **Pointer**, or **Bitmap** radio push button. If you select a bitmap, you can specify the size of the bitmap image in pixels.



Specify the figure type. Bitmaps require the independent form size.

Figure 9-14. Icon Editor: New Figure Dialog

- ✗ **Open...** — Opens an existing icon, pointer, or bitmap file.
- ✗ **Save** — Saves the edited image in the current file.
- ✗ **Save as...** — Saves the edited image in a new file.
- ✗ **Next** — Opens the next file specified in the list of files. This option is available only if you invoked ICONEDIT from a command prompt and specified more than one filename.

The Edit menu specifies various functions to be performed on the image area:

- ✗ **Undo** — Removes the last changes you made.

Warning

It's not easy to predict how much Undo will remove. Use this option very carefully or you may lose all of your work.

- ✗ **Cut** — Copies the selected area to the clipboard and fills the selected area with the screen color. This option is available only if you previously selected an area.
- ✗ **Copy** — Copies the selected area to the clipboard. This option is available only if you previously selected an area.
- ✗ **Paste** — Copies the image from the clipboard into the image area. The Icon Editor draws a rectangle showing where the pasted image will be placed. You can move this rectangle around by holding mouse button 1 and moving the mouse. When you release mouse button 1, the image is pasted where the rectangle indicates. This option is available only if there is an image on the clipboard.

Tip

If you paste the image to the wrong place, just select **Edit/Undo** to remove it and try again.

- ✗ **Clear** — Fills the selected area with the screen color. This option is available only if you previously selected an area.

If you cleared the wrong area, just select **Edit/Undo** to replace the original contents and try again.

- **Select** — Selects a rectangle in the image area. To specify the selected area, move the cursor to one corner of the area to be selected, press and hold mouse button one, move the cursor to the opposite corner, and release the mouse button.
- **Select all** — Selects the entire image area.
- **Stretch paste** — Copies the image from the clipboard and stretches it to fit the selected area. To use this option, you have to copy an image to the clipboard (either from the Icon Editor or from some other graphics program), select an area to receive the image, then select **Stretch paste.** This option is available only if there is an image on the clipboard and you previously selected an area.

If you paste the image to the wrong area, just select **Edit/Undo** to remove it and try again.

- **Fill** — Fills the selected area with the color currently assigned to mouse button 1. This option is available only if you previously selected an area.

If you fill the image with the wrong color, just select **Edit/Undo** to replace the original contents and try again.

- **Flip horizontal** — Takes the selected area and reverses it horizontally (so that the left side becomes the right side and the right side becomes the left side). This option is available only if you previously selected an area.

If you flip the wrong area, just select **Edit/Undo** to replace the original contents and try again.

- **Flip vertical** — Takes the selected area and reverses it vertically (so that the top becomes the bottom and the bottom becomes the top). This option is available only if you previously selected an area.

If you flip the wrong area, just select **Edit/Undo** to replace the original contents and try again.

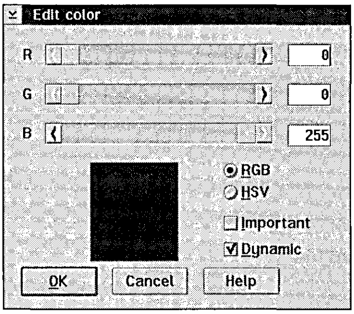
- **Circle** — Draws a circle or ellipse in the selected area. The circle or ellipse will border on the edges of the selected area. The color of the circle is the color assigned to mouse button 1. The current pen size controls the width of the line used to draw the circle.

To draw a filled-in circle, first select a pen size that is at least as big as half the width of the circle. (You may have to use the Option/Set pen shape option to do this.) Then select the area for the circle, select the color for the circle, and select the **Edit/Circle** option. After drawing the circle, don't forget to reset the pen size back to normal.

The Palette menu specifies various functions that can be performed on the set of editing colors:

- **Load default palette** — Resets the color set to its default values.
- **Open...** — Loads a palette from a file you previously saved to disk.
- **Save** — Saves the current color set in a file. This option has meaning only if you previously loaded a palette from a file.
- **Save as...** — Saves the current color set in a new file.
- **Edit color...** — Displays the Edit color dialog (Figure 9-15), which allows you to modify the color assigned to mouse button 1.

Figure 9-15.
Icon Editor: Edit Color Dialog



R 0
G 0
B 255
● RGB
○ HSV
□ Important
☑ Dynamic

- **Swap colors** — Swaps the assignment of the colors of the mouse buttons. If you select the **Preserve figure** option, only the assignment of colors to the mouse buttons changes. Selecting the **Don't preserve figure** option changes the assignment of colors not only to the mouse buttons but to all areas in the image area. For example, let's say that the color red is assigned to mouse button 1 and the color blue is assigned to mouse button 2. If you select **Swap colors** and then select the **Don't preserve figure** option, all red areas in the image area change to blue, all blue areas in the image area change to red, mouse button 1 is assigned the color blue, and mouse button 2 gets assigned the color red.

- ✗ **Set default palette** — Saves the current color set as the predefined palette. (For more information, see the discussion of the **Edit predefined...** option on the Device menu following Figure 9-16.)

The Options menu sets the current display and drawing options:

- ✗ **Test** — Allows you to see what your new icon or pointer looks like by making the mouse pointer into the full-size icon or pointer. You have to select this option again to go back to the normal editing mode.
- ✗ **Grid** — Displays a grid of horizontal and vertical lines in the image area. This makes it easier to see where the individual pixels are located.
- ✗ X **background** — Draws an *x* in any pixel that contains the screen or inverted screen color. This option is available only if you are editing an icon or a pointer.
- ✗ **Draw straight** — Allows you to draw only horizontal or vertical lines.
- ✗ **Pen size** — Allows you to specify the size of the drawing "pen," which can vary from 1 × 1 pixel to 9 × 9 pixels. When you press one of the mouse buttons, an area of the specified size is set to the color assigned to the mouse button. If you select one of the pen sizes from this option, the pen size and shape set by the Set pen shape option is canceled.
- ✗ **Preferences** — Allows you to choose the way the editor works. The options you can choose are
  - **Save prompting** — Prompts you to save your file if you try to leave the Icon Editor without saving your work after your last change.
  - **Suppress warnings** — Stops the Icon Editor from displaying warning messages.
  - **Save state on exit** — Lets the Icon Editor save all of your preferences, options, colors, and so on when you exit the editor.
  - **Display status area** — Displays the status area on the primary window.
  - **Reset options and modes** — Resets all Icon Editor options to their default state.

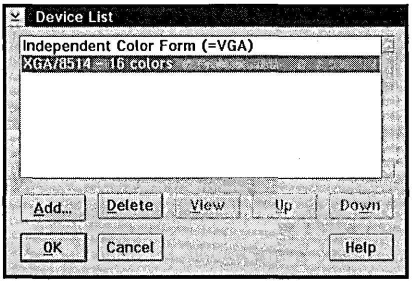
- **Set pen shape** — Sets the shape and size of the drawing pen to the size of the selected area. Selecting this option cancels the pen size you set with the Pen size option. This option is available only if you previously selected an area.
- **Hotspot** — Allows you to specify the hot spot for an icon or pointer. The hot spot is the exact pixel to which the pointer is pointing. For example, the normal mouse cursor is an arrow. The hot spot for this pointer is the tip of the arrow. The default hot spot is the center of the icon or pointer. To select the hot spot, turn on the Hotspot option, move the cursor to the pixel you want to designate as the hot spot, and click mouse button 1.

Warning

You have to turn off the Hotspot option to return to normal editing mode.

The Device menu lets you select the device characteristics for the image you are editing. Icons, pointers, and bitmaps allow you to store different images for different types of devices, all in the same file. For example, an icon may have an Independent Color Form and an XGA/8514 Form. The appropriate image is selected when the icon appears, based on the characteristics of your display. You can specify the following options:

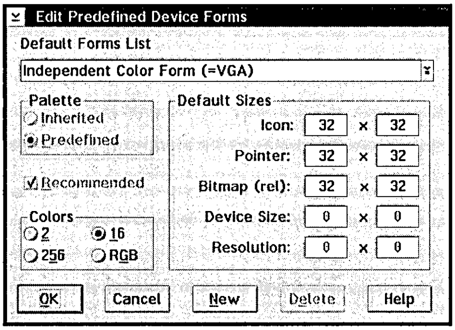
- **List...** — Displays the Device List dialog, as shown in Figure 9-16. This dialog allows you to select forms for editing and to add new forms to the file.


Device List

Independent Color Form (=VGA)
XGA/8514 - 16 colors

Figure 9-16.
Icon Editor: Device List Dialog

- **Edit predefined...** — Displays the Edit Predefined Device Forms dialog, as shown in Figure 9-17. This dialog allows you to modify the characteristics of the forms for each supported device type.



Default Forms List: Independent Color Form (=VGA)

Palette
○ Inherited
● Predefined

☑ Recommended

Colors
○ 2 ● 16
○ 256 ○ RGB

Default Sizes
Icon: 32 x 32
Pointer: 32 x 32
Bitmap (rel): 32 x 32
Device Size: 0 x 0
Resolution: 0 x 0

Figure 9-17. Icon Editor: Edit Predefined Device Forms Dialog

The Tools menu specifies miscellaneous color modes:

- **Color fill** — Allows you to fill an area with a selected color. The cursor changes to a paint bucket. Select the color with which you want to fill an area, move the cursor so that the tip of the dripping paint points to a pixel within the area, and press the mouse button corresponding to your selected fill color. All pixels that are connected to the target pixel and have the same color as that pixel are changed to the fill color.

Warning

Don't forget to turn off Color fill mode when you are done with it, or you may get a surprise the next time you click the mouse in the image area!

- **Find color** — Allows you to find the color associated with a pixel in the image area. The mouse pointer changes to a question mark with an arrow at the bottom. The tip of the arrow is the hot spot. Move the cursor to the pixel that interests you and click one of the mouse buttons. The mouse button that you clicked will be assigned to the color at that pixel.

Tip

Don't forget to turn off Find color mode or you won't be able to draw anything else in the image area.

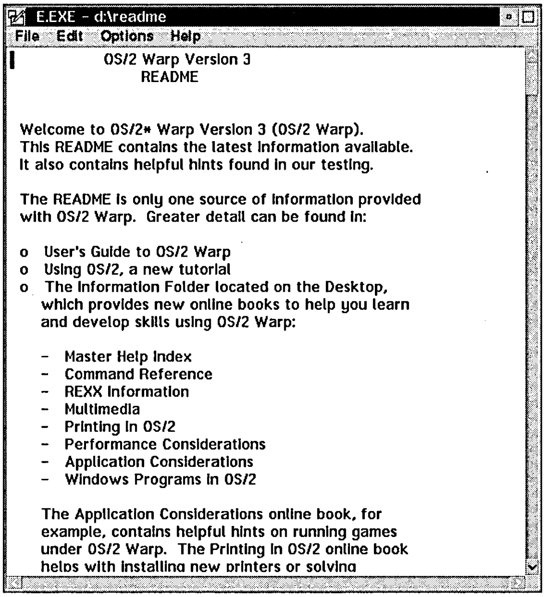
OS/2 System Editor

The OS/2 System Editor (Figure 9-18) is a minimum-function text editor. If you just want to edit a simple text file such as a REXX script, you can use the System Editor. It is capable of editing only one file at a time.

Tip

If you are interested in serious text editing, you should use the Enhanced Editor.

Figure 9-18.
OS/2 System Editor



```
            OS/2 Warp Version 3
                 README


 Welcome to OS/2* Warp Version 3 (OS/2 Warp).
 This README contains the latest information available.
 It also contains helpful hints found in our testing.

 The README is only one source of information provided
 with OS/2 Warp.  Greater detail can be found in:

 o  User's Guide to OS/2 Warp
 o  Using OS/2, a new tutorial
 o  The Information Folder located on the Desktop,
    which provides new online books to help you learn
    and develop skills using OS/2 Warp:

    -  Master Help Index
    -  Command Reference
    -  REXX Information
    -  Multimedia
    -  Printing in OS/2
    -  Performance Considerations
    -  Application Considerations
    -  Windows Programs in OS/2

    The Application Considerations online book, for
    example, contains helpful hints on running games
    under OS/2 Warp.  The Printing in OS/2 online book
    helps with installing new printers or solving
```

The best way to learn how to use the System Editor is to try it out. To invoke it from the Productivity folder, double-click on its icon. From a command prompt, type E or E **filename,** where filename is the name of a file to be edited. You can request most functions directly from the System Editor's menus.

The File menu contains the following functions related to the file you are editing:

- New — Creates a new file and clears the edit window.
- Open... — Opens an existing file and loads it into the edit window.
- Save — Saves the current file.
- Save as... — Saves the edit window into a new file.
- Autosave... — Specifies that the System Editor is to automatically save your file after you make a certain number of changes. This option is available only if you are editing a named file (that is, a file that you opened using the **File/Open** option or that you named using the **File/Save as...** option).

The Edit menu performs editing operations on the edit window:

- ✗ **Undo** — Removes the last change done in the edit window. This menu item changes to tell you what operation you can undo.
- ✗ **Redo** — Reinstates the last change you removed using Undo.
- ✗ **Cut** — Copies the highlighted text to the clipboard and removes it from the edit window.
- ✗ **Copy** — Copies the highlighted text to the clipboard.
- ✗ **Paste** — Copies the text from the clipboard to the position in the edit window marked by the cursor.
- ✗ **Clear** — Clears the highlighted text to blanks.
- ✗ **Find...** — Displays the Find dialog, which allows you to search for and replace text in the edit window.
- ✗ **Select all** — Highlights all text in the edit window.

The Options menu allows you to specify various options that control the operation of the program:

- ✗ **Set font...** — Displays the Font dialog, which allows you to specify the font in which to display all text in the edit window.
- ✗ **Set colors...** — Displays the Set Colors dialog, which allows you to specify the foreground and background colors used to display all text in the edit window.
- ✗ **Word wrap** — Allows you to turn word wrapping on or off. If word wrapping is turned on, carriage returns occur automatically when your typing reaches the right side of the edit window. If word wrapping is turned off, you control where new lines start; the line of text can be as long as you want it to be.

You can see from this introduction to the menu structure that the System Editor has very few functions. You can use the Help information to learn more about the program.

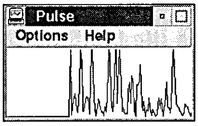
Picture Viewer

The Picture Viewer displays the contents of a Presentation Manager metafile, a standard Picture Interchange Format (PIF) data file, or a Print Spooler file that does not contain printer-specific information. It can also print out the contents of the data file.

Pulse

As shown in Figure 9-19, **Pulse** displays the current processor utilization as a continuous graph. The values plotted on the graph are proportional to the percentage of the processor that is currently being used. For example, if your computer's processor is being used only 25 percent of the time (that is, 75 percent of the time OS/2 Warp is in the "idle" state), the line will be near the bottom of the graph. On the other hand, if your computer's processor is being used 95 percent of the time (that is, OS/2 Warp is idle only 5 percent of the time), the line will be near the top of the graph.

Figure 9-19.
Pulse

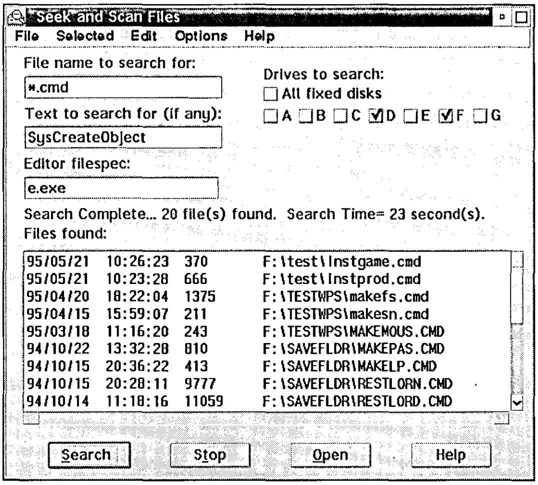


The Options menu allows you to customize the Pulse display. You can select the following options:

- **Background color** — Allows you to select the background color for the graph.
- **Graph color** — Allows you to select the color for the lines of the graph.
- **Smooth** — Allows you to specify that the lines of the graph are to be made into a smoother curve.
- **Centered** — Allows you to specify that the graph is to move back to the center of the window when it reaches the right side. When the Centered option is not selected, the graph scrolls smoothly as it reaches the right side of the window.
- **Freeze screen** — Stops the graph from updating. This allows you to study the graph without its changing. The data is still being captured by Pulse. When you turn this option off, the graph is updated with the saved data.
- **Fill** — Fills in the space under the graph with the graph color. The space above the graph is always filled with the background color.

Seek and Scan Files

You can use the Seek and Scan Files program (see Figure 9-20) to look for a file or a series of files. The filenames may have a particular pattern, or the files themselves may contain data to be searched for.



File name to search for: *.cmd

Text to search for (if any): SysCreateObject

Editor filespec: e.exe

Drives to search:

□ All fixed disks

□ A □ B □ C ☑ D □ E ☑ F □ G

Search Complete... 20 file(s) found. Search Time= 23 second(s).

Files found:

```
95/05/21  10:26:23  370     F:\test\Instgame.cmd
95/05/21  10:23:28  666     F:\test\Instprod.cmd
95/04/20  18:22:04  1375    F:\TESTWPS\makefs.cmd
95/04/15  15:59:07  211     F:\TESTWPS\makesn.cmd
95/03/18  11:16:20  243     F:\TESTWPS\MAKEMOUS.CMD
94/10/22  13:32:28  810     F:\SAVEFLDR\MAKEPAS.CMD
94/10/15  20:36:22  413     F:\SAVEFLDR\MAKELP.CMD
94/10/15  20:28:11  9777    F:\SAVEFLDR\RESTLORN.CMD
94/10/14  11:18:16  11059   F:\SAVEFLDR\RESTLORD.CMD
```

Figure 9-20. Seek and Scan Files

You must fill in the information in the Seek and Scan Files Dialog to specify what files you want to find and where to look for them. This information is in the following fields:

- **File name to search for** — This field specifies the name of the file to be searched for. The name may contain "*" as a wild card character. You can specify one or more drive letters at the beginning of the name. For example, **cd:*.cmd** specifies to search for all files whose names end with ".cmd" on the C: and D: drives. If you do not specify any drive letters, the locations to search are taken from the Drives to search field.

- **Text to search for (if any)** — This specifies the text to look for within each file that matches the filename criterion. If you specify text in this field, only files that match the filename criterion and also contain the specified text are reported in the Files found list.

- **Drives to search** — This specifies which devices to search. If you check the **All fixed disks** check box, all devices attached to your system are searched. Otherwise, you must check the boxes that correspond to the drive or drives that you want to search.

The search may be modified by the Options menu items, as follows:

- ✗ **Search subdirectories** — Specifies to look in all subdirectories. If this option is not selected, only the subdirectory specified in the File name to search for field is searched. If no subdirectory is specified, only the root directories for the specified drives are searched.
- ✗ **Display found text** — Specifies that the lines containing the search text are displayed, instead of the names of the files containing the text.
- ✗ **Ignore case** — Specifies that the search for text within the files is to be case-insensitive. If this option is turned off, the search for text is case-sensitive.
- ✗ **Clear on search** — Specifies that the Files found list box is to be cleared automatically whenever a search is started. If this option is not specified, the newly found files are added to the previous contents of the list box.
- ✗ **Set defaults** — Specifies that the options you set are to be saved and used the next time you run the Seek and Scan Files program.

Once you have filled in the search criteria and specified the correct search options, click on the **Search** push button to start the search.

As the search progresses, the Files found list fills with the names of the files that match the search criteria (or the text lines that match if the Display found text option was selected). If you want to terminate the search operation before all files have been searched (because you already found the files you are looking for), you can click on the **Stop** push button.

When the search operation is complete, there are various things you can do with the list of files returned:

- ✗ You can save the list of filenames by selecting the **File/Save** or **File/Save as...** option. The **File/Save as...** option allows you to specify the name of the file in which to save the list of filenames. The **File/Save** option saves the list in the file you previously saved in.

- ✘ You can edit one or more of the found files. Select the names of the files you want to edit, specify the name of the text editor you want to use in the **Editor filespec** input field, and then either:

    Double-click on the name of the file in the **Files found** list box.

    Click on the **Open** push button.

    Select the **Selected/Open** option from the menu.

  For any of these actions, the specified text editor is invoked, and the names of the selected files are passed to it.

  Note

  For an editor such as the System Editor, the specified editor is invoked once for each selected filename.

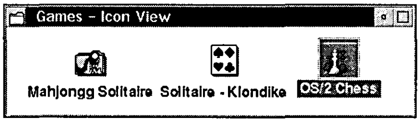
- ✘ You can invoke the found program files. If the files you found are executable programs, you can invoke them by selecting them in the **Files found** list box and then selecting the **Selected/Process** option from the menu.
- ✘ Invoke a command for each of the found files. You can select the **Selected/Command...** option from the menu. The File Command dialog is displayed, allowing you to specify the command to be invoked for each found file. Enter the command you want to execute and then click on the **OK** push button.
- ✘ Copy one or more of the found filenames to the clipboard. Select the names of the files you want to copy to the clipboard and then select the **Edit/Copy** option from the menu.

You can clear the Found files list box at any time by selecting the **Edit/Clear list** option from the menu.

Games folder

The Games folder (Figure 9-21) contains the games that you selected during phase 2 of System Installation or during Selective Install. If you didn't select any games from the Tools and Games category, this folder will be empty.

Figure 9-21.
Games Folder



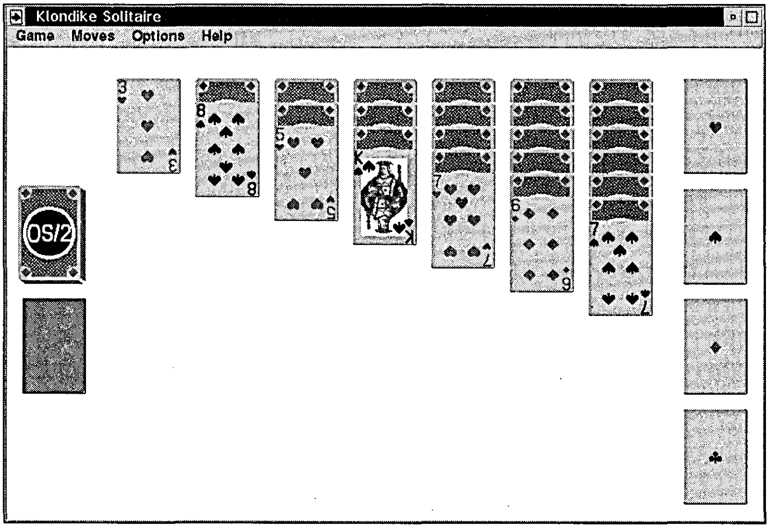
Mahjongg Solitaire

This folder contains the Mahjongg Solitaire game, as well as alternate playing patterns and tile sets. Double-click on the Mahjongg icon to start the game. You can use the Help information to learn how to play the game.

Solitaire - Klondike

This is the traditional solitaire card game, as shown in Figure 9-22. Double-click on its icon to start the game. You can use the Help information to learn how to play the game.

Figure 9-22.
Solitaire — Klondike



OS/2 Chess

This is a one- or two-player chess game (Figure 9-23). One player can play against the computer. Two players can play on the same machine or on different machines connected across a supported network. Double-click on the OS/2 Chess icon to start the game. You can use the Help information to learn how to play the game.

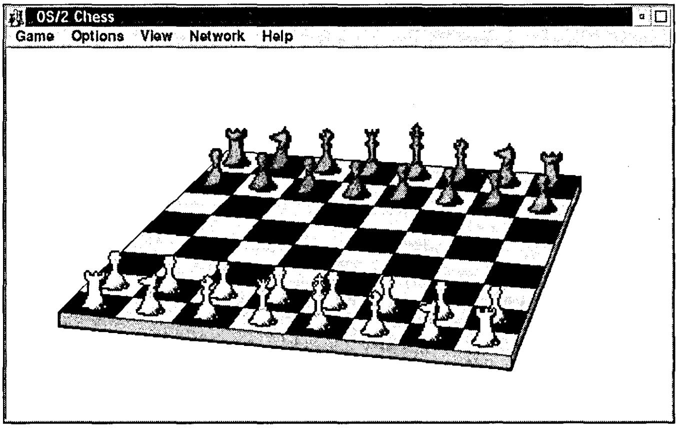


Figure 9-23.
OS/2 Chess

Startup folder

The Startup folder is one of two ways to automatically start programs or open objects when you start your system. The other way is the use of the STARTUP.CMD file.

To open Workplace Shell objects automatically when the system starts, put copies or shadows of the objects you want to start in the Startup folder. When the Workplace Shell starts, it automatically opens all objects in all Startup folders.

Note

If you specify **RESTARTOBJECTS=NONE** in the CONFIG.SYS file, the objects in the Startup folder are not opened automatically.

If you want to start programs automatically when the system starts, independent of the Workplace Shell, you can construct a file named STARTUP.CMD and put it in the root directory of the boot drive. When the system starts, this command file will be invoked. This file may be a simple batch file or a REXX script.

Warning

Do not put in the STARTUP.CMD file any statements that require the Workplace Shell to be running, since STARTUP.CMD is invoked before the Workplace Shell is started.

Tip

If you want to execute a series of commands that require the Workplace Shell to be started, you can construct a command file and put a shadow of the file into the Startup folder. The command file will be invoked after the Workplace Shell is initialized.

It is often difficult to predict the order in which the Workplace Shell will start the objects in the Startup folder. If you must start objects in a particular order, you should construct a REXX script to open the objects that you want and put that script (or a shadow of the script) into the Startup folder.

Shredder

The Shredder deletes any Workplace Shell object that is dragged to the Shredder icon. See Chapter 5 for a description of the Shredder object.

Minimized Window Viewer

The Minimized Window Viewer is a folder containing icons for all objects that have been minimized to the viewer (versus those minimized to the Desktop). See Chapter 5 for a description of the Minimized Window Viewer.

Objects in the Information Folder

The Information folder (see Figure 9-24) contains a wealth of information about OS/2 Warp.

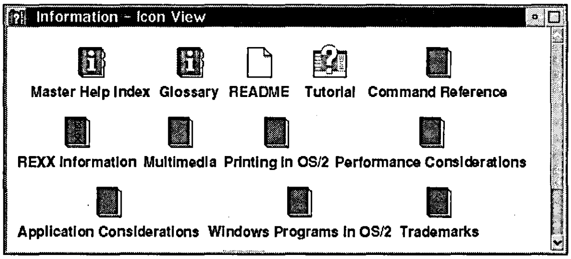
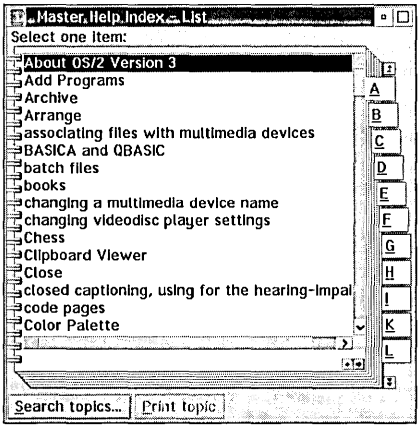


Figure 9-24.
Information Folder

Master Help Index

The Master Help Index (Figure 9-25) contains brief summary information from all OS/2 Warp help files. You can look through the Master Help Index to find this information.

Figure 9-25.
Master Help Index


Master Help Index - List

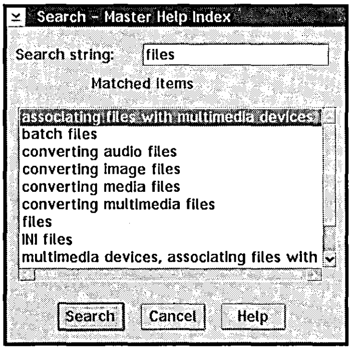
Select one item:

About OS/2 Version 3
Add Programs
Archive
Arrange
associating files with multimedia devices
BASICA and QBASIC
batch files
books
changing a multimedia device name
changing videodisc player settings
Chess
Clipboard Viewer
Close
closed captioning, using for the hearing-impai
code pages
Color Palette

Click on a letter tab on the right side to position the index to topics that start with the specified letter. If the letter you want is not visible, click on the double arrows above or below the index tabs to scroll through the letter tabs. You can then use the scroll bar to the left of the index tabs to scroll through the topics. When the topic you want is visible, double-click on the topic to display it.

You can also search for a topic by clicking on the **Search topics...** push button (or by selecting **Search topics...** from the pop-up menu). This displays the Search dialog (Figure 9-26). In the Search string input field, specify the text you want to search for; then click on the **Search** push button. The list of topics containing the text you requested appears in the Matched items list box. To view any topic in this list, double-click on it.

Figure 9-26.
Master Help Index: Search Dialog


Search - Master Help Index

Search string: files

Matched items

associating files with multimedia devices
batch files
converting audio files
converting image files
converting media files
converting multimedia files
files
INI files
multimedia devices, associating files with

You can print any topic by selecting the name of the topic and then clicking on the **Print topic** push button.

Glossary

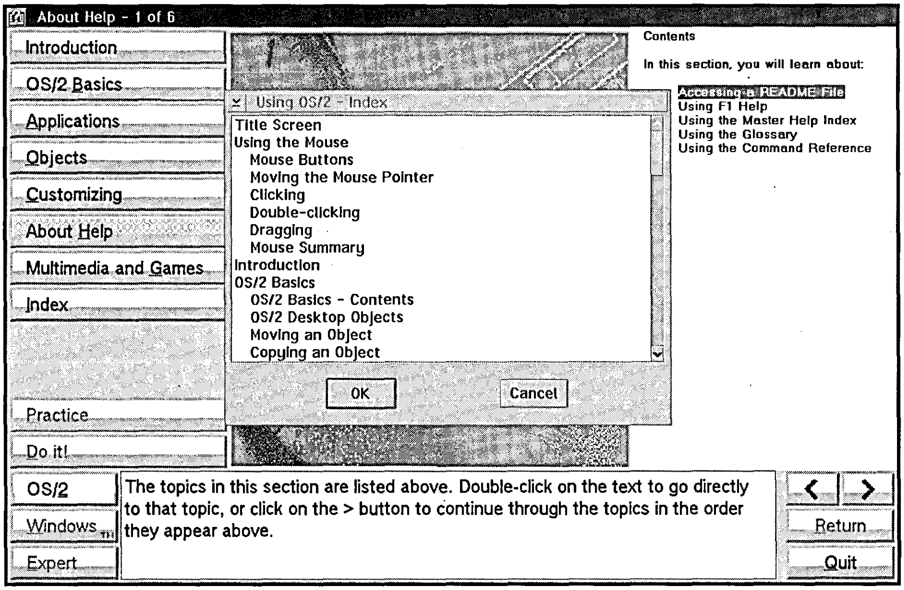
The Glossary looks exactly the same as the Master Help Index, except that the contents are a glossary of terms used within OS/2 Warp.

Tutorial

The Tutorial (Figure 9-27) is an introduction to OS/2 Warp. To start the Tutorial, just double-click on its icon and follow the instructions.

README

The README file gives you the latest information about OS/2 Warp. To read this file, double-click on its icon. This invokes the System Editor so that you can scroll through the file.



Title Screen
Using the Mouse
  Mouse Buttons
  Moving the Mouse Pointer
  Clicking
  Double-clicking
  Dragging
  Mouse Summary
Introduction
OS/2 Basics
  OS/2 Basics - Contents
  OS/2 Desktop Objects
  Moving an Object
  Copying an Object

Contents

In this section, you will learn about:

Accessing a README File
Using F1 Help
Using the Master Help Index
Using the Glossary
Using the Command Reference

The topics in this section are listed above. Double-click on the text to go directly to that topic, or click on the > button to continue through the topics in the order they appear above.

Figure 9-27.
Tutorial

Books

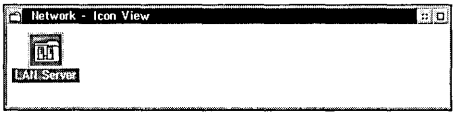
The eight books in the Information folder give you information about various aspects of OS/2 Warp. These books are

Book Title	Contents
Application Considerations	How to run your existing application programs under OS/2 Warp
Command Reference	A description of all the commands you can execute at an OS/2 Command Prompt, in a batch file, and in the CONFIG.SYS file
Multimedia	Information about sound and video support in OS/2 Warp and how to install the Multimedia support
Performance Considerations	Information on how to tune OS/2 Warp for the best performance
Printing in OS/2	Installing and using printers in OS/2 Warp
REXX Information	How to write REXX scripts
Trademarks	A list of IBM and non-IBM trademarks used in the books in the Information folder
Windows Programs in OS/2	Installing and using Windows programs under WIN-OS/2

Objects in the Network Folder

The Network folder (Figure 9-28) is always present in your Desktop folder, but it is hidden until you install software that allows you to communicate with a local area network (LAN).

Figure 9-28.
The Network Folder



The Network folder contains Network Group folders that represent connections to local area networks.

I have IBM LAN Requester 3.0 installed on the system that I am showing in this section. OS/2 Warp support for local area networks is the same no matter what LAN software you have installed. However, the LAN software may install different objects in the OS/2 Warp folders, so what you see may not be exactly the same as what I have pictured here.

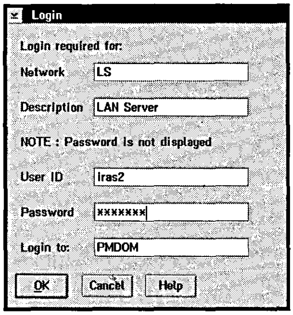
The Network folder shown in Figure 9-28 contains one Network Group folder for LAN Server. This represents the connection to a local area network through IBM LAN Requester 3.0.

The Network folder resides in your Desktop folder. If you create any objects in this folder, they reside on your local system, not anywhere in the network. For example, if you copy a data file to the Network folder, you will find the actual data file in the \DESKTOP\NETWORK directory on your boot drive.

The Network Group folder object

You must be logged on to the local area network before you can open the Network Group folder. If you did not previously log on to the network, the Network Group folder displays the Login dialog (Figure 9-29) when you try to open the folder. You can also get the Login dialog by selecting the **Login...** item from the Network Group folder's pop-up menu.

Figure 9-29.
Login Dialog



Login

Login required for:

Network: LS

Description: LAN Server

NOTE : Password is not displayed

User ID: lras2

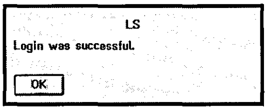
Password: xxxxxxx

Login to: PMDOM

The Network and Description fields tell you what network you are being requested to log on to. You must fill in the User ID, Password, and Login to fields. For IBM LAN Requester, the value in the Login to field is automatically retrieved from the IBMLAN.INI file that was created when you installed the IBM LAN Requester software.

When you have supplied all of the required information, click on the OK push button to log on to the network. If the logon is successful, you will see a message box similar to the one shown in Figure 9-30. When you click on the OK push button, the Network Group folder opens.

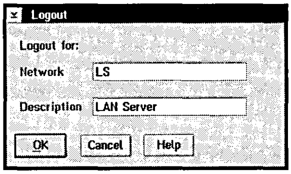
Figure 9-30.
Login Successful Message Box


LS

Login was successful.

Once you are successfully logged on to a network, the **Login...** pop-up menu item is replaced with a **Logout...** menu item. If you select this item, the Logout dialog (Figure 9-31) is displayed to allow you to log off from the network.

Figure 9-31.
Logout Dialog


Logout

Logout for:

Network LS

Description LAN Server

Warning

If you log off from a network, you won't be able to access any shared resources on that network until you log on again.

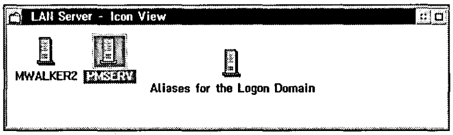
The Network and Description fields tell you what network you are requesting to log off from. Click on the OK push button to log off from the specified network.

Figure 9-32 shows an example of an open Network Group folder. It contains Server folders for each server that is defined in the LAN domain that you logged on to.

Note

There is also a folder labeled Aliases for the Logon Domain. This folder contains the same contents as the Server folder that represents the Domain Controller for the domain that you logged on to.

Figure 9-32.
Network Group Folder



The Network Group folder represents a directory that resides in the \DESKTOP/NETWORK directory. If you create any objects in a Network Group folder, those objects will exist only on your local system, not on the network.

The Server folder object

For each server that you have access to, there is one Server folder (Figure 9-33) in the Network Group folder.

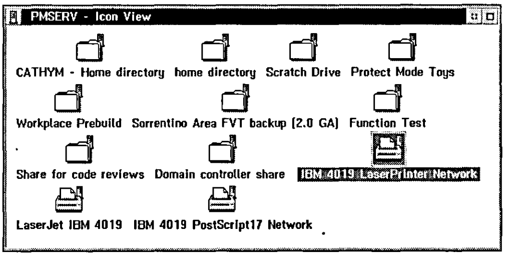


Figure 9-33.
Server Folder

The Server folder contains Shared Directory folders and Remote Printer objects for every shared directory and printer defined by that server.

The Server folder represents a directory that resides in the \DESKTOP/NETWORK directory. If you create any objects in a Server folder, those objects will exist only on your local system, not on the network.

The pop-up menu for the Server folder contains two LAN-related items:

✗ **Logout...** — Displays the Logout dialog (shown in Figure 9-31) to let you log off from the network.

✗ **Access another...** — Displays the Access another network directory dialog (Figure 9-34) to allow you to access another shared resource.

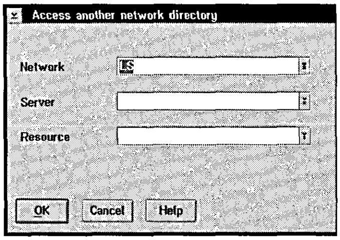


Figure 9-34.
Access Another Network Directory Dialog

The Access another network directory dialog enables you to connect to any shared resource that you have the right to access, even if it isn't in a Server folder contained in the Network folder. For example, you may have the right to access shared resources on many network domains, even though the Network Group folder contains only some of those domains.

You must fill in the required information:

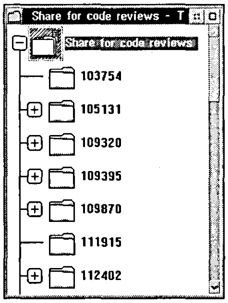
- **Network** — The network that the shared resource is on
- **Server** — The name of the server that shares the resource
- **Resource** — The name of the shared resource

When you have filled in all of the required information, click on the OK push button to connect to the resource.

The Shared Directory folder object

The Shared Directory folder (Figure 9-35) looks like a regular folder. It represents a directory that resides on a server. You can open a Shared Directory folder in icon view, tree view, or details view, just like a regular folder.

Figure 9-35.
Shared Directory Folder



The Shared Directory folder contains folders and data files. Because the shared directory resides on a remote system, it can't contain any abstract objects (since abstract objects must be represented in the OS2.INI file of the local system).

The characteristics of the objects residing in a Shared Directory folder vary with the file system on the server. For example, if the server is an OS/2 system, the file system may be either the FAT file system or the HPFS file system, and the data files and directories can have extended attributes. However, if the server is a Windows NT system, the file system doesn't support extended attributes.

You may have read-only access or read-write access to the objects in the Shared Directory folder, depending on how the directory was shared by the server. Also, a Shared Directory folder may represent a directory on a CD-ROM, which is read-only.

Warning

The Workplace Shell doesn't know that a Shared Directory folder represents a read-only directory until you take some action that requires the system to write to the shared object. The system can't warn you in advance that you have requested an illegal operation, such as copying a file to a read-only directory.

The Settings Notebook for a Shared Directory folder contains a Network Status page, as shown in Figure 9-36. This page contains the following information about the Shared Directory folder:

- **Network** — Specifies the type of network that this folder resides on. The value LS indicates an IBM LAN Server network.
- **Server** — Specifies the name of the server.
- **Resource** — Shows the name of the shared resource. For an IBM LAN Server, this is the alias name that was specified when the directory was shared.
- **Description** — Contains the descriptive name of the shared resource.
- **Status** — Specifies the status of the shared directory, as follows
    - **Available** — You can access the directory.
    - **Login required** — You must type a valid user ID and password before gaining access to this directory. To log in, select the **Login** item from the shared directory's pop-up menu.
    - **Not available** — You can't access the directory at this time.

  Warning

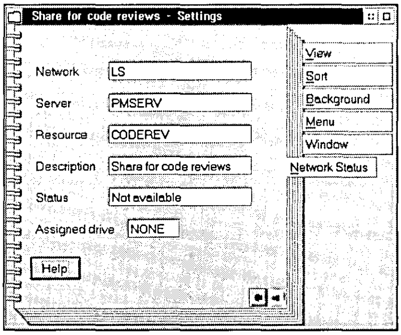
  The Status field doesn't always contain the correct status for the resource. For example, the Status field for the Shared Directory shown in Figure 9-36 indicates "Not available," even though the directory was accessible.

- **Assigned drive** — Specifies the drive letter that is currently assigned to the shared directory. If this field contains NONE, then there is no drive letter assigned.

  Note

  You can access folders and data files on a shared directory through the Workplace Shell without assigning a drive letter to the shared directory—or through any program that supports Universal Naming Convention (UNC) filenames. The only reason to assign a drive letter is to work with the directory from a command prompt or from a program that doesn't support UNC filenames (that is, a program that requires a drive letter to specify the location of a file).

Figure 9-36.
Shared Directory Folder: Network Status Page

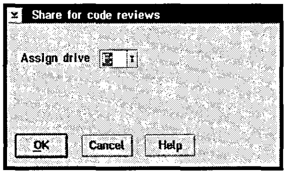

Share for code reviews - Settings

Network: LS
Server: PMSERV
Resource: CODEREV
Description: Share for code reviews
Status: Not available
Assigned drive: NONE

The pop-up menu for a Shared Directory folder contains three LAN-related items:

- **Access another...** — Displays the Access another network directory dialog to allow you to connect to another shared directory, as shown in Figure 9-34.
- **Logout...** — Displays the Logout dialog (see Figure 9-31) to allow you to disconnect from a network.
- **Assign drive...** — Displays the Assign Drive dialog (see Figure 9-37) to allow you to assign a drive letter to the shared directory. Once you assign a drive letter, that letter will be used every time you log on to the network.

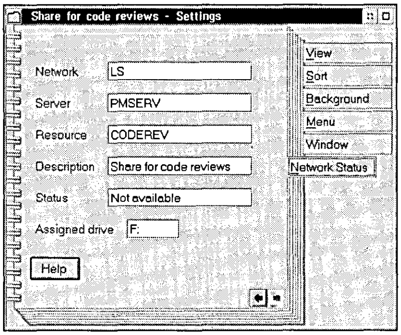
Figure 9-37.
Assign Drive Dialog


Share for code reviews

Assign drive

The Assign Drive dialog allows you to assign a drive letter to a shared resource. Select the drive letter in the **Assign drive** spin button and then click on the **OK** push button to assign the drive letter to the shared resource.

After you assign a drive letter to a resource, the drive letter is displayed on the Network Status page for the Shared Directory folder, as shown in Figure 9-38.



Network: LS

Server: PMSERV

Resource: CODEREV

Description: Share for code reviews

Status: Not available

Assigned drive: F:

Figure 9-38. Shared Directory Folder: Network Status Page (Drive Letter Assigned)

The Remote Printer object

The Remote Printer object represents a printer that is shared by a server. You can print to a remote printer the same as you can print to a local printer. See Chapter 14 for information on using the Remote Printer object.

The Network Shadow object

The Network Shadow object doesn't reside in the Network folder. It is a shadow of a shared resource that you create somewhere on your Desktop.

If you use a shared directory or data file often, you might create a shadow of that object on your Desktop (or in some other folder). This alleviates the tedium of opening the Network folder, Network Group folder, Server folder, and Shared Directory folder before you can finally see the directory or data file.

If you put a shadow of a shared resource on your Desktop (or in any folder that you open), the system has to connect to the network before it can "awaken" the object. If the SET AUTOSTART= statement in the CONFIG.SYS file contains the keyword CONNECTIONS, the Workplace Shell automatically connects to the network when your system starts. You will be presented with the Login Dialog to log on to any network that you were connected to when you previously shut down the system.

The Nowhere folder

The Nowhere folder resides in the root directory of your boot drive. This folder is always hidden.

All Workplace Shell objects (except for the Root Directory folder object) must reside in a folder. Several objects used by the Workplace Shell don't belong to any real folder. Therefore, they are created in the Nowhere folder.

Uncensored

The Nowhere folder was originally named the "no where" folder, meaning that the folder, and the objects created within it, are "no where to be found." However, lately, people have been calling it the "now here" folder, since they are finding that the objects created within it are "now here."

The objects created within the Nowhere folder include

- ✗ All shadow objects on the LaunchPad. These objects must reside somewhere. The LaunchPad is not a folder; it is an abstract object. The only logical home for these shadow objects is the Nowhere folder, since you should never see these objects except on the LaunchPad itself.
- ✗ Print job objects in an open view of a printer. These are transient objects. Since the printer isn't a folder, these objects are created in the Nowhere folder.
- ✗ Printer Driver, Printer Port, and Print Queue Driver objects – These are transient objects that are created only when you open the Settings Notebook for a printer. Again, they have no visible home, so they are created in the Nowhere folder.

Nowhere Folder Caution

Warning

Even though you never see the Nowhere folder, and the \NOWHERE directory always looks empty, do not delete this directory. The Workplace Shell needs this directory for the proper operation of certain functions, such as the LaunchPad.

Summary

There you have it. The objects in the OS/2 System folder provide various functions to help you get the most out of the Workplace Shell. The Information folder is your source of information about OS/2 Warp. The Network folder contains all shared resources.

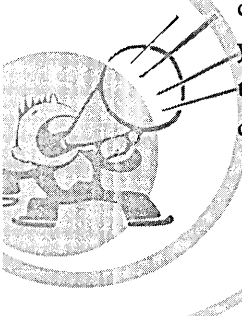
You can move these objects around or you can leave them in the standard folders. You might want to move these objects to consolidate folders or to make some objects more easily accessible. It is your system; make it look the way you want.

Ira H. Schneider is an Advisory Programmer on the Workplace Shell Development Team. He has been instrumental in developing OS/2 Warp, including performance improvements and functional enhancements to the Workplace Shell. He has also worked on the OS/2 Swapper and on OS/2 System Initialization, and he was Lead Architect for OS/2 1.2. Ira has been with IBM since 1969.

Chapter 10

CONFIG.SYS Options Affecting the Workplace Shell

Secret Incantations

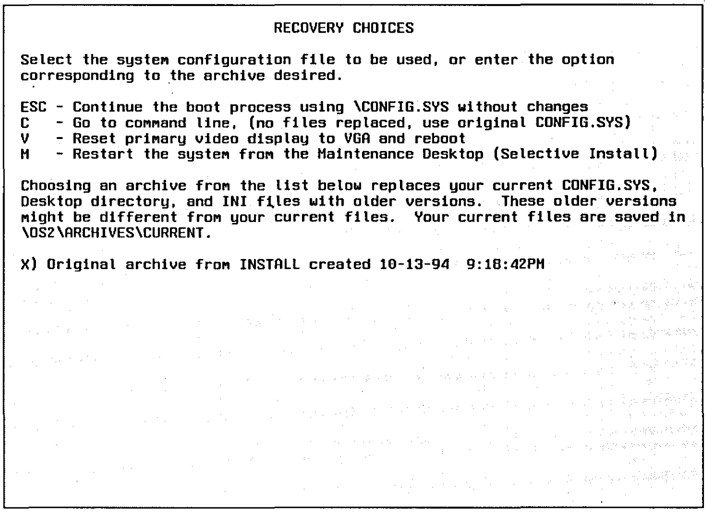


The CONFIG.SYS file contains a lot of different information to configure your system. If you have ever looked at this file, you know how complicated it is.

Many statements in the CONFIG.SYS file set up the basic hardware and software configuration of the system. These statements are built by System Installation, and you will probably never need to change them. I won't even attempt to describe these statements in this section. The statements I *will* explain have to do with configuring the user interface — the Workplace Shell.

You can edit the CONFIG.SYS file with your favorite text editor. If you make a change to the CONFIG.SYS file and the system won't start anymore, don't panic! Restart the system, press **Alt-F1** when the word OS/2 appears at the upper-left corner of the screen, then type C when prompted for the recovery options (see Figure 10-1). This will boot the system with the original CONFIG.SYS file and start a command prompt. You can then use the command **TEDIT CONFIG.SYS** to edit the CONFIG.SYS file and fix the problem. After you save the corrected CONFIG.SYS file, restart the system by pressing **Ctrl-Alt-Del.**

Figure 10-1.
Recovery Choices Screen



```
                          RECOVERY CHOICES

Select the system configuration file to be used, or enter the option
corresponding to the archive desired.

ESC - Continue the boot process using \CONFIG.SYS without changes
C   - Go to command line, (no files replaced, use original CONFIG.SYS)
V   - Reset primary video display to VGA and reboot
M   - Restart the system from the Maintenance Desktop (Selective Install)

Choosing an archive from the list below replaces your current CONFIG.SYS,
Desktop directory, and INI files with older versions.  These older versions
might be different from your current files.  Your current files are saved in
\OS2\ARCHIVES\CURRENT.

X) Original archive from INSTALL created 10-13-94  9:18:42PM
```

You can start OS/2 Warp with different CONFIG.SYS files at different times by using the facilities provided by the Archive and Recover feature. See Chapter 11 for information about how to specify which CONFIG.SYS file to use when you boot your system.

Order of Processing of CONFIG.SYS Statements

Even though I am not going to explain most of the statements in the CONFIG.SYS file, it is important for you to understand the order in which these statements are processed.

The CONFIG.SYS file is processed by a part of OS/2 Warp known as System Initialization. This component is responsible for initializing the base OS/2 operating system components. Its last official duty is to start the Protected Mode Shell (see the section on Starting the user interface later in this chapter).

The statements in the CONFIG.SYS file are not processed sequentially, as you might think. System Initialization parses the CONFIG.SYS file early in its processing and divides the statements into groups related to specific OS/2 Warp functions. When it is ready to initialize the function related to a group, System Initialization processes each CONFIG.SYS statement for that group in sequential order.

The statements in the CONFIG.SYS file are parsed sequentially. If you specify a simple statement multiple times, the last value you specify is retained. For example, if you specify **BUFFERS=60** then later specify **BUFFERS=30**, the number of buffers allocated is 30. If you specify multiple conflicting keywords on a statement, the last keyword is remembered. For example, MEMMAN=NOSWAP,NOMOVE,SWAP has the same effect as MEMMAN=SWAP,NOMOVE.

The IOPL= and the PRINTMONBUFSIZE statements are parsed, but they are ignored in OS/2 Warp.

Here is a listing of the groups of CONFIG.SYS statements and the order in which they are processed:

Statement	*When Processed*
BASEDEV=	Loading base device drivers
LIBPATH= **SET=**	Initializing the global environment
CODEPAGE= **COUNTRY=** **DEVINFO=**	Initializing code page support
DEVICE= **IFS=**	Loading device drivers and file systems
SWAPPATH=	Initializing the Swapper

continued

Statement	*When Processed*
BREAK= **DEVICE=** (DOS device driver) **DEVICEHIGH=** (DOS device driver) **FCBS=** **FILES=** **LASTDRIVE=** **PROTECTONLY=** **RMSIZE=** **SHELL=**	Initializing the VDM support
CALL= **RUN=**	Initializing daemon processes (background server processes)
PROTSHELL=	Initializing the Protected Mode Shell

The statements within each group are processed in the order in which they were specified in the CONFIG.SYS file. For example, all CALL= and RUN= statements are processed as a single group. The CALL= statements are not processed before all RUN= statements. However, all CALL= statements and RUN= statements are processed before the PROTSHELL= statement is processed, no matter where they appear in the CONFIG.SYS file.

The statements that are not listed in this table are processed when their related OS/2 Warp components are initialized. The order in which these statements are specified is not important, since they are all parsed sequentially, but they do not interact with each other.

Workplace Shell Related Statements

This set of statements is used to configure the Workplace Shell. In most cases, the statements written into the CONFIG.SYS file by System Installation are correct, and you don't have to change them. You might want to make small changes to customize the Workplace Shell to work the way you want it to.

Starting the user interface

The user interface in OS/2 Warp normally consists of two programs: the Protected Mode Shell and the Workplace Process.

The Protected Mode Shell is the first program started after System Initialization is complete. It may be either a Presentation Manager (PM) application program or a non-PM program. The characteristics of the system are vastly different for these two different types of shells.

- ✗ PM shell program: The shell program is responsible for initializing the Presentation Manager environment. When the program creates the first PM message queue, Presentation Manager automatically starts the Session Manager (which is the support for multiple sessions), starts the Window List (which allows you to control any running program), and starts the Workplace Process. In this environment, you may have multiple sessions, PM and non-PM programs, DOS and protect-mode programs, and a graphical user interface.
- ✗ Non-PM shell program: In this environment, you may have only a single session, run protect-mode programs only, and have a text-only environment. The shell can invoke other protect-mode programs, but OS/2 Warp will not provide any mechanism for switching between them — that is the responsibility of the shell program.

PROTSHELL= statement

The PROTSHELL= statement tells the system what program to start as the Protected Mode Shell. If you are using the Workplace Shell, this is \OS2\PMSHELL.EXE on the boot drive.

You can specify **PROTSHELL=D:\OS2\CMD.EXE** if you want a simple, single command line interface. This statement may be useful if you want to start the system at a command prompt, make some changes to the system (for example, change a dynamic link library containing a Workplace Shell object class or change the OS2.INI file), and then start the normal Protected Mode Shell by invoking **PMSHELL.EXE** from the command prompt.

You may also change the PROTSHELL= statement if you want to use an alternate user interface. For example, you can use a text-mode shell to run your system. A text-mode shell may be useful in a memory-constrained system in which you want to use only text-mode programs.

Warning

The Protected Mode Shell program must never exit. Once the program exits, there is no user interface, and the system will hang. If the Protected Mode Shell is a Presentation Manager program, it must initialize the PM environment correctly. However, since the initialization process requires the use of undocumented OS/2 Warp interfaces, it is probably not feasible to write your own PM shell program.

SET RUNWORKPLACE= statement

The SET RUNWORKPLACE= statement tells Presentation Manager what program to start as the Workplace Process. If you are using the Workplace Shell, the command is also D:\OS2\PMSHELL.EXE.

If you remove the SET RUNWORKPLACE= statement from the CONFIG.SYS file, the Protected Mode Shell program will start but there will be no Workplace Process. This may be useful if the Protected Mode Shell program is also your user interface.

There are several uses for changing the RUNWORKPLACE= statement. Here are some examples:

- ✘ For a turnkey system, you may have a single PM application program that you want to run. For example, if OS/2 Warp were running as the control program for a cash register terminal in a retail store, you could use the SET RUNWORKPLACE= statement to invoke the PM program that runs the cash register. You could run OS/2 Warp in a very small memory computer without the Workplace Shell running.
- ✘ You can debug a Workplace Shell class you are developing by running the Workplace Shell under the control of a debugger. For example, you can specify **SET RUNWORKPLACE=IPMD PMSHELL.EXE** in the CONFIG.SYS file. IPMD (the debugger that is part of the IBM CSET++ product) will initialize and prompt you for input. You can set an IPMD "load occurrence" breakpoint for your dynamic link library (DLL) and let IPMD invoke PMSHELL.EXE. When your DLL is loaded, IPMD will prompt you for input. At that time, you can put breakpoints into your code and let the system continue running.

If you don't change the PROTSHELL= statement, PMSHELL.EXE will initialize the normal Presentation Manager functions for you:

- PM is initialized.
- Ctrl-Esc displays the Window List.
- Alt-Esc switches between sessions.
- STARTUP.CMD is invoked.
- REXX works.

However, the Workplace Process is responsible for everything else. If you change the global RUNWORKPLACE environment variable, the program you specify is responsible for the rest of the system:

- Starting and managing the Print Spooler
- Installing PM Printer Drivers
- Listing and starting programs
- Providing a safe shutdown mechanism
- Providing a lockup mechanism

Workplace Shell Configuration Files

The Workplace Shell and its application programs have access to two files containing configuration information. The System Configuration file is supposed to contain configuration information common to the system. This information is independent of which user is currently using the system; it refers, instead, to the actual hardware and software configuration of the system. The User Configuration file is supposed to contain configuration information specific to the user. The theory is that you can start the system with the common System Configuration file and a User Configuration file customized for a particular user. In this way, you can get multiple-user support under OS/2 Warp.

Note

That is the theory. The reality is that the information is not split cleanly between the two configuration files, and it is difficult to generate a User Configuration file tailored to a particular user.

There are some add-on products available today that use this feature. They start their own program instead of PMSHELL.EXE. This program presents the user with a Logon panel, asking the user to identify himself. When the user has been identified, the program selects the appropriate User Configuration file and starts the Workplace Shell with the common System Configuration file and the selected User Configuration file.

Warning

I would not recommend your trying this at home, unless you are a trained professional. It is not easy to correctly generate the User Configuration file and a matching desktop structure.

SET SYSTEM_INI= statement

The SET SYSTEM_INI= statement tells PMSHELL.EXE where to find the System Configuration file. The default value is \OS2\OS2SYS.INI on the boot drive. You would normally not change this statement.

SET USER_INI= statement

The SET USER_INI= statement tells PMSHELL.EXE where to find the User Configuration file. The default value is \OS2\OS2.INI on the boot drive. You would normally not change this statement unless you want to support different users or different customization values at different boot times.

Starting the Workplace Shell

The Workplace Shell looks at various global environment variables when it is starting. The variables tell it how to configure itself. You might change some of these values to customize the look of the Workplace Shell.

SET DESKTOP= statement

The SET DESKTOP= statement tells the Workplace Shell where to find the Desktop folder. Normally, the Workplace Shell finds this value in the User Configuration file. However, if the file has been corrupted or if you simply want to start the system with a special Desktop directory structure, you can specify the fully qualified path of the Desktop directory on the SET DESKTOP= statement.

For example, you might specify **SET DESKTOP=D:\MYDESKTOP** to tell the Workplace Shell to open D:\MYDESKTOP as the Desktop folder.

Warning

Be sure that the directory you specify in the SET DESKTOP= statement contains the name of a directory known to the Workplace Shell as a WPDesktop class folder (see Chapter 5 for a description of the WPDesktop object class).

SET AUTOSTART= statement

The SET AUTOSTART= statement specifies a list of Workplace Shell functions that are started automatically when the Workplace Shell is started.

Warning

If you remove one of the keywords from the SET AUTOSTART= statement, that function of the Workplace Shell won't start. You may cripple the user interface by removing the wrong keyword.

The keywords that can be specified on the SET AUTOSTART= statement are

- ✗ PROGRAMS — This allows the Workplace Shell to restart programs that were running when you shut the system down. If you remove this keyword, the Workplace Shell won't start any programs automatically, unless they are specified within the Startup folder.

  Tip

  One of the features supported by the Mouse object is the Comet Cursor (see Chapter 7). Because the Comet Cursor feature is actually implemented as a separate program, the Comet Cursor will not start automatically if the PROGRAMS and FOLDERS keywords don't appear on the SET AUTOSTART= statement. You can bypass this problem by specifying the statement **START COMETRUN.EXE** in your STARTUP.CMD file.

- ✗ TASKLIST — This allows the Workplace Shell to start the Window List. If this keyword is not specified, you will not be able to press **Ctrl-Esc** to get the list of open windows. You would remove this keyword only if you replace the Window List feature of the Workplace Shell with some other program, or if you have some other way to switch between open windows.

  Uncensored

  This sounds good, doesn't it? This is the way the TASKLIST keyword was intended to work, but the truth is that removing the TASKLIST keyword from the AUTOSTART environment variable doesn't stop the Window List from initializing, so you are stuck with it!

- ✗ FOLDERS — This allows the Workplace Shell to open folders that were open when the system was shut down. If you don't specify this keyword, the Workplace Shell will not open any folders when it is starting.

The Desktop is a folder. If you remove the FOLDERS keyword, the Workplace Shell will not be able to open the Desktop either! Be very careful about removing this option.

- CONNECTIONS — This allows the Workplace Shell to reconnect to all servers you were connected to when the system was shut down.

If this option is specified and you had an active network connection when you shut down the system, the Workplace Shell will attempt to reestablish that connection when you start the system. If you have statements in your STARTUP.CMD file to start the network requester and log on to the network, the Workplace Shell may try to reestablish the network connections before the logon has been completed. In this case, the system will prompt you for your password. You can stop this behavior by removing the CONNECTIONS keyword from the SET AUTOSTART= statement in the CONFIG.SYS file. If you remove the CONNECTIONS keyword, you will have to reestablish all network connections yourself. You may want to remove the keyword if you start the requester, log on, and establish network connections in your STARTUP.CMD file.

- LAUNCHPAD— This automatically allows the Workplace Shell to create the LaunchPad object (if it doesn't exist) and open the LaunchPad.

If you do not want to use the LaunchPad, remove this keyword from the SET AUTOSTART= statement, delete the LaunchPad object, and restart your system. The LaunchPad will be gone forever!

If your LaunchPad disappears and double-clicking on an open space in a folder doesn't bring it back or if you deleted the LaunchPad object and now you want it back, make sure the LAUNCHPAD keyword is specified and restart your system. The Workplace Shell will create a new LaunchPad object (with its default buttons) and open it.

SET RESTARTOBJECTS= statement

While you are opening and closing objects, the Workplace Shell is keeping a running list of all open objects. The Workplace Shell also maintains a list of all objects that are open in each Work Area folder. You can also put objects or shadows of objects into the Startup folder. When the system is restarted, these lists and the Startup folder can be used to restore your Desktop to the state it was in when you shut down the system. The SET RESTARTOBJECTS= statement tells the Workplace Shell which of these lists to use when restarting the system.

You may specify one of the following options on the SET RESTARTOBJECTS= statement:

- YES — Reopen all objects that were open when the system was shut down and open all objects in the Startup folder. This is the default value.
- NO — Don't reopen any objects. The only objects that will be reopened automatically are the Desktop folder and the LaunchPad.

Tip

The Comet Cursor program does not automatically restart if RESTARTOBJECTS is set to either NO or STARTUPFOLDERSONLY. You can bypass this problem by putting the statement **START COMETRUN.EXE** in your STARTUP.CMD file.

- STARTUPFOLDERSONLY — Open only the objects in the Startup folder. This allows you to control the objects that are automatically opened when you restart your system. If you specify this option, the objects that were open when you shut down the system are ignored. (This option is quite a mouthful: **SET RESTARTOBJECTS=STARTUPFOLDERSONLY.** You have to type the entire statement correctly, or it will be ignored.)

Developer's Desktop

I have **SET RESTARTOBJECTS=STARTUPFOLDERSONLY** specified in my CONFIG.SYS file. With this option specified, I know exactly how my system will restart. I control the programs that are automatically started by putting the appropriate commands into my STARTUP.CMD file, and I control the Workplace Shell objects that are automatically opened by putting shadows of the objects I want into the Startup folder. I prefer this option to the default desktop metaphor, which is to restart the system in the same state it was in when I shut down the system.

- REBOOTONLY — Reopen objects only if the Workplace Shell is being started because the system was just turned on or you restarted the system by pressing **Ctrl-Alt-Del.** This keyword, which can be combined with one of the other options, indicates that the Workplace Shell is not to reopen any objects if it is being restarted due to a failure of the Shell or due to a program requesting the Shell to be closed and restarted. If you don't specify this option, all objects in the Startup folder and all objects that were open when the system was shut down are reopened every time the Workplace Shell is started (as specified by the other options).

SET SHELLEXCEPTIONHANDLER= statement

The Workplace Shell normally handles exceptions that occur during its execution by cleaning up (as much as possible) and terminating the failing function. Most of the time, you don't even know there has been a system failure. If things get really bad, the Workplace Shell terminates itself. When this happens, you see the Desktop folder close, you normally see a message box explaining that an error occurred, and the Workplace Shell restarts itself and reopens the Desktop folder.

If you are trying to debug a problem, either one in the Workplace Shell itself or in an object class you are developing, these actions may cover up the problem you are trying to find. The SET SHELLEXCEPTIONHANDLER= statement can be used to stop the Workplace Shell from handling exceptions.

If you add the statement **SET SHELLEXCEPTIONHANDLER=OFF** to the CONFIG.SYS file, the Workplace Shell will let the system handle exceptions within the Workplace Shell. If an exception occurs, the system will display a message box giving you information about the failure that might be used to diagnose the problem.

Tip

This environment variable controls exception handling only within the Workplace Shell. It does not affect other OS/2 components or applications supplied by Workplace Shell object classes. It also doesn't affect any object class you are developing. If an exception occurs in a component other than the Workplace Shell, it will be handled normally.

SET AUTOREFRESHFOLDERS= statement

Whenever a file is created, deleted, copied, moved, or renamed, the File System informs the Workplace Shell about the change. This information is used to update the contents of the folder object that correspond to the directory containing the affected file (or both directories in case a file is moved from one directory to another). It is also used to update a Program Reference object that refers to the file, allowing you to see the current state of all files in an open view of a folder.

These file changes can be caused by a program you invoke or by commands you execute at an OS/2 or DOS command prompt.

Tip

It takes the system time to process these notifications. If you change files often but rarely have an open view of the folders in which these changes are taking place, you can improve the performance of your system by specifying **SET AUTOREFRESHFOLDERS=NO** in the CONFIG.SYS file. This stops the Workplace Shell from trying to find the folder in which these changes are reflected. If you do have a folder open when a file is changed, you can see the change by selecting the **Refresh now** item in the folder's pop-up menu.

SET OBJECTSNOOZETIME= statement

Whenever you open a view of a folder, the Workplace Shell has to "wake up" all objects within that folder to find their icons and display their pop-up menus or perform other operations on them. The system has to use memory to keep each object awake. When you close the folder view, the Workplace Shell can put these objects right back to sleep. But, if you closed the folder and then reopened it, the Workplace Shell would have to wake up the objects again, taking a noticeable amount of time.

Uncensored

When you close an open view of a folder, the Workplace Shell puts all objects within that folder on a list of objects that are "snoozing" (actually, more like taking a catnap). When the objects have had a long enough nap (that is, you haven't reawakened them), the Workplace Shell decides that it is safe to put them to sleep. If you reopen the folder view while the objects are "snoozing," they can be awakened immediately.

You can control the "snooze" time by adding the **SET OBJECTSNOOZETIME=time** statement into the CONFIG.SYS file, where "time" is the number of seconds that objects "snooze" before being put to sleep. The default value is 30 seconds. Do not set the "snooze" time to zero.

Tip

If you frequently close and then reopen the same folder, you might want to increase the "snooze" time so that objects are not put to sleep so quickly. However, if you normally don't reopen a folder right after you close it, you might want to decrease the "snooze" time so that objects will go to sleep more quickly, freeing their memory for other uses.

Note

There is one exception to the general statement that the Workplace Shell puts objects in a folder to sleep when the folder is closed. The Workplace Shell preloads several folders when it starts running. These folders include the OS/2 System folder, the Drives folder, the System Setup folder, the Command Prompts folder, the Information folder, the Templates folder, the Productivity folder, and the Games folder. The objects within these folders are *never* put to sleep, allowing the folders to open quickly any time you need them.

SET DEFERICONLOAD= statement

Uncensored

If you used a version of OS/2 previous to OS/2 Warp, you may have noticed that we changed the way we display objects in an Icon View of a folder. In previous versions of OS/2, we retrieved the icon for every object in the folder while the folder was opening. If you had folders that contained many more objects than would fit on the screen, it would sometimes take a very long time to open the folder. It appeared that the visible icons "popped in" faster, but you then had to wait while we loaded all of the icons for objects you couldn't see until you scrolled forward through the folder view. Besides wasting time, it took a large amount of system memory to load all of those icons.

In OS/2 Warp, we've changed this process. We now load icons only when the folder tries to display them. Sometimes, the visible icons may appear to "pop in" more slowly. However, the overall time taken (and the memory required) to open a folder has decreased dramatically since we are no longer loading the icons that are not yet visible. When you scroll forward in the folder, you might notice the delay caused by the Workplace Shell loading the new set of icons for you to see. This performance improvement is most noticeable for folders that contain lots (hundreds) of objects.

Tip

If you don't like this new behavior, you can add the **SET DEFERICONLOAD=NO** statement into the CONFIG.SYS file. This command requests the Workplace Shell to load the icons as it awakens each object and not wait for the icon to be visible in the folder view.

SET MENUSTYLE= statement

You may have noticed that the pop-up menu for certain objects can get rather large. Many of the items in the pop-up menu are not needed because these actions can be performed directly using the mouse or the keyboard. The statement SET MENUSTYLE= allows you to remove these redundant items from the pop-up menus.

If you specify SET MENUSTYLE=SHORT, the Workplace Shell will remove the following options from all pop-up menus: Create another, Move, Copy, Create shadow, Print, Delete, Help, and Find. (The first six options can be performed using drag and drop, Help can be obtained by pressing **F1**, and the Find function can be invoked from the LaunchPad.)

Setting the global environment

You can invoke OS/2 Warp programs under the Workplace Shell by using a Program Reference object or by starting a Command Processor. You can use the SET statement in the CONFIG.SYS file to set values into the global environment used every time a program is started from the Workplace Shell.

Some of these variables can be used to make much more efficient use of the system.

SET OS2_SHELL= statement

You probably already know that this statement can be used to specify the Command Processor that is to be given control when you specify * as the program name in a Program Reference object. But, it has another use.

Shhh! Tip

You can specify a command to be executed by the Command Processor *before* it presents you with a command prompt. This means that you can predefine your command prompt environment.

Developer's Desktop

For example, I want my Command Processor windows to be 40 lines high instead of the default 25 lines. I added the following statement into my CONFIG.SYS file: **SET OS2_SHELL=D:\OS2\CMD.EXE /k mode co80,40**. The **/k** parameter tells the command processor to execute the following statement, then to present me with a prompt. The **mode** command sets the video mode in the window to 40 lines of 80 characters. By putting this statement into the CONFIG.SYS file, the **mode** command is executed every time I open an OS/2 command window.

Shhh! Tip

You can get almost the same effect if you put **/k mode co80,40** into the parameter field of the OS/2 Window Program Reference. However, there is one big difference. If the mode command is executed as a parameter of the Program Reference object, the Command Processor will not change the title bar and the Window List to reflect the name of each command it executes. If you put the mode command into the OS2_SHELL environment variable, the Command Processor will change the title bar and the Window List every time you invoke a command.

LIBPATH= statement

Many OS/2 programs use dynamic link libraries to contain code that can be shared between programs. The LIBPATH= statement tells OS/2 Warp where to look for these dynamic link libraries.

The LIBPATH= statement is normally modified by any application program you install if the program requires the use of dynamic link libraries (DLLs). The LIBPATH= statement can, however, cause performance problems because the system has to search many DLLs when trying to find functions needed by a program, and the LIBPATH= statement can become longer than 255 characters, which is a limit for some text editors.

So, what can you do? You can help alleviate these problems by making use of the BEGINLIBPATH and ENDLIBPATH environment variables.

After you install an application program, look at the CONFIG.SYS file and see what the program added to the LIBPATH= statement. If it added one or more DLLs to the LIBPATH= statement, they are probably needed only when that program is invoked. Look at the Program Reference objects the program installed. If they refer to batch files, it is easy to fix this problem, as follows:

1. Edit the batch file for the program with your favorite text editor.
2. If the program added its DLLs to the beginning of the LIBPATH= statement, add to the beginning of the batch file a statement that reads **SET BEGINLIBPATH=dlls,** where **dlls** are the names added to the LIBPATH= statement. If the program added its DLLs to the end of the LIBPATH= statement, add the statement **SET ENDLIBPATH=dlls** to the beginning of the batch file.
3. Edit the CONFIG.SYS file and remove the DLLs from the LIBPATH= statement.

It is as simple as that. The DLLs will now be available for the program that needs them and will not cause the system to search through these application-specific DLLs when it is running other programs.

If the Program Reference objects directly invoke executable programs, you can fix the problem as follows:

1. Use a text editor to create a batch file to invoke the program.
2. Put a SET **BEGINLIBPATH=dlls** or SET **ENDLIBPATH=dlls** statement at the beginning of the batch file.
3. Add a statement to invoke the program. You can copy the program name and parameters from the Program Reference object.
4. Change the Program Reference object to invoke the batch file instead of the program.
5. Edit the CONFIG.SYS file and remove the DLL names from the LIBPATH= statement.

SET PATH= and SET DPATH= statements

The SET PATH= statement is used to define the set of directories to search for an executable program, and the SET DPATH= statement defines the set of directories to search for data files. Guess what? The SET PATH= and SET DPATH= statements can have the same problem as the LIBPATH= statement: Application programs overload these global environment variables so that their programs can be found easily. In most cases, this problem can be resolved by appending the appropriate paths to local environment variables instead.

You can use the same technique to shorten the SET PATH= and SET DPATH= statements you use for the LIBPATH= statement:

1. Edit the batch file used to invoke the program (or create one).
2. Look at the CONFIG.SYS file and see where the program's directory was added to the SET PATH= statement. If the program added paths to the beginning of the statement, add to the batch file the statement SET PATH=**new-paths**;%PATH% (where **new-paths** is the set of paths added by the application program). If the program added its paths to the end of the statement, add the statement SET PATH=%PATH%;**new-paths** to the batch file. (The ";" character is needed only if the global SET PATH= statement doesn't end with a ";".) This adds the required paths to the PATH environment variable only for this program, not globally.

3 Follow the same procedure for the SET DPATH= statement. The data paths will also be local.

4 Edit the CONFIG.SYS file and remove the added paths from the PATH= and DPATH= statements.

Note

The Workplace Shell uses the global PATH= environment variable when it is trying to load a program you specify in a Program Reference object. If the program you specify is not in a directory listed in the global PATH= environment variable, you should add its path into the Program Reference object rather than change the global SET PATH= statement in the CONFIG.SYS file.

SET TMP= statement

Many programs create temporary files during their execution and need the name of the directory to use in creating these files. The Workplace Shell is no exception. For example, if you select the **Edit...** push button on the General page of an object's Settings Notebook, the Workplace Shell must create a temporary file containing the icon data to pass to the Icon Editor.

The Workplace Shell uses the path specified by the SET TMP= statement as the directory in which to create its temporary files. If you don't specify a SET TMP= statement in the CONFIG.SYS file, the Workplace Shell uses the directory in which the OS2.INI file resides for all temporary files.

SET HELP= statement

The SET HELP= statement specifies the list of directories that contain help files. The System Installation program sets this variable to point to the directories containing the OS/2 help files. When you install an application program that uses the OS/2 help facility, it adds its directories to this statement.

The paths specified in the HELP global environment variable are used for three functions:

1 The Master Help Index object uses the HELP environment variable to find all available help files and to build the Master Index that is presented to you.

2 When you request Help for an application program, the help file for that program can reside either in the current directory or in one of the directories specified by the HELP environment variable.

3 If you are displaying Help information and you request a search of all libraries, the help files in the directories specified by the HELP environment variable are searched for the requested information.

SET GLOSSARY= statement

The SET GLOSSARY= statement specifies the list of directories containing glossary files. The System Installation program sets this variable to point to the directory containing the OS/2 Glossary file.

The paths specified in the GLOSSARY global environment variable are used to build the glossary presented to you when you open the Glossary object.

SET BOOKSHELF= statement

The SET BOOKSHELF= statement specifies the list of directories containing book files. The System Installation program sets this variable to point to the directories containing the OS/2 books. When you install an application program, it may append the name of its book directory to this variable.

The paths specified in the BOOKSHELF global environment variable are used for two functions:

1 When you invoke the VIEW command and specify the name of an .INF file, the View program looks for the book in the directories specified by the BOOKSHELF environment variable.

2 If you are displaying a book and you request a search of all libraries, the book files in the directories specified by the BOOKSHELF environment variable are searched for the requested information.

Starting the DSOM environment

OS/2 Warp supports the Distributed System Object Model (DSOM). In order to use a DSOM client program, you must have two other programs running: the DSOM daemon process (background server process) and a DSOM server program. The DSOM support within OS/2 Warp supplies the daemon, and the Workplace Shell supplies a server.

Starting the DSOM daemon process

In order for any DSOM server to be started, the DSOM daemon process must be running. There are three ways to start the DSOM daemon process:

1 The SET WPDSERVERSTART=ON statement in the CONFIG.SYS file starts both the DSOM daemon and the Workplace Shell DSOM server.

2 The SOMDD command starts the DSOM daemon process directly.

Warning

Do not use this command if you are using the Workplace Shell DSOM server. Even though the SOMDD command starts the daemon process, the Workplace Shell doesn't know that it is running and can't provide any error recovery in case something goes wrong with it.

3 The WPDSINIT command can be used to start the DSOM daemon and the Workplace Shell DSOM server. See Chapter 15 for a complete description of the WPDSINIT command.

Tip

If you are going to run a DSOM client program often, use the statement SET WPDSERVERSTART=ON in the CONFIG.SYS file to start the DSOM daemon and the Workplace Shell DSOM server. This way you can set up the program once and forget about it. The DSOM daemon and server will automatically start every time you start your system. If you are going to use a DSOM client program only occasionally, you should use the WPDSINIT command to start the DSOM daemon and server when you need it.

The DSOM daemon process normally doesn't display any messages. However, if you are developing a DSOM client program and need to see any diagnostic messages generated by the DSOM daemon process, you can include the statement SET SOMDD.DISPLAY=ON in the CONFIG.SYS file. This statement instructs the DSOM daemon process to open a window in which it displays its messages.

Starting the Workplace Shell DSOM server

You may use various server programs with your DSOM client. The Workplace Shell supplies a DSOM server program that you can use.

There are three ways of starting the Workplace Shell DSOM server program:

1 The SET WPDSERVERSTART=ON statement in the CONFIG.SYS file starts both the DSOM daemon process and the Workplace Shell DSOM server program.

2 The WPDSINIT command can be used to start both the DSOM daemon process and the Workplace Shell DSOM server program.

3 The DSOM daemon process will automatically start the Workplace Shell DSOM server program if a DSOM client requests service from either the Workplace Shell DSOM server or from any DSOM server.

Tip

There is normally no need for you to start the Workplace Shell DSOM server explicitly as long as you have the DSOM daemon running when you need it.

Running Programs

OK, so I lied. I said I wouldn't talk about any CONFIG.SYS statements that weren't directly related to the Workplace Shell.

RUN= statement

The RUN= statement is used to start a detached process, just like the DETACH command (see Chapter 15). However, there are some important differences.

The DETACH command is invoked from an OS/2 Warp command prompt after the Workplace Shell is running. The entire system is initialized at this time — PM and the Workplace Shell are running and all daemons (including the DSOM daemon) are available to process requests. When you run a program from the RUN= statement, all you can count on is that the base operating system has been initialized. If the program you invoke is a daemon process, that is OK.

Warning

However, if the program you invoke requires Presentation Manager or Workplace Shell services, you may have some serious problems. For example, if the program tries to create a PM message queue and the Protected Mode Shell hasn't started yet, your program could be the first program to create a PM message queue. Since PM thinks the first program that creates a message queue is the Protected Mode Shell, your program will probably confuse the system beyond all hope!

Note

You must be aware that all RUN= statements are processed at the end of System Initialization just before the Protected Mode Shell is started. The programs invoked by the RUN= statements are started asynchronously, so they will compete with themselves and the Protected Mode Shell for the system. You cannot guarantee the order in which these programs will actually execute — you can only guarantee that they are started in the order you specify.

CALL= statement

The CALL= statement is used to run a program before System Initialization starts the Protected Mode Shell. The programs invoked by the CALL= statements are run synchronously. Each program must complete and return control to System Initialization before the next program is run. After the last CALL= statement is processed, System Initialization invokes the Protected Mode Shell.

Tip

The CALL= statement allows you to perform some last-minute updates to the system just before invoking the Protected Mode Shell. For example, if you are developing a Workplace Shell object class, you can put your latest version of the dynamic link library into some directory (such as \TESTDLL on the boot drive) and put the following statement into the CONFIG.SYS file: **CALL=D:\OS2\CMD.EXE /C COPY D:\TESTDLL D:\OS2\DLL** on the boot drive. This copies your test DLL into the \OS2\DLL directory. You normally can't update a Workplace Shell dynamic link library while the system is running, since all DLLs that are in use are locked. The CALL= command allows you to update the DLL before the Workplace Shell locks the file.

Tip

You can also get to a command prompt and modify anything you like before the Protected Mode Shell is invoked. To do this, add the statement **CALL=D:\OS2\CMD.EXE** to the CONFIG.SYS file. After you are done modifying the system, you would let System Initialization complete (and invoke the Protected Mode Shell) by entering an **EXIT** command at the command prompt. For example, if you are testing various DLLs, you might want to remove one of your test DLLs before invoking the Protected Mode Shell. You can do this at the command prompt.

Developer's Desktop

Here is something I use the CALL= command for. I have a MediaVision Pro Audio 16 sound adapter in my system at home. This particular sound adapter monitors the signal lines going to the computer's speaker and reproduces the speaker sound. This might be a good feature except that it plays the speaker sounds too loudly. I found a shareware OS/2 program that allows me to set the volume for each section of the sound card. I added the following statement into my CONFIG.SYS file to change the volume for the speaker emulation to a reasonable level: **CALL=C:\PROAUDIO\PAS.EXE SET SPKR TO 40.**

I did not want to put this statement into the STARTUP.CMD file because the PAS.EXE program can't affect the sound card while the card is in use. If I have a long-running system start-up sound effect playing, the sound card is in use when the STARTUP.CMD file is invoked, and the PAS.EXE program doesn't work. By running the PAS.EXE command from a CALL= statement, I am guaranteed that the sound card is not being used when the PAS.EXE command is invoked.

Summary

That's it for the CONFIG.SYS file. For the most part, you don't have to modify this file unless you want to do some special customization.

These are the main things I changed in my own CONFIG.SYS files:

- I removed CONNECTIONS from the AUTOSTART environment variable so that I don't have to respond to the password prompt.
- On my test system at work, I added CALL= statements to copy DLLs I am testing to a directory on the LIBPATH and to invoke CMD.EXE. On my system at home, I added a CALL= statement to change the volume settings for my sound card.
- I moved application-specific DLLs and paths from the LIBPATH, PATH, and DPATH statements to local environment variables for those applications.
- I added other SET statements to define global environment variables that I can test within some of my REXX scripts.
- I added the **SET RESTARTOBJECTS=STARTUPFOLDERSONLY** statement.
- I added the command **/k mode co80,40** to the OS2_SHELL environment variable.

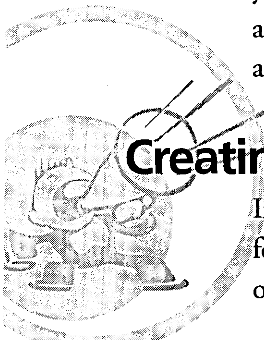
Ira H. Schneider is an Advisory Programmer on the Workplace Shell Development Team. He has been instrumental in developing OS/2 Warp, including performance improvements and functional enhancements to the Workplace Shell. He has also worked on the OS/2 Swapper and on OS/2 System Initialization, and he was Lead Architect for OS/2 1.2. Ira has been with IBM since 1969.

Chapter 11

Boot Time Services

Saved by the Shell

OS/2 Warp gives you access to two services that can be utilized before the operating system itself is booted. These services, both of which are described during this chapter, are the Boot Manager and the Archive and Recover feature. The Boot Manager allows you to manage which operating system is started. The Archive and Recover feature allows you to archive your desktop directory and several key files and then restore them at boot time if needed.



Creating a Boot Manager

In this section I will give you a brief overview of working with the Boot Manager followed by a complete example of creating a Boot Manager partition and adding two operating systems to the Boot Manager.

Boot Manager overview

You can use the Boot Manager to control which operating system the computer should run:

1. Use the **FDISK** or **FDISKPM** command to create your Boot Manager partition.
2. Use **FDISK** or **FDISKPM** to create your primary and/or logical partitions.
3. Assign each partition that has an operating system on it a name that will appear on the Boot Manager screen.
4. Make the Boot Manager partition the Startable partition.
5. Close **FDISK** or **FDISKPM**, shut down the system, and then reboot.

You will be shown the Boot Manager selection screen. Choose the operating system you wish to start and then press **Enter**. For more information on drives, partitions, and the FDISK and FDISKPM commands, please see Chapter 15.

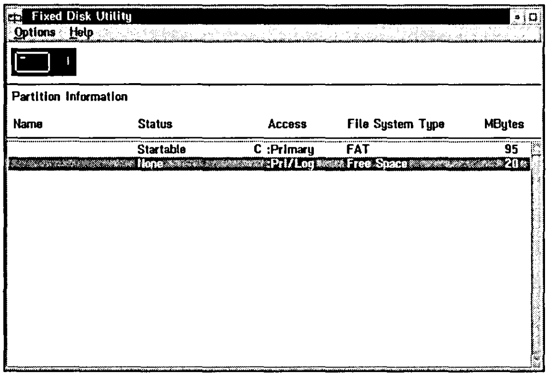
The information given above was an overview of the process in a nutshell. What follows is a complete walk-through of the process.

Boot Manager example

For this example I will be adding the DOS operating system as an additional C: drive partition. I will be using the FDISKPM command to create and configure the Boot Manager and the DOS partitions. If you attempt to do this before the OS/2 Warp operating system is installed, you will need to run the FDISK command instead. You can run the FDISK command from the install media.

In Figure 11-1 we see the FDISKPM screen showing the current configuration of the hard drive. The machine being used has a single hard drive as indicated by the single picture of a hard disk labeled with a 1. The hard drive has been partitioned with OS/2 Warp receiving 95MB of disk space of the 115MB originally available on the hard disk. There are currently 20MB available. Normally I would have used the additional 20MB as a data partition by configuring it as my logical drive D. For the purposes of this example we will make it a primary partition and install the DOS operating system on it. But first we have to add Boot Manager.

Figure 11-1. FDISKPM Screen Before Boot Manager Is Added



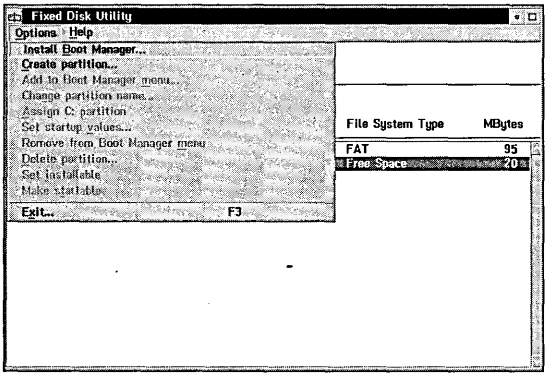
Partition Information

Name	Status	Access	File System Type	MBytes
	Startable	C :Primary	FAT	95
	None	:Pri/Log	Free Space	20

Creating the Boot Manager partition

We create the Boot Manager partition by first selecting the line with the Free Space File System Type and choosing the Options menu. From the Options pull-down menu we choose the **Install Boot Manager...** menu choice as shown in Figure 11-2.

Figure 11-2. Install Boot Manager Menu Choice



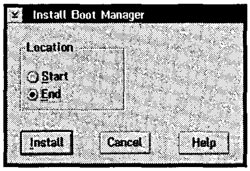
File System Type	MBytes
FAT	95
Free Space	20

You are now at the Install Boot Manager dialog box as shown in Figure 11-3. Your choices are to either:

✗ install Boot Manager at the Start of all the free space on the hard disk or

✗ install Boot Manager at the End of all the free space on the hard disk.

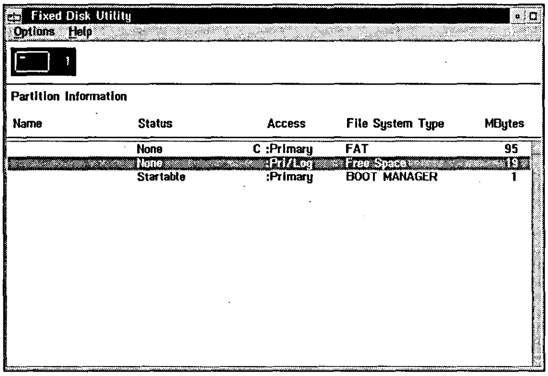
Figure 11-3. Install Boot Manager Dialog



I have chosen the second option, installing Boot Manager at the End of all the free space. This is just a preference. I like the Boot Manager partition as either the first or the last partition on the disk. If it is in the middle, as it would end up if I had chosen the first option, it just violates my sense of order.

So after making my choice of End, I then choose the Install push button to actually have the Boot Manager partition created. This results in an FDISKPM screen as shown in Figure 11-4. The Boot Manager partition is marked as Startable for you by the FDISKPM program. As we indicated in our selection on the Install Boot Manager dialog box, it is at the end of the hard disk. You will notice under the MBytes column that the Boot Manager partition consumes one full megabyte of our hard disk. We now have 19 megabytes available for our DOS partition.

Figure 11-4. Boot Manager Partition Created



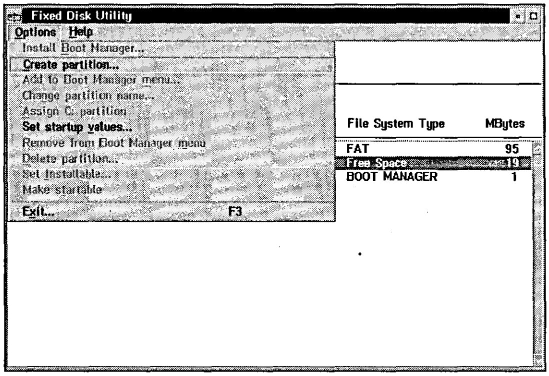
Partition Information

Name	Status	Access	File System Type	MBytes
	None	C :Primary	FAT	95
	None	:Pri/Log	Free Space	19
	Startable	:Primary	BOOT MANAGER	1

Creating the DOS partition

To create the DOS partition you need to choose the **Create partition...** menu choice from the Options menu as shown in Figure 11-5 with the Free Space File System Type selected.

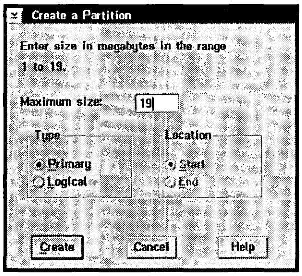
Figure 11-5. Create Partition Menu Choice



File System Type	MBytes
FAT	95
Free Space	19
BOOT MANAGER	1

In the Create a Partition dialog box, as shown in Figure 11-6, you describe the partition you want to create and then choose the **Create** push button. The fields of the dialog box are described below:

- ✗ **Maximum size** — This is the size, in megabytes, you want the new partition to be. You can choose any number in the range displayed above the entry field.
- ✗ **Type** — Choose the partition type, either Primary or Logical. Since the DOS operating system can only be installed on drive C:, I have to choose Primary. If I was just creating a data partition or if I was preparing to install a different version of OS/2, I would have chosen the Logical type of partition.
- ✗ **Location** — If I had decided to create the partition smaller than its maximum possible size, I would have been given the choice of starting locations, just like I was when I created the Boot Manager partition. Your choice will depend on how you want your drives to be lettered or displayed on the Boot Manager screen.



Enter size in megabytes in the range 1 to 19.

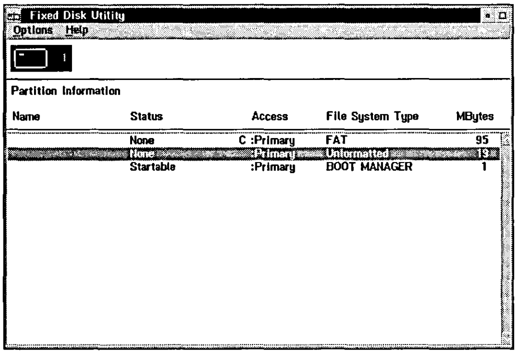
Maximum size: 19

Type
● Primary
○ Logical

Location
● Start
○ End

Figure 11-6. Create a Partition Dialog

Figure 11-7 shows the FDISKPM screen after the new partition is created. Since it is a primary partition it shares the C drive letter with the other primary partitions on the same physical disk.



Partition Information

Name	Status	Access	File System Type	MBytes
	None	C :Primary	FAT	95
	None	:Primary	Unformatted	19
	Startable	:Primary	BOOT MANAGER	1

Figure 11-7. DOS Partition Created

Configuring the Boot Manager menu

Now that the new partitions are created we have to add the operating system partitions to the Boot Manager menu. To do that you choose the **Add to Boot Manager menu...** menu choice from the Options menu as shown in Figure 11-8. I will be adding to both my existing OS/2 Warp partition and to my new DOS partition.

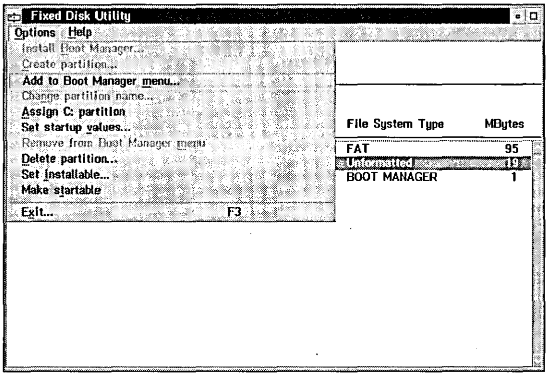
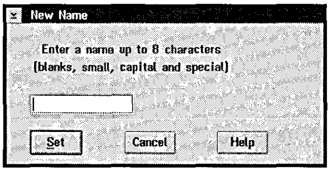


Figure 11-8. Add to Boot Manager Menu Choice

After choosing the Add to Boot Manager menu choice you are routed to the New Name dialog box as shown in 11-9. In this dialog box you enter a descriptive name for your partition and then choose the **Set** push button. We need to do this twice, once for the OS/2 Warp partition and again for the new DOS partition.

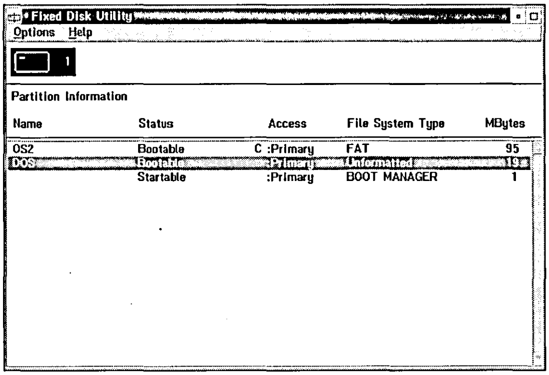


Enter a name up to 8 characters
(blanks, small, capital and special)

Figure 11-9. New Name Dialog

After the partitions are named then the FDISKPM screen will appear as shown in Figure 11-10.

Figure 11-10.
Partitions Named

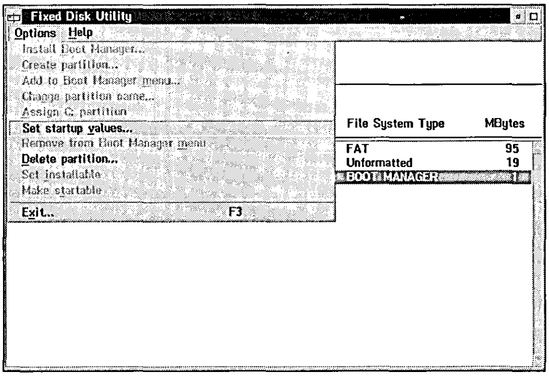

Fixed Disk Utility

Partition Information

Name	Status	Access	File System Type	MBytes
OS2	Bootable	C :Primary	FAT	95
DOS	Bootable	:Primary	Unformatted	19
	Startable	:Primary	BOOT MANAGER	1

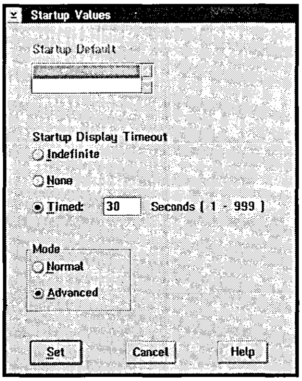
In Figure 11-11 I am selecting the **Set startup values...** menu choice from the Options pull down menu. This leads to the **Startup Values** dialog box as shown in Figure 11-12.

Figure 11-11.
Set Startup Values
Menu Choice


Fixed Disk Utility

File System Type	MBytes
FAT	95
Unformatted	19
BOOT MANAGER	1

Figure 11-12.
Startup Values
Dialog


Startup Values

Startup Default

Startup Display Timeout
○ Indefinite
○ None
● Timed: 30 Seconds (1 - 999)

Mode
○ Normal
● Advanced

The Startup Values dialog box allows you to set the characteristics of the Boot Manager screen. The fields of the Startup Values dialog box are described below:

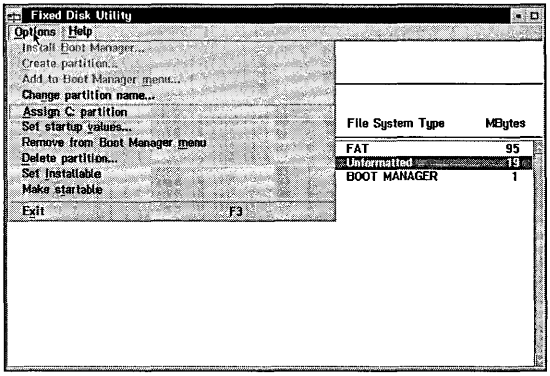
- **Startup Default** — Use this field to select which partition will be the default partition to start if the Startup Display Timeout time elapses. In order to use this field you have to have had one of the named partitions highlighted when you selected the Set startup values menu choice.
- **Startup Display Timeout** — Use the **Indefinite** choice to cause the Boot Manager to wait for the user to make an operating system selection from the Boot Manager screen. Use the **None** choice to have the Boot Manager start the default partition immediately. Use the **Timed** choice to have the Boot Manager offer the user the number of seconds specified to make a choice of which operating system to start before the Boot Manager automatically starts the default partition.
- **Mode** — Choose the **Normal** mode to display a basic Boot Manager. Use the Advanced mode to have the Boot Manager display more detailed information about each partition on the boot manager screen.

Setting the C: Partition

The DOS partition has been created but as you will notice in Figure 11-10, it is still unformatted. In order to install DOS onto the new partition we will need to boot from a diskette. We will want the new partition to be the visible partition when we boot from a diskette. Currently the OS2 partition is the visible partition.

As shown in Figure 11-13 I am choosing the **Assign C: partition** menu choice from the Options menu after first ensuring that the new partition was the selected partition in the FDISKPM window.

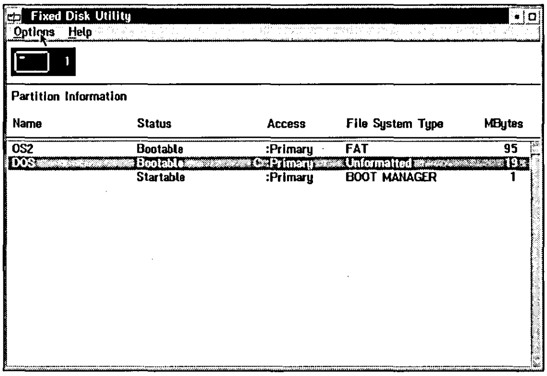
Figure 11-13.
Assign C: Partition Menu Choice



File System Type	MBytes
FAT	95
Unformatted	19
BOOT MANAGER	1

This will result in an updated FDISKPM window as shown in Figure 11-14, which confirms that the DOS partition will be the visible C: drive when the system is booted from diskette.

Figure 11-14.
DOS as C: Drive

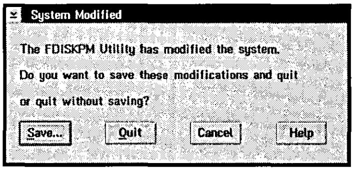


Partition Information

Name	Status	Access	File System Type	MBytes
OS2	Bootable	:Primary	FAT	95
DOS	Bootable	C:Primary	Unformatted	19
	Startable	:Primary	BOOT MANAGER	1

Now that all the changes have been made it is time to close the FDISKPM utility. After choosing **Close** from the System menu I am presented with the System Modified dialog box as seen in Figure 11-15. I choose the **Save...** button to save all of my changes. At this point I shut my system down, boot my DOS diskette to format, and install DOS on that partition. When I next boot my system, I will be presented with the Boot Manager screen, and I can now start either the DOS or the OS/2 Warp operating systems.

Figure 11-15.
System Modified Dialog



The FDISKPM Utility has modified the system.

Do you want to save these modifications and quit or quit without saving?

Using the Archive and Recover Feature

The Archive and Recover feature allows you to:

✗ access an OS/2 Command Line before the system is booted

✗ reset your display back to VGA mode

✗ restart the system at the Maintenance Desktop

- recover your archived Desktop
- archive your Desktop directory and several key OS/2 system files
- create unique CONFIG.SYS files for different boot environments

To gain access to these features, press **Alt-F1** when you see the white bar and OS/2 in the upper left-hand corner of your screen as the system is booting. You will be presented with the Recovery Choices screen from which you can access the Archive and Recover functions described below.

Accessing an OS/2 Command Line before the system is booted

If you find that you need to change your configuration or if you are having trouble starting your OS/2 Warp system you may want to gain access to an OS/2 Command Line so that you can perform whatever actions you need to take before the system starts. For example, you might want to use the **TEDIT** command to edit CONFIG.SYS. This way you do not have to wait for the operating system to completely boot to make your modifications just so you can then reboot. This option can be a real time saver under these circumstances. To access this function, press the letter C on the keyboard at the Recover Choices screen.

Note

In this scenario, OS/2 Warp uses the file C:\OS2\BOOT\CONFIG.X as the CONFIG.SYS file. It does not replace your CONFIG.SYS file, it just uses C:\OS2\BOOT\CONFIG.X as your configuration file. You are then placed at an OS/2 Fullscreen Command Line. After making your configuration changes you can either reboot your system with the **Ctrl-Alt-Del** key sequence or use the **EXIT** command to have the system reboot itself.

Resetting your display back to VGA mode

You may find that your system is having trouble operating at a higher resolution display mode. In this case you would have to reset your system back to VGA mode in order to make the modifications you need to in order to get your display working in the higher resolutions. To reset your display back to VGA mode, press V on the Recovery Choices screen. This will update your CONFIG.SYS file to operate correctly in VGA mode.

Restarting the system at the Maintenance Desktop

The OS/2 Warp operating system is installed in two phases. The first phase loads all the required parts of the operating system from the install media. You then boot OS/2 Warp to begin the second phase of the installation process. This phase is called the Selective Install phase, which is used to install the optional parts of the operating system from the install media. If you look at the Desktop during this phase it appears to be a normal OS/2 Warp Desktop but it is not. It is another Desktop called the Maintenance Desktop. If you want to get back to this Desktop you can press M from the Recovery Choices screen.

Recovering your archived Desktops

After phase two of install completes you then reboot your system for the first time. We refer to this as the First Boot. During the First Boot, when the Presentation Manager is first initialized but before the Workplace Shell is, a special archive, called the INSTALL archive, is created.

The INSTALL archive, like all Desktop archives, consists of a complete copy of the Desktop directory structure and all its files under the C:\OS2\ARCHIVE directory. The INSTALL archive is stored in the directory C:\OS2\ARCHIVE\0X. The first user archive is stored in the directory C:\OS2\ARCHIVE\01.

In addition to the Desktop directory structure, a set of key files is also archived. The set of files to be archived is stored in the file C:\OS2\ARCHIVES\OS2.KEY. By default, the file has the following contents:

```
KEYFILE:OS2.INI

KEYFILE:OS2SYS.INI

KEYFILE:C:\CONFIG.SYS

KEYFILE:C:\STARTUP.CMD

KEYFILE:C:\AUTOEXEC.BAT

KEYFILE:C:\OS2INIT.CMD
```

You can modify the contents of the C:\OS2\ARCHIVES\OS2.KEY file and add the names of additional files to be archived.

Tip

Type the letter X at the Recovery Choices screen to have the system restored to the state of the Desktop and key files as they existed at First Boot. For example, this is a great thing to do if you want to create a new Desktop.

Type the numbers 1, 2, or 3 at the Recovery Choices screen to have the system restored to the state of the Desktop and key files as they existed at the time the user created those archives. The numbers 1, 2, and 3 refer to the menu choices on the Recovery Choices menu for the user-created archives. The process of creating a user archive is discussed in the following section.

When you recover your system from an archive, your current system is archived under the directory C:\OS2\ARCHIVES\CURRENT. If you do another recover it destroys the contents of the C:\OS2\ARCHIVES\CURRENT directory before saving the current Desktop there.

Archiving your Desktop directory

OS/2 Warp allows you to store up to three user-created archives. If and when you create that fourth archive the oldest of the existing three archives is first discarded. The second oldest becomes the oldest and what was the previous newest becomes the second oldest.

Tip

I strongly recommend that after you get your OS/2 Warp system configured the way you want it, you create an archive of your current configuration. You do that by opening up the Desktop Settings Notebook to the Archive page, as shown in Figure 11-16, and performing the following steps:

1 Check the Create archive at each system restart check box.

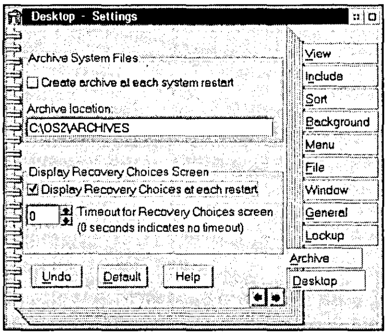
2 Shutdown OS/2 Warp.

3 Restart the system.

4 The archive gets created while the system is restarting.

5 Uncheck the Create archive at each system restart check box.

Figure 11-16. Archive Settings Page



Archive System Files

□ Create archive at each system restart

Archive location:

C:\OS2\ARCHIVES

Display Recovery Choices Screen

☑ Display Recovery Choices at each restart

0 Timeout for Recovery Choices screen (0 seconds indicates no timeout)

I recommend that you repeat this process whenever you make a configuration change to your Desktop environment or to one of the key files, such as CONFIG.SYS.

Creating unique CONFIG.SYS files for different boot environments

There may be situations in which you need to frequently boot your OS/2 Warp operating system under different configurations. A good example of this would be if you are a laptop computer user with a docking station. When using your system at the docking station you may require different device drivers loaded then when your system is not attached to the docking station.

The first step is to copy your current CONFIG.SYS file to a file called C:\OS2\BOOT\CONFIG.?, where ? is any single alphabetic character except X, C, M, or V. For the purpose of this example we will use the letter U, for undocked. So I will copy my CONFIG.SYS file to C:\OS2\BOOT\CONFIG.U.

Now I am going to modify my CONFIG.SYS file for the docked configuration. Then I will copy the new CONFIG.SYS file to C:\OS2\BOOT\CONFIG.D. The D stands for docked. I will then copy C:\OS2\BOOT\CONFIG.D or C:\OS2\BOOT\CONFIG.U back to CONFIG.SYS, depending upon which configuration I use more often. The CONFIG.SYS file is the one used if no special choices are made at the Recovery Choices screen or if the Recovery Choices screen is not accessed.

Your next step is to customize the Recovery Choices screen to display your new options. You do this by:

1. Running the command **ATTRIB -r C:\OS2\BOOT\ALTF1BOT.SCR**
2. Editing the file **C:\OS2\BOOT\ALTF1BOT.SCR**

3 Adding the following lines to the bottom of the file:

```
U       Press U to boot in your undocked configuration
D       Press D to boot in your docked configuration
```

Now when you access the Recovery Choices screen you will see your added choices. If you either bypass the Recovery Choices screen or press **ESC** your system will be booted with CONFIG.SYS.

Note

Your CONFIG.SYS file is not copied over; instead either the CONFIG.U or the CONFIG.D file is booted directly. Your CONFIG.SYS file is left undisturbed. Any changes you make to CONFIG.SYS may also need to be made to the CONFIG.U and the CONFIG.D files.

It is also possible to have a command file executed each time you choose one of your customized choices. For our U choice we need to create a file called C:\OS2\BOOT\ALTF1U.CMD. For our D choice we need to create a file called C:\OS2\BOOT\ALTF1D.CMD.

Tip

Remember, to have the Recovery Choices screen displayed every time the system is started, check the Display Recovery Choices at each restart checkbox on the Archive page of the Desktop Settings Notebook. I would also recommend setting the Timeout for Recovery Choices screen, on the Archive Settings Notebook page, to 30 seconds when using the Recovery Choices screen to control the starting of the system for multiple configurations.

Summary

These boot time options help make OS/2 Warp a flexible operating system. OS/2 Warp gives you many options for how to best work with your computer hardware and gives you the flexibility of determining the most appropriate operating system to run in different situations.

Peter G. Magid is an Advisory Programmer on the Workplace Shell Development Team. As the User Interface Design Lead for the OS/2 Workplace Shell, Peter has had chief design responsibilities for what users see and how they interact with the Workplace Shell. Peter also plays an active role in reviewing and consulting on the content and direction of the Workplace Shell programming model, and he developed the code for the LaunchPad.

Chapter 12

The BonusPak Applications

More Than Just the Fax, Please

The BonusPak, which ships with OS/2 Warp, is a suite of full 32-bit applications (see Figure 12-1). Some of these applications replace the applets in the Productivity folder from previous OS/2 versions and others add new applications and functions to your base operating system so you don't have to buy additional OS/2 applications.

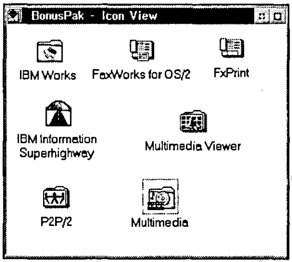
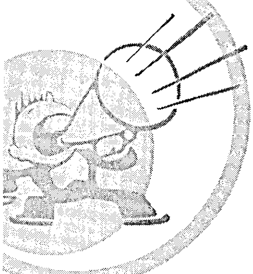


Figure 12-1. The IBM BonusPak



The BonusPak includes the following applications:

- ✗ **IBM Works** — An integrated package that gives you a word processor, spreadsheet, database manager and report writer, and the Personal Information Manager (PIM) to keep track of your appointments.
- ✗ **FaxWorks for OS/2** — An application that allows you to send and receive faxes through your PC.
- ✗ **The IBM Information Superhighway** — A collection of three communications applications
  - **IBM Internet Connection** — An application that enables you to access worldwide information, communicate with others, and retrieve software.
  - **CompuServe Information Manager for OS/2** — An application that allows you to access diverse information, gives you the ability to download software, and enables you to communicate with others.
  - **HyperACCESS Lite** — A program that provides several applications for accessing bulletin boards.
- ✗ **Person to Person** — An application that allows you to communicate with people at different locations as if you were all in the same place.
- ✗ **System Information Tool** — An application that enables you to retrieve information about your hardware setup.
- ✗ **VideoIN** — An application that allows you to create full multimedia productions including video, audio, text, and animation.
- ✗ **Multimedia Viewer** — An application that enables you to view multimedia productions even if you don't have the hardware needed to create such productions.

This chapter concentrates on the most popular applications, IBM Works and the IBM Information Superhighway. Overviews are offered for the other applications.

How to Install the BonusPak

Developer's Desktop

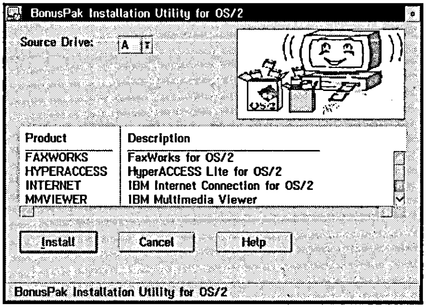
Below is the configuration of my own system setup. This will give you an idea of the kind of system that allows you to make full use of the BonusPak applications:

- IBM ThinkPad 700C, 486-SLC2 Processor, micro channel
- 12MB memory, 177MB hard disk
- Internal IBM (PNB) 9600 modem
- SVGA monitor with 512K video memory; 256 color, 640 × 480 resolution

The minimum hard disk requirement for installing the whole BonusPak is 38.5MB. The memory requirement depends on how many applications you are running, but with 12MB and a large swapper space available, you can use the BonusPak with reasonable performance.

If you have the diskette version of the BonusPak, you will find the BonusPak Installation Utility (see Figure 12-2) on the diskette labeled Installation Utility. If you have the CD-ROM version, the utility is stored in the directory for your country. The utility allows you to install the following applications:

- IBM Works (including PIM)
- FaxWorks for OS/2
- Three programs that are referred to collectively as the Information Superhighway
    - IBM Internet Connection for OS/2
    - CompuServe Information Manager
    - HyperACCESS Lite for OS/2
- Person to Person
- VideoIN
- Multimedia Viewer
- System Information Tool



BonusPak Installation Utility for OS/2

Source Drive: A

Product	Description
FAXWORKS	FaxWorks for OS/2
HYPERACCESS	HyperACCESS Lite for OS/2
INTERNET	IBM Internet Connection for OS/2
MMVIEWER	IBM Multimedia Viewer

BonusPak Installation Utility for OS/2

Figure 12-2.
The IBM BonusPak Installation Utility

The BonusPak Installation Utility allows you to install one, more, or all BonusPak applications via one installation panel.

Note

Before installing your BonusPak application using the BonusPak Installation Utility, be sure you have installed OS/2 REXX, because the utility will not work without it.

To install the BonusPak from diskettes, type **A:\INSTALL** at an OS/2 command prompt. To install it from a CD-ROM, change to your country directory (for example, CD \US for the United States) and type **INSTALL**. In both cases, you will get the BonusPak window, as shown in Figure 12-2.

Note

You may also install your BonusPak applications from a drive other than A: or your CD-ROM drive (for example, from a LAN drive). You just have to know the drive and directory where your BonusPak applications are located. The BonusPak Installation Utility has to be locally on your system; it cannot reside on a LAN server.

As soon as you start the BonusPak Installation Utility, the installation window will appear. The Source Drive field is set by default to A: or the letter for your CD-ROM drive. If you wish to install your BonusPak from another drive, you can select that here.

IBM Works

IBM Works (see Figure 12-3) includes a word processor, a spreadsheet, a chart program, a database, a report writer, and the Personal Information Manager (PIM) for your appointments. All IBM Works applications are 32-bit OS/2 applications written for OS/2 Warp. They are separate applications but act as an integrated set of programs.

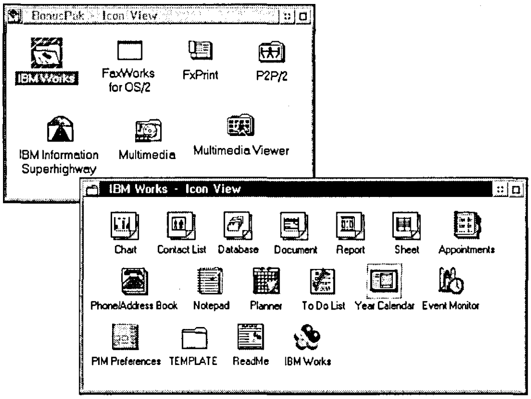


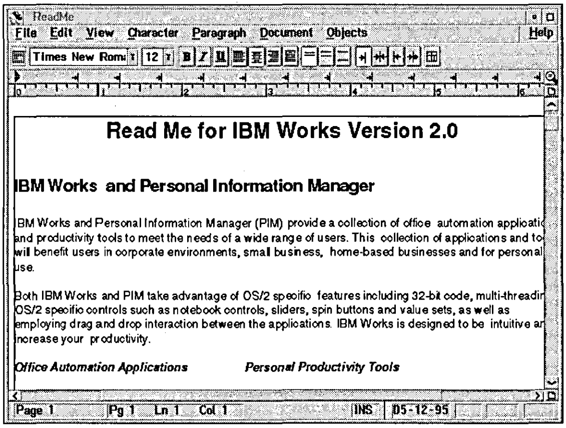
Figure 12-3.
The IBM Works Programs

Following are just some of the things you can do with the various applications included with IBM Works:

- Write letters, business reports, or other documents
- Organize your travel bills or manage your finances
- Create graphs of travel expenses
- Store information on the contents of your CD collection
- Keep track of your insurance clients
- Print reports on any information you keep track of
- Fax recipes to your mother or quotes to your customers
- Schedule appointments
- Keep the addresses of your friends and/or customers

IBM Works word processor

The IBM Works word processor (see Figure 12-4) is a fully functional 32-bit OS/2 program that has simple and easy-to-use functions and at the same time takes advantage of OS/2 multithreading. It also allows you to include tables and graphic objects in the documents you create.



Read Me for IBM Works Version 2.0

IBM Works and Personal Information Manager

IBM Works and Personal Information Manager (PIM) provide a collection of office automation applicati
and productivity tools to meet the needs of a wide range of users. This collection of applications and to
will benefit users in corporate environments, small business, home-based businesses and for personal
use.

Both IBM Works and PIM take advantage of OS/2 specific features including 32-bit code, multi-threadir
OS/2 specific controls such as notebook controls, sliders, spin buttons and value sets, as well as
employing drag and drop interaction between the applications. IBM Works is designed to be intuitive ar
increase your productivity.

Office Automation Applications *Personal Productivity Tools*

Figure 12-4. The IBM Works Word Processor

The BonusPak version of IBM Works is different from the existing product, Legato (which is available only in Europe, the Middle East, Africa, and Asia), in that it is an object-oriented application. This means that the BonusPak version supports drag and drop, as do other object-oriented applications. It also means that you do not start the IBM Works application in the traditional way by opening a file to edit. Rather, you work with data templates that provide you with the full functionality of the program (this method is also called "data-centric" as opposed to "application-centric").

A Template folder (see Figure 12-5) in the IBM Works folder provides you with prepared documents and spreadsheets. You can store the forms, letters, and spreadsheets you are using for your daily work without having to type the same things over and over again.

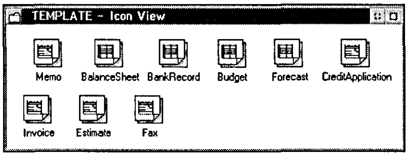


Figure 12-5. The IBM Works Template Folder

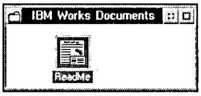
Note

Users who are familiar with Legato will notice that the BonusPak IBM Works folder contains a familiar icon labeled "IBM Works." If you click on this icon, you open the same menu you know from Legato. The menu was left to allow you to maneuver through IBM Works the conventional way until you are familiar with the object-oriented version.

The IBM Works ReadMe file gives you an overview of the full functionality of IBM Works. IBM Works does not have a tutorial but its extensive help feature guides you through its functions.

Because IBM Works uses the drag-and-drop function, you can easily exchange data between the IBM Works applications. For example, you can drag and drop one or more contacts from the Phone/Address Book or Contact List onto documents created with any of the other IBM Works applications. You may even attach a complete contact list to a particular document. As soon as you have dropped a contact or contact list on one of the IBM Works documents, it will display a graphic of a business card attached to the document icon (see Figure 12-6).

Figure 12-6.
IBM Works Document with Contact Attached

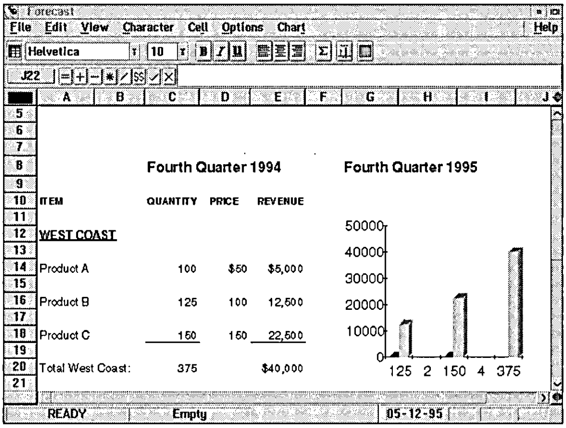


What advantage does this drag-and-drop feature offer you? You can drag and drop contacts to create distribution lists for a mail-merged document. As soon as you print the document, IBM Works merges the addresses with the document.

You may even combine the drag-and-drop functions of IBM Works with IBM FaxWorks for OS/2 (see the FaxWorks for OS/2 section for information on how to do this).

IBM Works spreadsheet

The IBM Works spreadsheet has all the basic functions you would expect from a spreadsheet program: Data is organized in columns and rows; you can carry out a variety of calculation functions on the data; several fonts and styles are available for formatting the data; you can make notes in specified cells; and you can also create charts through the chart function of IBM Works (see Figure 12-7).



	A	C	D	E
8		Fourth Quarter 1994		
10	ITEM	QUANTITY	PRICE	REVENUE
12	WEST COAST			
14	Product A	100	$50	$5,000
16	Product B	125	100	12,500
18	Product C	150	150	22,500
20	Total West Coast:	375		$40,000

Figure 12-7. IBM Works Spreadsheet and Embedded Chart

Using the chart function

For example, after you create a new spreadsheet or use one of the spreadsheet templates provided with IBM Works, suppose you want to create a bar chart of the data.

1 Select the area of your data that you want to see in a bar chart.

2 Click on the chart icon. You will see your mouse pointer turn into a crossbar.

3 Draw a rectangle by holding down mouse button 1, dragging the mouse to create a frame, and then releasing the button.

Now you will see a bar chart window within that spreadsheet. You can then print out the spreadsheet with your bar chart on it, or double-click on the chart if you'd like to modify it.

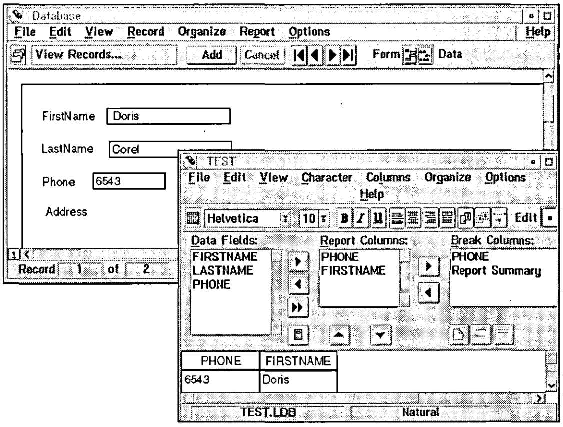
Linking spreadsheets, charts, and text documents

But what if you need that chart embedded in your text document? Simply copy the chart to the clipboard, paste it into your text document, and then print that document (or you could even fax it to someone).

You can also interlink the data in your spreadsheet with your bar chart and your text document. To do that you need to produce a spreadsheet with a bar or pie chart as described before. Then you copy the chart to the clipboard. In the text document you select Paste Special and Text to merge it with your text. Now you have created a link that provides your text document with the latest updates to the data in your spreadsheet. MS Windows provides a similar method for creating data links between applications, but this OS/2 function is not the same.

IBM Works database and report writer

The IBM Works database (see Figure 12-8) is a fully functional 32-bit database, but it is easy to use. You can organize text and graphical information and import from and export to the most popular database formats. The IBM Works report writer works with the IBM Works database to produce customized reports.


Database
File Edit View Record Organize Report Options Help
View Records... Add Cancel Form Data
Record 1 of 2
TEST
File Edit View Character Columns Organize Options Help
Helvetica 10 Edit
TEST.LDB
Natural

FirstName Doris
LastName Corel
Phone 6543
Address

Data Fields: FIRSTNAME LASTNAME PHONE
Report Columns: PHONE FIRSTNAME
Break Columns: PHONE Report Summary

PHONE	FIRSTNAME
6543	Doris

Figure 12-8.
Database and Report Writer

To create your database, click on the Database icon in the IBM Works folder. You will be asked if you want to create a new database or use an existing one. If you wish to use an existing one, just click on that bullet and select the database file.

If you wish to create a new database, select to open a new database. You will see a screen through which you can design your custom database. You specify field name, type, parameters, and conditions through the Insert Field window. For example, to create a simple name, phone number, and address database, you will create fields for first name, last name, phone number, and address.

Note

If you are creating fields with phone numbers, you have to decide which format you want to use for them. If you wish to write your phone numbers with nonnumerical characters such as dashes to separate them, you should use the Text field type. If you select the Number field type, you cannot separate the parts of the phone number with other characters for easier reading.

As soon as you have completed your entry for a field and clicked on **OK**, you may place that field on your database page. You will see a rectangle with a crossbar inside that you can move with your cursor to the position on the page where you want the field to reside.

After all fields have been created and placed, you are ready to enter data in your database. While creating the database layout you are in the Form mode. To switch to the Data mode, click on the symbol next to the word "Data" on the button bar beneath the pull-down menu (it is the symbol that has many *A*s on it). Then you can enter the data in your newly created database (see Figure 12-8). Use the Tab key to jump from one field to the next. Each time you complete an entry, click on **Accept** to go to the next empty data set. If you have many similar entries, click on **Copy Prev** to copy the previous data entry and just alter the fields that need to be changed. Once you have finished entering your data, press **Cancel**, which changes the entry choices on the ribbon to four arrows. These arrows allow you to move around in the database.

Now you may create a report of your database and even produce a printout. Select **Report** from the title bar and open a **New** report. This will automatically open the report writer with the data fields you have just created. Select the data fields you wish to include in your report by clicking on the field name and then clicking on the right arrow. This will add the field names in the area marked **Report Columns**. On the bottom of the report writer you can see a preview of the printout of the database.

You may change the sort order of the selected report columns by clicking on the arrows underneath the Report Columns section of the screen. The up arrow selects an ascending sort and the down arrow allows a descending sort. You will see the results of this sort immediately.

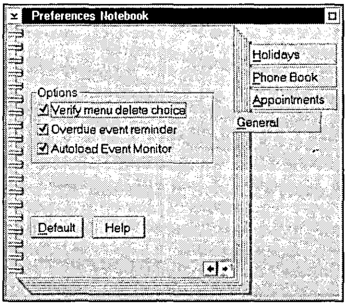
If you want to turn on the count on the entries in your database, select **Report Summary** in the Break Columns section of the screen.

Personal Information Manager (PIM)

The Personal Information Manager (PIM) is a small bundle of applications that allow you to manage your daily activities. It is your time manager, your alarm clock, your phone book, your notebook, and your to-do list. Because PIM has extensive on-line help, the following section covers these applications rather briefly.

The PIM Preferences Notebook allows you to set up your PIM for your individual needs. The General page (see Figure 12-9) allows you to **Verify menu delete choice,** which warns you with a message if you try to delete any information from your IBM Works PIM. It lets you select whether you want to see the **Overdue event reminder** when you open the IBM Works PIM. And you may also select whether you want to **Autoload the Event Monitor** whenever another IBM Works PIM program is opened (see Event Monitor section for more details about this feature).

Selected notebook pages are shown throughout this section to demonstrate some of the specific settings you can change.



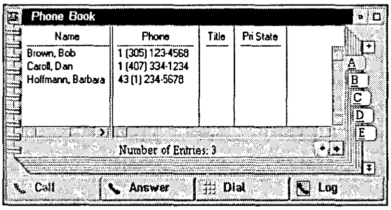
Options

☑ Verify menu delete choice
☑ Overdue event reminder
☑ Autoload Event Monitor

Figure 12-9. The PIM Preferences General Page

Phone/Address Book

You can use the PIM Phone/Address Book (see Figure 12-10) instead of your conventional phone book.

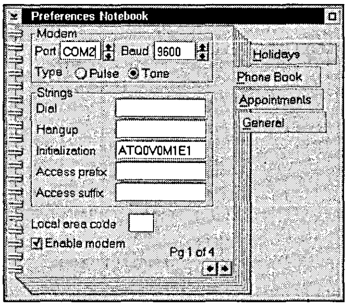


Name	Phone	Title	Pri State
Brown, Bob	1 (305) 123-4568		
Caroll, Dan	1 (407) 334-1234		
Hoffmann, Barbara	43 (1) 234-5678		

Number of Entries: 3

Figure 12-10. The Phone and Address Book

Be sure your modem is enabled in the PIM Preferences Notebook (see Figure 12-11). Enter your modem settings on the first page of the Phone Book section. Be sure to check mark the entry **Enable modem.** Other Phone Book preference settings may be changed on the other three pages of the notebook. For example, you can modify the layout of your Phone Book entry fields and choose to add columns to be displayed in the Phone Book.



Preferences Notebook

Modem

Port COM2 Baud 9600

Type ○ Pulse ● Tone

Strings

Dial

Hangup

Initialization ATQ0V0M1E1

Access prefix

Access suffix

Local area code

☑ Enable modem

Pg 1 of 4

Figure 12-11. The PIM Preferences Phone Book Page

The phone feature is very handy because the program does the dialing for you and you don't have to remember a lot of phone numbers. It also has a call log feature that allows you to make notes during a phone call and then store a record of the call. This is a useful feature if you wish to follow up your calls and refer to them at a later time, or if the person you try to call does not answer.

During the call you can keep track of the elapsed time, enter a call status, make comments in your phone log window, and if necessary, add an item to your PIM To-Do list or to your PIM Appointment Book.

Tip

Even if you do not have the appropriate hardware for making PIM phone calls, you can still take advantage of the call log feature. Assign your modem to a COM port that is not in use, for example, COM3. This way the modem will not be activated for dialing, but you can use this feature to monitor your calls with the appropriate phone numbers. For example, if you wish to keep track of your phone calls, you can use your telephone to call, but use the call log feature in OS/2 Warp to type in your comments and track the time duration of your calls.

Appointments

The Appointment Book (see Figure 12-12) allows you to organize your appointments easily. Once you have opened the Appointment Book, you will see a layout of the current week. Clicking on the day you wish to add an appointment will open a window for entering a new appointment. The Appointment Book shows you a one-week view with the current date marked with a red border.

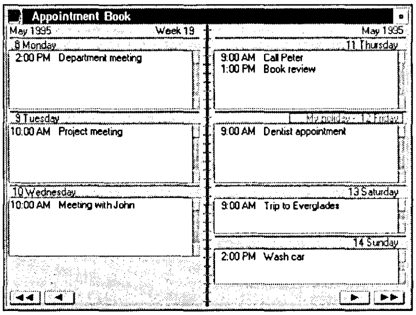


Figure 12-12. The Appointment Book

May 1995 Week 19

8 Monday
2:00 PM Department meeting

9 Tuesday
10:00 AM Project meeting

10 Wednesday
10:00 AM Meeting with John

May 1995

11 Thursday
9:00 AM Call Peter
1:00 PM Book review

My holiday - 12 Friday
9:00 AM Dentist appointment

13 Saturday
9:00 AM Trip to Everglades

14 Sunday
2:00 PM Wash car

At the bottom of the weekly view you see arrows that allow you to move forward and backward in your calendar. Clicking on any single arrow moves you either one week forward or one week backward. Clicking on any double arrow moves you either two weeks forward or two weeks backward.

The PIM Preferences Holidays page (see Figure 12-13) allows you to specify the types of holidays you will add to your Appointment Book. The holiday and the date will appear in the Appointment Book in the color you select on this page.

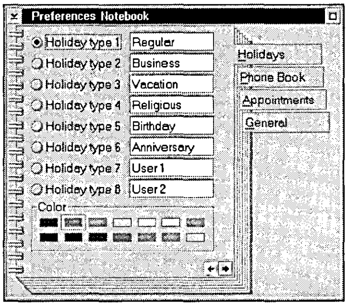


Figure 12-13. The PIM Preferences Holidays Page

● Holiday type 1 Regular
○ Holiday type 2 Business
○ Holiday type 3 Vacation
○ Holiday type 4 Religious
○ Holiday type 5 Birthday
○ Holiday type 6 Anniversary
○ Holiday type 7 User 1
○ Holiday type 8 User 2

The Appointment Book has two pop-up menus. The first menu allows you to add, change, or delete special dates on the calendar. To get to this menu, click with mouse button 2 anywhere on the gray border of the Appointment Book. You will also find this menu's functions on the year calendar, the planner, and the To-Do list. Now if you want to enter a special holiday on a certain day, go to that menu, select **Options** and then **Holidays**. Select the appropriate holiday type and add, change, or delete your entry. Under Options — Type you may select the holiday you have defined in the PIM Preferences (see Figure 12-13).

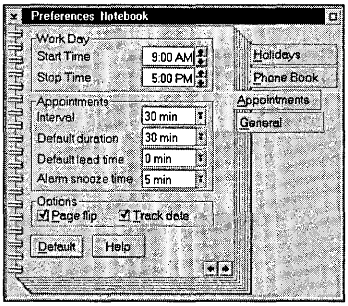
Tip

Be sure to add all necessary data in the holiday field. Once you are finished, press the **Add** button to add this holiday in the holiday list. If you forget to enter the month for example, the entry will not show up in your Appointment Book.

To bring up the second pop-up menu, position the pointer within a day's field and click on mouse button 2. You can enter a new appointment, switch to the To-Do list, or print your calendar in several formats. You can also open this pop-up menu by double-clicking on the day's field with mouse button 1.

Use the PIM Preferences Appointments page (see Figure 12-14) to enter the length of your work day and desired appointment intervals.

Figure 12-14. The PIM Preferences Appointments Page



Preferences Notebook

Work Day
Start Time 9:00 AM
Stop Time 5:00 PM

Appointments
Interval 30 min
Default duration 30 min
Default lead time 0 min
Alarm snooze time 5 min

Options
☑ Page flip ☑ Track date

Tip

You can set an alarm to remind you of an appointment shortly before its occurrence. You may even launch a particular program when the appointment time occurs. For example, you can launch a text document with the agenda and notes for a scheduled conference call.

Tip

The Recurring Appointments feature allows you to specify when a particular kind of appointment will reoccur. For example, you can schedule monthly department meetings or weekly exercise workouts with this feature as reminders without having to enter them each time.

Note

A unique feature for a personal information tool such as this one is that you may choose to either assign a certain time to an appointment or mark it just as a note. This is useful for events that have to happen on a certain day, but do not have to happen at a certain time.

You may find it surprising that the Appointment Book has no additional features in the title pull-down menu. To access other Appointment Book features, click with mouse button 2 on the frame of the Appointment Book. Here are all the other features you would expect, such as viewing the next week, viewing the previous week, go to, and other options.

You may use drag and drop to make your work easier. For example, open the Address Book and select the name of the person you have to meet. If you click on that entry with mouse button 2, you can drag it across to your Appointment Book. As soon as you move over the day field of the Appointment Book you will notice the business card icon next to your mouse cursor. Drop that card on the day you need to make the appointment. Now the New Appointment entry field pops up and the name you have selected is entered in the Title field.

Notepad

The IBM Works PIM Notepad (see Figure 12-15) is a simple, free-form notebook that allows you to store all types of information. You may enter text or insert graphic files such as bitmaps and metafiles. When adding pages, a table of contents is automatically created to help you keep an overview of the added pages. The only disadvantage is that you may not use the drag-and-drop feature between here and other IBM Works applications. You can only drag and drop within the Notepad's table of contents to rearrange documents inside.

Figure 12-15.
The Notepad

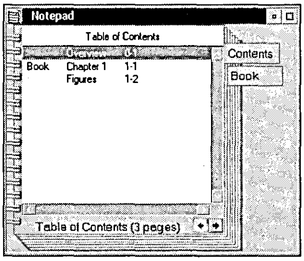


Table of Contents

	Overview	[illegible]
Book	Chapter 1	1-1
	Figures	1-2

Planner

The Planner (see Figure 12-16) allows you to view your appointments in a different way. It gives you a monthly view in a grid of time and days. You may also use the drag-and-drop feature as described in the Appointment Book section below. Online help gives you the information you need to maneuver through the Planner.

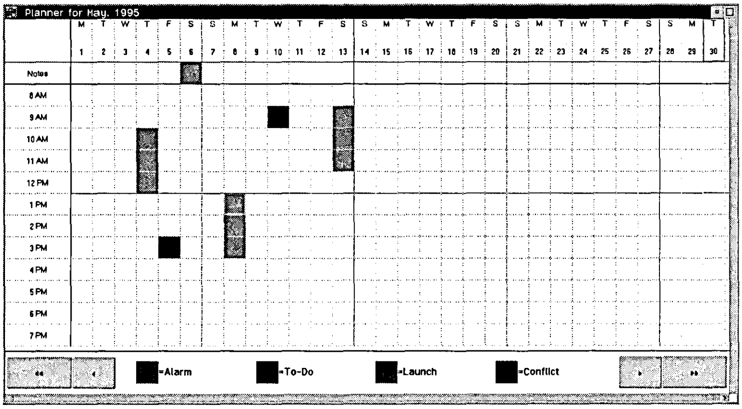
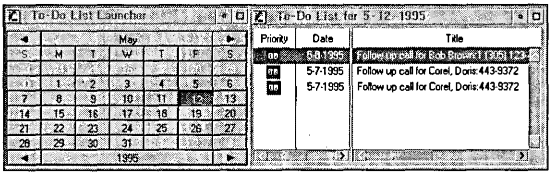


Figure 12-16. The IBM Works Planner

To-Do List

The To-Do List Launcher (see Figure 12-17) allows you to enter all your to-dos for any date. Use it either by double-clicking the date on which you need to add a to-do item, or you can drag and drop names from the Address Book onto the To-Do List Launcher.



To-Do List Launcher

◄			May			►
S	M	T	W	T	F	S
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			
◄			1995			►

To-Do List for 5-12-1995

Priority	Date	Title
	5-8-1995	Follow up call for Bob Brown:1 (305) 123-
	5-7-1995	Follow up call for Corel, Doris:443-9372
	5-7-1995	Follow up call for Corel, Doris:443-9372

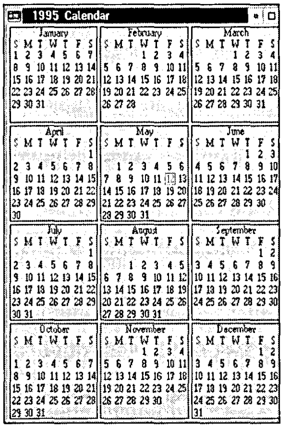
Figure 12-17. The To-Do List Launcher and the To-Do List

Year Calendar

The Year Calendar shows you the whole year (see Figure 12-18). Clicking on a date with mouse button 1 opens the To-Do List for that day. Clicking on mouse button 2 allows you to switch to other PIM functions and options. You may also print from here your yearly or monthly calendar.

Note

When you are printing from the year calendar, the utility FxPrint will be used by default for printing. If you have a fax modem attached to your PC and you wish to send a fax to someone, then you don't need to use the FxPrint utility. But if you wish to print to your local printer, you have to change the settings. These settings are not as you might expect in the calendar print function; they are included in the FxPrint queue. Select **Settings** from the pop-up menu of the FxPrint icon and switch to the printer driver and output port of your choice. Then start your print job, and as soon as the Print window appears, select your printer from the Printer Setup list.



January

S	M	T	W	T	F	S
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

February

S	M	T	W	T	F	S
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28				

March

S	M	T	W	T	F	S
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

April

S	M	T	W	T	F	S
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30						

May

S	M	T	W	T	F	S
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

June

S	M	T	W	T	F	S
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	

July

S	M	T	W	T	F	S
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					

August

S	M	T	W	T	F	S
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

September

S	M	T	W	T	F	S
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30

October

S	M	T	W	T	F	S
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

November

S	M	T	W	T	F	S
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30		

December

S	M	T	W	T	F	S
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

Figure 12-18.
The Year Calendar

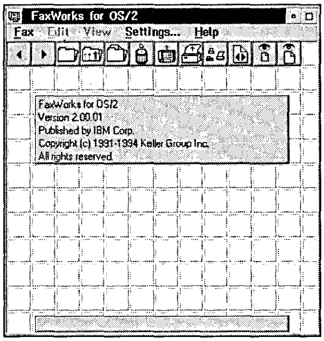
Event Monitor

The Event Monitor keeps track of your appointments and alarms and reminds you as soon as one is due or overdue.

If you've selected **Autoload Event Monitor** on the General page of the PIM Preferences Notebook, the Event Monitor will pop up and warn you about any overdue events when you start any of your PIM applications.

FaxWorks for OS/2

FaxWorks for OS/2 (see Figure 12-19) is 32-bit fax software that allows you to send and receive faxes of unlimited length (the only limit is your storage size) from your PC with an attached fax modem. It includes features such as drag and drop, printing and viewing faxes in normal and fine resolution, variable page length and portrait, as well as landscape. It uses the full power of OS/2 and the Workplace Shell to provide you with an easy-to-use WYSIWYG (what you see is what you get) interface.



FaxWorks for OS/2
Version 2.00.01
Published by IBM Corp.
Copyright (c) 1991-1994 Keller Group Inc.
All rights reserved

Figure 12-19.
FaxWorks for OS/2

A separate fax printer driver is delivered and installed along with FaxWorks for OS/2. This allows you to send faxes and documents via drag and drop even from other OS/2 applications. Most OS/2 applications support printing their document objects via drag and drop, so in those cases you may use this feature with the FaxWorks for OS/2 fax driver.

FaxWorks for OS/2 has its own address book, where you can store addresses and fax numbers of your friends or colleagues. The good news is you may access this address book from your FaxWorks for OS/2 Send Fax menu by selecting **Phone Book** next to the To field and letting the program automatically fill in the name, company, and fax number. The bad news is that there is no connection between this address book and the IBM Works Address Book. That means you have to enter at least those contacts with faxes in two different address books.

Using FaxWorks for OS/2 so that you may send faxes immediately by using drag and drop needs some setting up. Besides a modem and an analog phone line (plus a cable connecting the two), you need to configure FaxWorks for OS/2 and install the FxPrint driver:

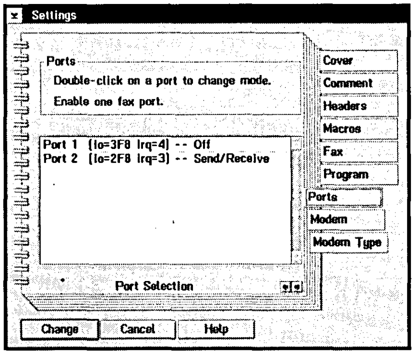
✗ To configure FaxWorks for OS/2

1. Select **Settings...** from the title bar of FaxWorks for OS/2; choose the Ports page tab, and select the port your fax is connected to. Click on that port until the Send/Receive option appears on the line (see Figure 12-20).
2. Select the **Modem** tab and enter the baud rate, speaker options, line type, and answer rings.
3. Select the **Modem Type** tab and choose your modem type. If you are not sure what type it is, check the documentation of your modem.

✗ To install the FxPrint driver (see Figure 12-21)

1. From the FxPrint pop-up menu, select **Set default printer.**
2. Open **Settings...** in the FxPrint pop-up menu, select the Printer driver tab, and select the FxPrint.Fax printer driver.
3. Select the Output tab and select the COM port your fax modem is attached to. Make sure that **Output to file** is not selected.

Figure 12-20.
Fax Port Setting for Modem



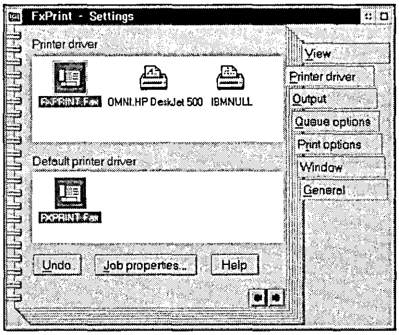
Ports

Double-click on a port to change mode.

Enable one fax port.

```
Port 1  (Io=3F8 Irq=4) -- Off
Port 2  (Io=2F8 Irq=3) -- Send/Receive
```

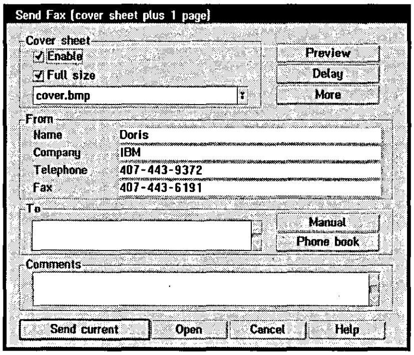
Figure 12-21.
FxPrint Pop-up Menu



Now you are ready to send your faxes. You can drag and drop the document you wish to send on the FxPrint icon. A window will pop up asking if the file you are sending is plain text or printer specific. Select the appropriate option and be patient. For a few seconds you will see nothing, although a lot is happening in the background.

Depending on the amount of memory and the swapper space you have, it will take a moment or two. You will see the LED of your hard disk flashing, which indicates that your print job is being prepared and FaxWorks for OS/2 has started. The Send Fax window will then pop up (see Figure 12-22). This is where you create your fax cover sheet.

Figure 12-22. The Send Fax Dialog Box


Send Fax (cover sheet plus 1 page)

Cover sheet
☑ Enable
☑ Full size
cover.bmp

From
Name Doris
Company IBM
Telephone 407-443-9372
Fax 407-443-6191

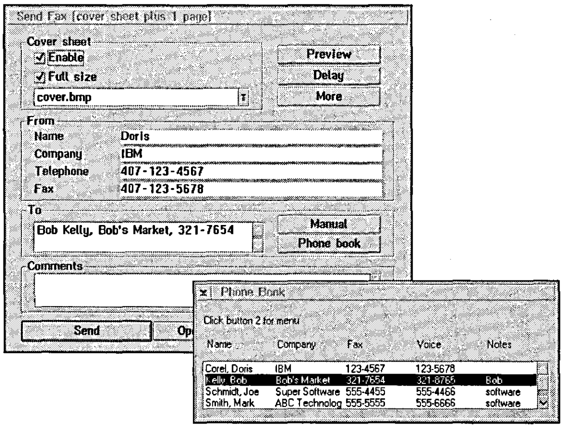
To

Comments

If you have selected the default entry on the Cover page of the FaxWorks Settings notebook, you will see your name, company, phone, and fax number in the fields under From. If these fields are empty, then you should now enter them in the default settings. In the To field, enter the name, company, and fax number of the person to whom you wish to send the fax. You can input this information manually or you can use the Phone Book.

Entering phone numbers is one of the tasks I like to do the least and I want to spare you from it as well. I recommend that you select **Phone book** from this screen, even if you know the fax recipient is not in the Phone Book. You can add new Phone Book entries at this point, so you won't have to type in the names and numbers again. To do this, position your mouse pointer inside the Phone Book (see Figure 12-23) and click mouse button 2. The menu that pops up allows you handle your Phone Book entries. To add a new entry select New and fill in the fields of the Phone Book Edit window.

Figure 12-23. Editing the Phone Book


Send Fax (cover sheet plus 1 page)

Cover sheet
☑ Enable
☑ Full size
cover.bmp

From
Name Doris
Company IBM
Telephone 407-123-4567
Fax 407-123-5678

To
Bob Kelly, Bob's Market, 321-7654

Comments

Phone Book

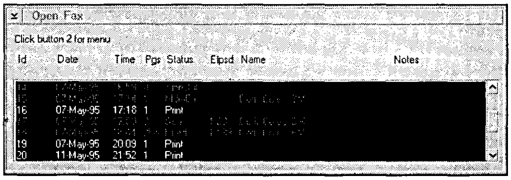
Click button 2 for menu

Name	Company	Fax	Voice	Notes
Corel, Doris	IBM	123-4567	123-5678	
Kelly, Bob	Bob's Market	321-7654	321-8765	Bob
Schmidt, Joe	Super Software	555-4455	555-4466	software
Smith, Mark	ABC Technolog	555-5555	555-6666	software

After you've finished your entry, you will be back in the Phone Book, where you may select the person to whom you want to send the fax. Double-click on the appropriate entry and watch the result. You will be back at the Send Fax window and the To field should be filled out with the information for the individual you just added to the Phone Book. You may add a comment regarding the fax before you send it. As soon as you press **Send**, a little window appears showing you the status of the transmission.

If the transmission is not successful, a message in that window will tell you so. If it is successful, you will see a record of transmission in the FaxWorks for OS/2 fax log (see Figure 12-24). This log keeps track of all incoming and outgoing faxes by the status Sent. If you click on mouse button two while the pointer is inside the list (and not on any window frame), you will see a menu with several options to print, re-send, and delete faxes from the list or to edit the log entry.

Figure 12-24.
Fax Log



Click button 2 for menu

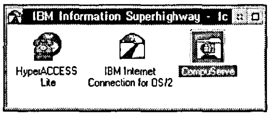
Id	Date	Time	Pgs	Status	Elpsd	Name	Notes
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]			
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]		[illegible]	
16	07-May-95	17:18	1	Print			
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	
19	07-May-95	20:09	1	Print			
20	11-May-95	21:52	1	Print			

The Information Superhighway

The OS/2 Warp BonusPak is shipped with a bundle of applications called the Information Superhighway (see Figure 12-25). The bundle consists of three products:

- ✗ IBM Internet Connection for OS/2
- ✗ CompuServe Information Manager
- ✗ HyperACCESS Lite

Figure 12-25.
The Information Superhighway Applications


IBM Information Superhighway - Ic
HyperACCESS Lite
IBM Internet Connection for OS/2
CompuServe

What is this Information Superhighway?

Advances in telecommunications are changing the way we think about communicating with people. Telephones are everywhere and their use is taken for granted. But according to John Naisbitt, author of *Global Paradox* (Avon Books, 1994), a telecommunication revolution is coming that does require a global telecommunications network of networks to allow everyone in the world to be connected to everyone else.

Although telephone communication offers voice links, it is often not enough to just talk to someone. Frequently you need access to information someone may have spent a great deal of time researching. The Information Superhighway gives you access to a variety of information resources. For example, you might want to find out what books have been written about the Internet or you might need the flight schedule for your favorite airline to book a well-deserved summer vacation. You can do all that on the Information Superhighway.

"But I can do that with my phone," you say. Yes, you can, but what if you need information from someone across the country (or the globe)? Time zone differences often make information retrieval more complicated than it seems. You can also use the Information Superhighway to write letters or to respond to a voice or fax message.

IBM Internet Connection

One of the most popular sources on the Information Superhighway is called the Internet. IBM has produced a 32-bit OS/2 application that gives you access to Internet.

What is the Internet Connection?

The IBM Internet Connection (also referred to as the Internet Access Kit) is the first OS/2 version of an application that allows you to access the Internet. The Internet Connection is a commercial service that taps into the Internet, giving you access to the networking resources (that is, the technology and expertise) of many people and companies worldwide. These resources are owned by companies of all sizes, by universities and colleges, and even by individuals.

Note

History of the Internet. The Internet was originated as a U.S. Defense Department network called ARPAnet. Over time, the network was extended to universities and research centers, and now, commercial networks. The term *Internet* means interconnected networks. It consists of thousands of individual networks with various routers, bridges, and other connections — all of which give the end user the impression that the Internet is one big network. The Internet is not owned or ruled by one company, but rather it is supervised by a council of volunteers. Currently, there are over 2 million computers (mainframes, UNIX systems, and PCs) tapped in, with over 20 million users.

What does the IBM Internet Connection include?

The IBM Internet Connection is a complete set of access and navigation tools designed to simplify and enhance your interaction with the Internet. The following is a description of these tools:

- ✗ **IBM Internet Dialer** — This is a tool that provides you with local dial access for low-cost worldwide connections in many areas.
- ✗ **Ultimedia Mail/2 'Lite'** — This is your Electronic Mail (e-mail) tool that allows you to send mail around the globe and is connected to mailbox servers that hold incoming mail when you're not connected.
- ✗ **NewsReader/2** — This tool connects you to a news server that taps you into discussion groups on your favorite subjects.
- ✗ **WebExplorer and Gopher** — These are tools that allow you to connect to servers filled with information. With these tools, you may begin exploring the Internet with practical interfaces.
- ✗ **IBM Internet Customer Services** — A customer support that's there for you when you need it for answering questions about the Internet, about Advantis, about registration to the Internet. It also provides information about your account.

The advantage of OS/2's multitasking environment is that you can perform several tasks at the same time. In terms of the IBM Internet Connection, you can search for a list of articles on NewsReader/2, start the WebExplorer searching for information, and while both of those things are happening, you can go to your Internet Dialer and download the latest software updates for your Internet phone list for accessing the Internet. This way you save logon time and money.

What do I need for the IBM Internet Connection to work?

If you want to run the IBM Internet Connection, I recommend you use the fastest modem possible. You'll need at least 9600 baud; 14,400 or 28,800 baud would be better — it all depends on how much money you are willing or able to spend.

Tip

When you configure your modem under the IBM Internet Connection, you are asked to select the value for the UART (Universal Asynchronous Receiver/Transmitter). The online help recommends that you set the UART value higher than the actual baud rate of your modem. For example, if you have a 9600 baud modem, I recommend you select a UART value of 14,400. For V.32, V.32bis, or V.34 modems, I recommend that you use a UART speed that is four times the baud rate of your modem.

Developer's Desktop

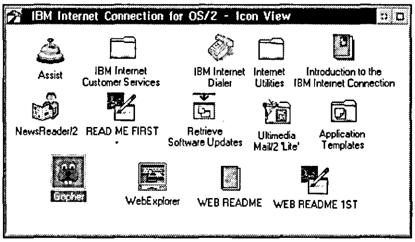
I use an internal IBM modem for a ThinkPad 700C, IBM (PNB) 9600, with a baud rate of 9600. I use the default settings provided with the IBM Internet Dialer, and the COM port I selected is COM2.

How do I get connected?

Before you can start "surfing the net" you need the following: a user ID, a user account, a credit card for billing, and the curiosity and time to explore a vast amount of information.

After you have installed the Internet Connection, you will find an Information Superhighway folder in your BonusPak folder. Inside that folder you will find a folder that contains the IBM Internet Connection for OS/2. Select the folder for the IBM Internet Connection. You will see several programs and a few more folders with Internet utilities (see Figure 12-26).

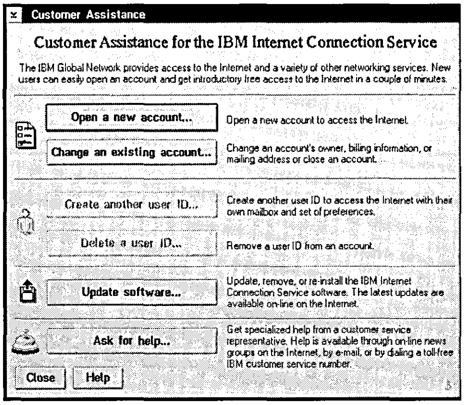
Figure 12-26. The IBM Internet Connection Folder



The first thing you need to do after installing the software is to register in order to get your Internet user ID. If you want to access the IBM Internet Connection you need a service provider, which is the administrative interface between your workstation and the outside information world. You may choose from various service providers, but the IBM Internet Connection already offers you the IBM service provider Advantis, so you don't have to look for one. If you already have a service provider, for example Delphi, you might as well keep using it.

With Advantis, you may connect to the Internet even if you don't have a service provider yet. Open the Internet Customer Services folder and click on the Registration object. To register, select **Open a new account...** (see Figure 12-27), which entitles you to a free trial period.

Figure 12-27. Customer Assistance Dialog Box



Customer Assistance

Customer Assistance for the IBM Internet Connection Service

The IBM Global Network provides access to the Internet and a variety of other networking services. New users can easily open an account and get introductory free access to the Internet in a couple of minutes.

Open a new account... — Open a new account to access the Internet.

Change an existing account... — Change an account's owner, billing information, or mailing address or close an account.

Create another user ID... — Create another user ID to access the Internet with their own mailbox and set of preferences.

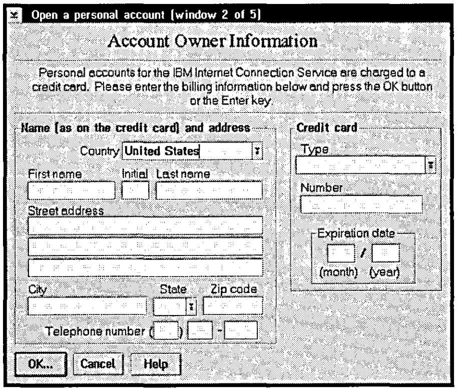
Delete a user ID... — Remove a user ID from an account.

Update software... — Update, remove, or re-install the IBM Internet Connection Service software. The latest updates are available on-line on the Internet.

Ask for help... — Get specialized help from a customer service representative. Help is available through on-line news groups on the Internet, by e-mail, or by dialing a toll-free IBM customer service number.

You will see a screen informing you about the terms, conditions, and charges for using the service provider. Follow the instructions for filling in the needed information (see Figure 12-28). When you are finished, you will be given an account ID, a user ID, and a temporary password. You will also get your Internet mail address, which consists of your user ID, the @ sign, and the address. For example, if your user ID is Robin1 and your address is mit.com, then your e-mail address is Robin1@mit.com (Internet mail addresses are not case sensitive).

Figure 12-28. Advantis Account Information Dialog Box



Open a personal account (window 2 of 5)

Account Owner Information

Personal accounts for the IBM Internet Connection Service are charged to a credit card. Please enter the billing information below and press the OK button or the Enter key.

Name (as on the credit card) and address

Country: United States

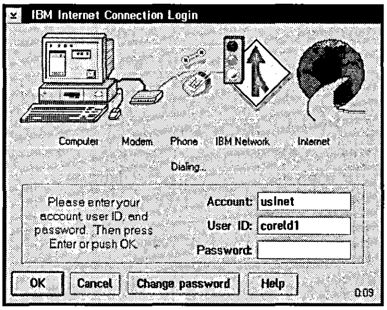
Note

You can allow others (for example, your spouse) to get a user ID that is paid for by your credit card. You have to be logged onto the Internet to do this. After you've registered yourself, click on the Customer Assistance icon in the IBM Internet Customer Services folder or the **Assist** button in the IBM Internet Connection for OS/2 and select **Open a new account....** If you select **Create another user ID...** you have to specify the name of the user who will have access to your account.

The IBM Internet Connection comes with two programs, NewsReader/2 and Gopher, which give you access to the information resources available on the Internet. With OS/2 Warp with WIN-OS/2, you also get the WebExplorer.

To access any of these applications you need to log on to the Internet first. You do this via the IBM Internet Dialer. Click on the icon with the blue telephone. You will see the IBM Internet Dialer window. Select **Dial** to get to the IBM Internet Connection Login dialog box (see Figure 12-29). Follow the login instructions that appear in the dialog box.

Figure 12-29.
IBM Internet Connection Login Dialog Box



IBM Internet Connection Login

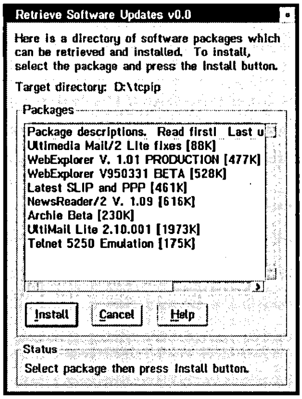
Dialing..

Please enter your account, user ID, and password. Then press Enter or push OK.

Account: usInet
User ID: coreld1
Password:

Once you have installed the software and logged on to the Internet, the versions of your Dialer & Customer Assistance programs, the Phone list, and the Modem list are checked for their dates. If the program detects an older date installed on your system, you will see a dialog box that allows you to retrieve software updates (see Figure 12-30).

You do not have to download the updated software at this point, even though it looks at first glance as though you have no choice. If you wish to ignore the download, just click on the pull-down menu button of the window title in the upper left corner and select **Close.** Then press **Cancel** at the Download latest software window.



Retrieve Software Updates v0.0

Here is a directory of software packages which can be retrieved and installed. To install, select the package and press the Install button.

Target directory: D:\tcpip

Packages

Package descriptions. Read first! Last u
Ultimedia Mail/2 Lite fixes [88K]
WebExplorer V. 1.01 PRODUCTION [477K]
WebExplorer V950331 BETA [528K]
Latest SLIP and PPP [461K]
NewsReader/2 V. 1.09 [616K]
Archie Beta [230K]
UltiMail Lite 2.10.001 [1973K]
Telnet 5250 Emulation [175K]

Status

Select package then press Install button.

Figure 12-30. Internet Dialer Software Update Dialog Box

How do I get to the good stuff?

Having access to the Internet is one thing, but how do you get to the good stuff? Besides sending letters around the world with Ultimedia Mail/2 'Lite', you get three programs to retrieve information. That's where the WebExplorer, NewsReader/2, and Gopher come in.

These programs give you access to the Information Superhighway via mouse clicks instead of typing and knowing domain names or IP addresses and the names of the programs you're looking for. The following sections describe what these programs are and how they work.

Ultimedia Mail/2 'Lite'. Ultimedia Mail/2 'Lite' (see Figure 12-31) is your e-mail tool that allows you to communicate with people who are connected to the Internet. The first time you log on with your new user ID, you will receive your first letter in the In-basket of the Ultimedia Mail/2 'Lite' folder. The In-basket is your mailbox, which collects all your mail.

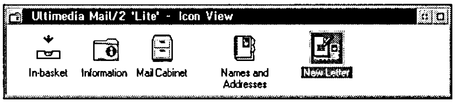


Figure 12-31. The Ultimedia Mail/2 'Lite' Folder

The Information folder contains a tutorial, which I recommend you try out to familiarize yourself with the Ultimedia Mail/2 'Lite' functions. Otherwise, you can jump right in.

Once you have received mail, store it in your mail cabinet to clear out the In-basket. Click on the New Letter object in the Ultimedia Mail/2 'Lite' folder and send yourself a short letter to practice with that tool. Besides sending text, you can even include pictures, such as OS/2 bitmaps or GIF files, and binary data such as a spreadsheet or an image file of a compressed diskette (for example, a file compressed with the tool PKZIP).

Note

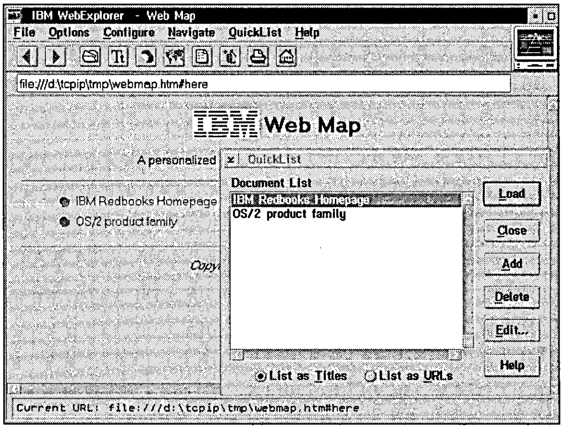
Sending a file with even a compressed 1.44MB diskette may take 30 minutes or more. Long connect times will be reflected in your telephone bill!

You can import data files either by dragging the file onto the letter field or selecting **Import** from the letter field's pop-up menu. If you want to save the currently active letter part to a file, use the **Export** option on the pop-up menu.

The Names and Addresses notebook helps you organize contact information for your Internet post office. Storing this information here and grouping it with the "group" function allows you to find your business and private addresses more easily.

The World Wide Web and the IBM WebExplorer. The World Wide Web is like an enormous electronic library. You can retrieve information from the World Wide Web using the graphical interface that is provided by the IBM WebExplorer (see Figure 12-32). Instead of using commands and network addresses, you just click on icons or text. The graphical interface also allows viewing of graphic images and even sound.

Figure 12-32.
IBM WebExplorer



file:///d:\tcpip\tmp\webmap.htm#here

A personalized

- IBM Redbooks Homepage
- OS/2 product family

Copy

Document List

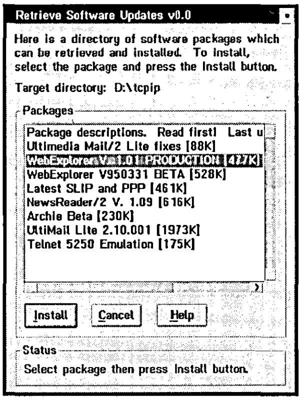
IBM Redbooks Homepage
OS/2 product family

◉ List as Titles ○ List as URLs

Current URL: file:///d:\tcpip\tmp\webmap.htm#here

Shhh!
Tip

If you have OS/2 Warp (that is, the version without WIN-OS/2), you can install the WebExplorer through the Internet by downloading it (see Figure 12-33). This is the same program that is shipped with OS/2 Warp with WIN-OS/2.



Retrieve Software Updates v0.0

Here is a directory of software packages which can be retrieved and installed. To install, select the package and press the Install button.

Target directory: D:\tcpip

Packages

Package descriptions. Read first! Last u
Ultimedia Mail/2 Lite fixes [88K]
WebExplorer V. 1.01 PRODUCTION [477K]
WebExplorer V950331 BETA [528K]
Latest SLIP and PPP [461K]
NewsReader/2 V. 1.09 [616K]
Archie Beta [230K]
UltiMail Lite 2.10.001 [1973K]
Telnet 5250 Emulation [175K]

Status

Select package then press Install button.

Figure 12-33. Retrieve Software Updates

You usually get connected automatically to the homepage of your service provider. This is an excellent way to help novice users get started. A homepage is basically the main menu for a specific topic. You start your search for information through the homepage. You may choose your startup page by first saving interesting addresses in the so-called QuickList. Whenever you find an interesting site, select **Add current document** from the QuickList pull-down menu or click on the icon with the thumb-up plus sign to add it to the QuickList. This helps you to navigate through Web documents and you will always be able to access the documents by selecting them from the history list in the Web Map.

The location addresses for these documents are called URLs (Uniform Resource Locators), which identify the documents by their location on a server or host in a subdirectory. The address includes the protocol followed by the fully qualified host name and may include the path and file name. If you store your favorite places in the QuickList, you'll save yourself from typing the URL every time you want to go to that location.

For example, an URL may look like this:

```
http://www01.ny.us.ibm.net/
```

Developer's Desktop

I use Advantis as my service provider. Thus, I am getting the IBM homepage, which offers lots of information and even lets me see and hear IBM's General Manager Lou Gerstner. I use the version of the WebExplorer that you have to download from the Internet for OS/2 Warp.

All settings of the WebExplorer are stored in a file called EXPLORE.INI. You can find that file in the \TCPIP\ETC directory. Unlike all other OS/2 .INI files, you can edit this one with your favorite text editor. This is useful if you wish to exchange your settings of your WebExplorer with other users. You will find it especially practical to exchange your list of favorite places, that is, the places you'd save in your QuickList.

Warning

Be careful when you edit this file. Save your original EXPLORE.INI first before you modify it. Be sure you know exactly what you want to modify within that file and how to apply the changes.

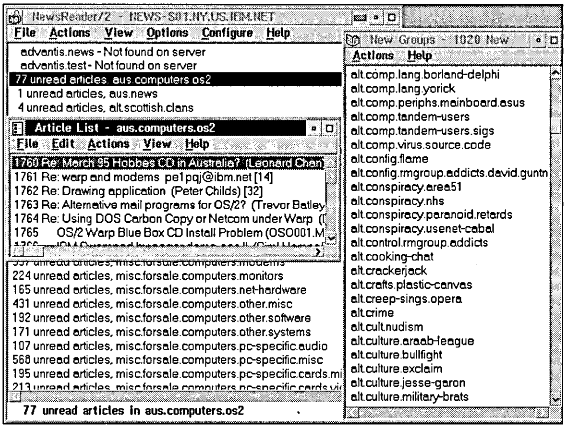
The following is an example of an EXPLORE.INI:

```
; Web Explorer INI file
; (edit this file with care)

[screen]
xleft=4
ybottom=0
width=640
height=480
fontfamily=Helvetica
fontsize=Normal
textcolor=black
linkcolor=blue
visitcolor=darkpink
backcolor=palegray

[cache]
CacheOn=Yes
CacheMem=Yes
cachedocs=16
cacheimages=32

[options]
SaveToDisk=No
InlineImages=Yes
UnderlineLinks=No
InternalViewer=Yes
CustomAnimations=Yes
ShowURL=Yes

[network]
HomePage=http://www01.ny.us.ibm.net
AutoLoad=Yes
Email=coreld1@ibm.net
Proxy=
News=news-s01.ny.us.ibm.net
Socks=
[advanced]
; advanced user settings - edit with care!
;
; mailcap= specifies full path to user mailcap file
; format is: mime/type; program_name params %s
; example: image/jpeg; jview -r %s
; no wildcards allowed, no piping, no unix commands
mailcap=
; extmap= specifies full path to user extension map file
; format is: mime/type extension_list
; example: image/jpeg jpg jpeg jpe
extmap=

[viewers]
; DO NOT edit this section
; data tags (e.g. viewer=gif) do not represent file extensions
; use mailcap in advanced section above to add NEW mime types
viewer=editor, e.exe
viewer=gif, ib.exe
viewer=jpeg, ib.exe
viewer=tiff, ib.exe
viewer=bmp, ib.exe
viewer=xbitmap, ib.exe
viewer=mpeg, vb.exe
viewer=quicktime, vb.exe
viewer=avi, vb.exe
viewer=avs, vb.exe
viewer=au, ab.exe
viewer=aif, ab.exe
viewer=wav, ab.exe
viewer=inf, view.exe

[quicklist]
quicklist= IBM Redbooks Homepage
http://www-i.almaden.ibm.com/redbooks/itsoord.html
quicklist= OS/2 product family
http://www.austin.ibm.com/pspinfo/os2.html
quicklist= Web Wide World of Music
http://american.recordings.com/wwwofmusic
quicklist= IBM Internet Connection Support
http://www.ibm.net/support/menu01d.html
quicklist= IBM World Wide Web Server
http://www.ibm.com
quicklist= IBM Personal Software Products
http://www.austin.ibm.com/pspinfo
```

NewsReader/2. NewsReader/2 provides you with access to USENET news groups (see Figure 12-34). You can get an alphabetical list of news groups if you select the NewsReader/2 icon. As with a newspaper, you may scan the groups (or headlines), select the ones you are interested in, and then read them. The first time you log on to

NewsReader/2, the list of news groups is created. Each time you connect to NewsReader/2 after that, you will be asked if you wish to update your news list if it is over a week old.

Figure 12-34. NewsReader/2 and Article List


NewsReader/2 - NEWS-S01.NY.US.IBM.NET
File Actions View Options Configure Help
Article List - aus.computers.os2
File Edit Actions View Help
New Groups - 1020 New
Actions Help

```
advantis.news - Not found on server
advantis.test - Not found on server
77 unread articles. aus.computers.os2
1 unread articles, aus.news
4 unread articles, alt.scottish.clans
```

```
1760 Re: March 95 Hobbes CD in Australia?  (Leonard Chan)
1761 Re: warp and modems  pe1pqj@ibm.net [14]
1762 Re: Drawing application  (Peter Childs) [32]
1763 Re: Alternative mail programs for OS/2?  (Trevor Batley
1764 Re: Using DOS Carbon Copy or Netcom under Warp  ([
1765     OS/2 Warp Blue Box CD Install Problem (OSO001.M
```

```
224 unread articles, misc.forsale.computers.monitors
165 unread articles, misc.forsale.computers.net-hardware
431 unread articles, misc.forsale.computers.other.misc
192 unread articles, misc.forsale.computers.other.software
171 unread articles, misc.forsale.computers.other.systems
107 unread articles, misc.forsale.computers.pc-specific.audio
568 unread articles, misc.forsale.computers.pc-specific.misc
195 unread articles, misc.forsale.computers.pc-specific.cards.mi
77 unread articles in aus.computers.os2
```

```
alt.comp.lang.borland-delphi
alt.comp.lang.yorick
alt.comp.periphs.mainboard.asus
alt.comp.tandem-users
alt.comp.tandem-users.sigs
alt.comp.virus.source.code
alt.config.flame
alt.config.rmgroup.addicts.david.guntn
alt.conspiracy.area51
alt.conspiracy.nhs
alt.conspiracy.paranoid.retards
alt.conspiracy.usenet-cabal
alt.control.rmgroup.addicts
alt.cooking-chat
alt.crackerjack
alt.crafts.plastic-canvas
alt.creep-sings.opera
alt.crime
alt.cult.nudism
alt.culture.araab-league
alt.culture.bullfight
alt.culture.exclaim
alt.culture.jesse-garon
alt.culture.military-brats
```

If you double-click on the NewsReader/2 icon and you have not yet started the IBM Internet Dialer, a window asks whether you want to connect right away or later and asks which dialer you want to use. If you select **Connect,** then the IBM Internet Dialer is started.

Once you are connected to NewsReader/2, you may select the articles you are interested in. The first list shows you the number of unread articles in each of the newsgroups available on the server you are connected to. Click on the newsgroup you are interested in and you will see an Article List window, with the titles of all articles in that newsgroup.

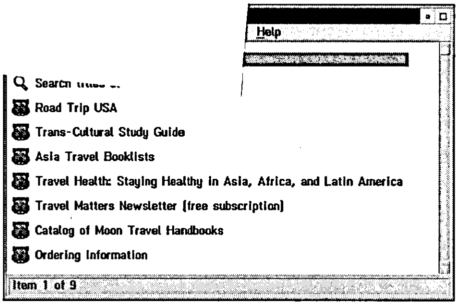
To give you an example of what such an address looks like, my news server is NEWS-S01.NY.US.IBM.NET.

If you wish to read a particular article, select **File** from the menu bar and then **Save** the article to your local disk to read it after you have logged off. This is especially useful for large or multiple documents. As you select articles to save, mark them as **Read or Saved** and proceed with your scan. After you have saved the articles you want to read, disconnect from the dialer and then you can read them offline through your favorite text editor.

IBM web explorer
Postmasters. raleigh. ibm. com
ANON
CD web-ex
WebXV11f.exe (Binary)
\TCPIP\ Help

›rovides a user-friendly interface to accessing nountain of information with an search for specific information with the e a gopher — through the information. st to you, you may read it.

List



Search ...
Road Trip USA
Trans-Cultural Study Guide
Asia Travel Booklists
Travel Health: Staying Healthy in Asia, Africa, and Latin America
Travel Matters Newsletter [free subscription]
Catalog of Moon Travel Handbooks
Ordering Information

Item 1 of 9

To minimize your connect time to the Internet, you may download documents and files just as you do it in the NewsReader/2. Select the document or file you wish to save. From the pull-down menu, choose **Selected** and then **Get**. You will see a window that allows you to name and save your file. You may only select one item to save at a time.

You can use "bookmarks" to keep track of your favorite places found through the Gopher. Select the topic you wish to read about by clicking on it once and then select **Gopher.** From the pull-down menu, select **Bookmark this menu** when you want to remember a site. From the same pull-down menu, select **Open bookmark window.** You will find all the sites you have ever saved as bookmarks, which allows you to quickly go to your favorite places. You may even select a new homepage, which will be presented to you as soon as you log on to the Gopher.

CompuServe Information Manager for OS/2

The CompuServe Information Manager (CIM-OS/2) is the first OS/2 version, and it runs in the OS/2 Presentation Manager environment, which makes CompuServe easy to learn and use (see Figure 12-36).

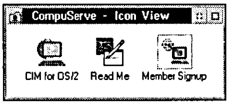


Figure 12-36.
CompuServe Services Folder

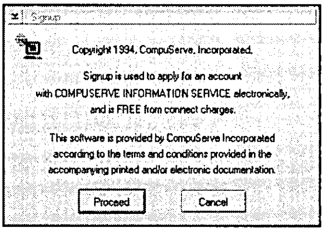
The object-oriented CIM-OS/2 allows you to navigate quickly and get easy access to the on-line information and services. It provides this access through reliable data transmission and uses CompuServe B+ Protocol for error detection and correction for all data transmissions. CIM-OS/2 offers several features that allow you to perform many tasks:

- The local message creation feature and an integrated filing cabinet help you save money on connect time. You can create messages and gather files to send to other CompuServe users offline, so you only pay the connect time for the actual transmission of your data. Similarly, you can retrieve messages and save them to your local storage, so you can read them and respond locally while you're offline.
- The integrated local address book allows you to maintain a list of your CompuServe friends and associates.
- The language and terms used within CIM-OS/2 are similar to the ones you would use on your desk with your papers, such as your In- and Out-basket functions.
- The Favorite Places function allows you to store information about the places you've found through CompuServe that you return to frequently.
- Message Maps provide visual representation of the flow of messages in a forum topic, for easy followup of forum entries.
- An Electronic Conferencing feature allows you to participate in more than one conversation at the same time.
- Graphic Interchange Format (GIF) and JPEG Support is provided as well. You can view GIF and JPEG images online or offline.
- CIM-OS/2 allows you also to make resumable downloads in both the desktop and terminal emulation modes.
- A stock portfolio option enables members to create and maintain a record of stock prices.

How do I start my CIM-OS/2 connection?

After installing CIM-OS/2, you will find a CompuServe folder in the Information Superhighway folder. The first thing you have to do is get registered, which you do by clicking on the Member Signup icon in the CompuServe folder. You will then see a Member Signup screen (see Figure 12-37).

Figure 12-37. CompuServe Member Sign Up Screen



Copyright 1994, CompuServe, Incorporated.

Signup is used to apply for an account with COMPUSERVE INFORMATION SERVICE electronically, and is FREE from connect charges.

This software is provided by CompuServe Incorporated according to the terms and conditions provided in the accompanying printed and/or electronic documentation.

What do I need for CIM-OS/2 to work?

As with the Internet Connection, you need to install the software and make sure your modem is attached correctly. Then you also need a user ID, a credit card, and a telephone number. Once you are registered, you may log in with your new user ID and password.

Note

The first thing you should do after you log in for the first time is change your password. Click on the **Services** pull-down menu of the title bar and select **Go**, or just click on the green traffic light icon on the bar beneath the pull-down menu. Type **password** in the entry field. You will get prompted to enter your old password once and then your new password twice. If you don't change your original password, after some time it will be revoked and you will have to sign up again.

You are now ready to explore this part of the Information Superhighway. While you are passing interesting sites on your journey, you should save your favorite sites in your Favorite Places list. You can do that easily by selecting the forum you wish to keep, opening the window with your favorite places and selecting **Add**.

Shhh! Tip

Following is a list of some interesting places to visit on CompuServe:

- ✘ IBM Online + CIS:IBM
- ✘ IBM Corporate Forums CIS:IBMCORP
- ✘ IBM OS/2 Users Forum + CIS:OS2USER

- IBM OS/2 Support Forum + CIS:OS2SUPPORT
- IBM OS/2 Forums CIS:IBMOS2
- IBM PCDOS (PSP Products) Forum + CIS:PCDOS
- IBM OS/2 Vendors Forums + CIS:OS2VEN
- IBM Personal Computer Forums + CIS:IBMPCFORUMS
- IBM Ultimedia Forums CIS:IBMULTI
- CIM for OS/2 Support Forum (FREE) CIS:OCIMSUP
- Practice Forum (FREE) CIS:PRACTICE
- Order from CompuServe (FREE) CIS:ORDER
- CompuServe Help Forum (FREE) CIS:HELPFORUM

CIM-OS/2 allows you to set up your workstation with several logon profiles. For example, you may need different profiles for working at home and at the office, if you dial in through different phone numbers from each location.

Click on **Special** from the pull-down menu and select **Setup Session Settings.** You can set the CIM defaults for your logon profile, your modem type, and your COM port. You will need to create a new session profile if you want to have different settings for different setups (for example, one for working on your office docking station and one for working offline). Your default settings are stored in CIM-OS/2 in the \CSERVE\OS2-CIM\CIM.INI file.

Warning

Be sure not to erase or modify the CIM.INI file in such a way that you may not use it anymore and that you have to re-install CIM-OS/2. To be on the safe side, make a backup copy of that file.

Developer's Desktop

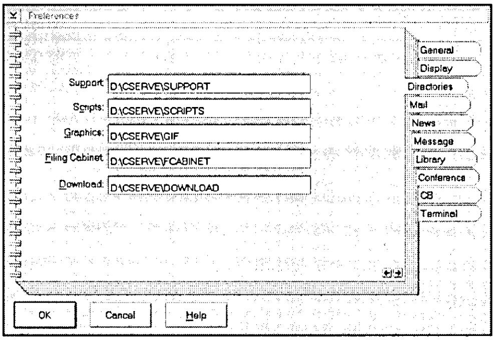
Following is my own CIM.INI file. I have two different logon settings, one for my Thinkpad when I'm "on the road" and one for my ThinkPad when it's in a docking station. The on-the-road setting has the modem attached to the COM1 port and the docking station setting has it attached to COM2.

```
[Connections]
CIS Connection=CIS Connection
Docking Station=Docking Station
Home=Home
```

```
[CIS Connection]
Alternate=[None]
User=User (CIS Connection)
Connector=Connector (CIS Connection)
DialType=Tone
Modem=Modem (CIS Connection)
Network-1=Primary Network (CIS Connection)

[User (CIS Connection)]
UserName=Doris Corel
UserID=99999,9999
Password=Wouldn't you like to know?
LogonParams=
HMITimeout=60

[Connector (CIS Connection)]
PortID=COM1:
CommType=Local
CheckCD=ON

[Primary Network (CIS Connection)]
Name=
Rate=9600
Phone-1=9,1,305-426-0228
Retries=2

[Modem (CIS Connection)]
SetName=
Prefix=AT
Suffix=^M
Init=ATS0=0 Q0 V1 &C1&D2^M
Reset=&F
Tone=DT
Pulse=DP
Escape=+++
Hangup=H0
Ack=OK
Success=CONNECT
Failure=NO CARRIER
DCOn=&q5
UseDC=ON
ECOn=&Q5S46=0
UseEC=ON
Speaker=ON

[Docking Station]
Alternate=[None]
User=User (Docking Station)
Connector=Connector (Docking Station)
DialType=Tone
Modem=Modem (Docking Station)
Network-1=Primary Network (Docking Station)
```

```
[User (Docking Station)]
UserName=Doris Corel
UserID=99999,9999
Password=It is secret here also
LogonParams=
HMITimeout=60

[Connector (Docking Station)]
PortID=COM2:
CommType=Local
CheckCD=ON

[Primary Network (Docking Station)]
Name=
Rate=9600
Phone-1=9,1,305-426-0228
Retries=2

[Home]
Alternate=[None]
User=User (Home)
Connector=Connector (Home)
DialType=Tone
Modem=Modem (Home)
Network-1=Primary Network (Home)

[User (Home)]
UserName=Doris Corel
UserID=99999,9999
Password=Here it is again
LogonParams=
HMITimeout=60

[Connector (Home)]
PortID=COM2:
CommType=Local
CheckCD=ON

[Primary Network (Home)]
Name=
Rate=9600
Phone-1=1,305-426-0228
Retries=2
```

The IBM CIM stores its default paths for the subdirectories it needs in a Preferences notebook, which you can access through the Special menu from the CIM window (see Figure 12-38).



Support: D:\CSERVE\SUPPORT

Scripts: D:\CSERVE\SCRIPTS

Graphics: D:\CSERVE\GIF

Filing Cabinet: D:\CSERVE\FCABINET

Download: D:\CSERVE\DOWNLOAD

Figure 12-38. CIM-OS/2 Preferences Notebook

If you select the Directories tab on the Preferences notebook, you will see the list of paths entered during the installation program. Altering the drive in any of these input fields on which CIM-OS2 was installed has a vital impact on these paths. When you start to connect to the CIM-OS/2 you will see an error message indicating that CIM "...could not open a script file." There is no indication of which script file or why CIM-OS/2 could not find it, because the path changed when you altered it in the above-mentioned input fields.

Also, if you have a docking station with a hard drive installed on it, you will experience, as soon as you slide your ThinkPad into its docked position and boot it, that all the paths are different because you now have two hard disks. Your drive D: on the ThinkPad becomes E: in the docking station because drive D: is already taken by the first partition of the hard disk in the docking station.

To solve this problem you may do the following

1 Every time you remove or insert the ThinkPad, go to the CIM-OS/2 Settings and change the path under Preferences on the Directories tab.

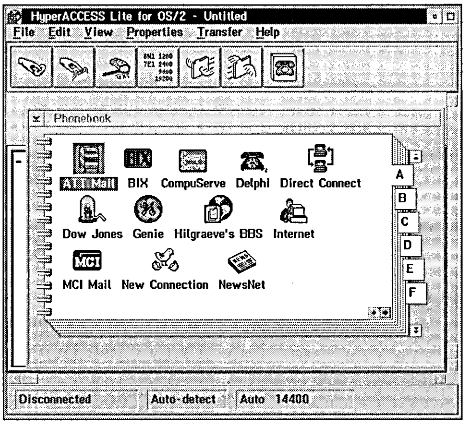
2 If you wish to avoid changing these paths every time you remove or insert the ThinkPad, I suggest the following: Install CIM-OS/2 twice, one time on your conventional drive (e.g. in my case the ThinkPad drive) and the other time on the alternate drive (my drive on the docking station).

I tried to create a second CIM object with the alternate path settings, but it did not work because CIM-OS/2 uses the same files to update the directory changes.

HyperACCESS Lite for OS/2

With HyperACCESS Lite for OS/2 (see Figure 12-39) you may explore the exciting world of modem communications with the power of OS/2 and the ease of the Workplace Shell. HyperACCESS Lite for OS/2 is an easy-to-use, reduced-feature-set version of Hilgraeve's best-selling HyperACCESS application. You just need to have a modem and a telephone line to access all these functions.

Figure 12-39. HyperACCESS Lite for OS/2



With HyperACCESS Lite, you will be able to perform the following functions:

✗ Enjoy easy installation and setup and get started fast with the clear, object-oriented design.

✗ Forget about setting parity, data bits, and stop bits: CommSense sets them all for you!

✗ Call bulletin board systems (BBSs), Internet, CompuServe, or remote systems of all kinds to get access to software updates.

✗ Transfer files fast with your choice of highly efficient, 32-bit file transfer protocols such as Zmodem, Ymodem, Xmodem, or Kermit.

✗ Emulate ANSI VT52 or VT100 terminals for communications with host computers.

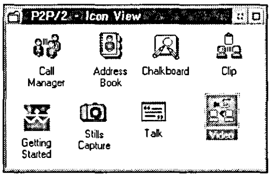
HyperACCESS Lite has also its own telephone book, which cannot be exchanged with any other BonusPak Phone Book.

HyperACCESS Lite is useful for retrieving programs from various bulletin boards. Many companies provide their software support through these bulletin boards, and if you have a fast modem, you have a quick way to retrieve software fixes.

IBM Person to Person

IBM Person to Person (see Figure 12-40) is another suite of applications that allows you to transform your PC into a communications workstation. You may communicate with your colleagues in the same building through your token ring connection and with your customers via asynchronous links.

Figure 12-40. Person to Person Applications



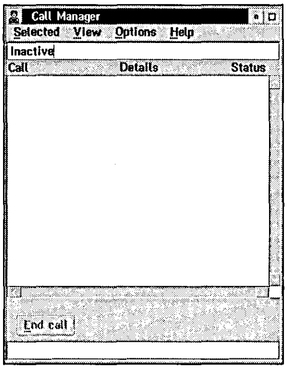
All you need to use Person to Person is a serial port with a modem attached to it or a token ring or other LAN adapter, depending on the setup you have. You can install the software through the BonusPak Installation Utility or by running the INSTALL.EXE file found on your IBM Person to Person diskette.

Note

IBM Person to Person has a two-stage installation process that is invoked by either the BonusPak Installation Utility or the INSTALL.EXE, where files are copied to your hard disk. Next you have to select **Install** for a second time to finish your IBM Person to Person installation. At that stage the program will be set up and configured.

The Call Manager feature controls Person to Person calls (see Figure 12-41). You must start the Call Manager before you use IBM Person to Person utilities such as Chalkboard or Talk. Use the Call Manager to monitor the status of calls and to end calls.

Figure 12-41. Person to Person Call Manager



The Call Manager window includes the following

- A status area showing the number of active calls and the current answer status
- A view area showing a list of all current calls or a chronological log of call events
- An **End call** push button for ending a call
- An information area that gives concise help about menu choices that you have highlighted for selection

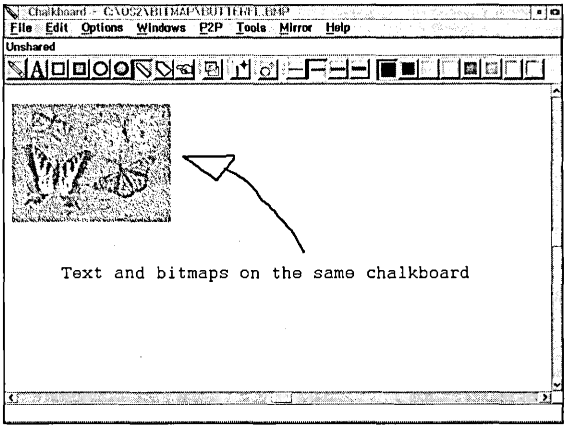
From the Call Manager menu bar you can choose

- **Selected** — To end the selected call
- **View** — To control what is displayed in the view area and to change your IBM Person to Person configuration
- **Options** — To set preferences for the way that your IBM Person to Person operates

To get help about Person to Person for a menu choice, highlight the choice and press **F1**.

You may use your IBM Person to Person program for video conferences. The only extra equipment you'll need will be a video camera on each communication side.

Through the Chalkboard you may interactively exchange information as if you were sitting next to the person you are communicating with and were both writing on the same piece of paper. You may even include images, such as bitmaps, in the chalkboard (see Figure 12-42).



Text and bitmaps on the same chalkboard

Figure 12-42.
Chalk Board

VideoIN

The VideoIN application provides support for capturing and editing digital motion video. One user can capture digital video with the necessary hardware and another can play it back using the software digital motion video player, even if the other person doesn't have the hardware needed for capturing the video (see Chapter 13 for a detailed description of VideoIN).

Multimedia Viewer

The Multimedia Viewer allows you to manage and organize your multimedia objects. These objects may be text, images, video, audio, or even animated objects. These objects are spread out on a light table with so-called thumbnail objects.

You may create several light tables for different types of multimedia objects, such as one light table only for BMP files, or one light table with all multimedia-related objects for your business presentation.

You can easily add objects to a light table by simply associating those objects with the appropriate light table. The advantage of this association is that you may even display the images of those objects with an existing light table reference even though the parent object is not present. The Multimedia Viewer comes with an audio browser, a video browser, and a text browser (see Chapter 13 for a detailed description on the Multimedia Viewer).

System Information Tool

The System Information Tool (see Figure 12-43) is designed to gather and display a variety of information about the hardware and software configuration of your system. The program is primarily designed to be used on IBM PS/1, PS/2, ValuePoint, and ThinkPad systems, but many features will function on systems from other vendors.

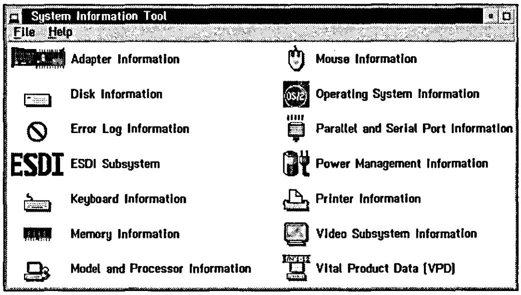


Figure 12-43. The System Information Tool Main Menu

With the System Information Tool, you can save the data you have gathered either to a file or send it to a printer. The System Information Tool provides information on the following topics:

- Micro Channel Adapter Identity, with configuration information available on many common adapters.
- Drive Information, including file system type, available space, drive size, and partition layout.
- Power Management Information
- Error Log Display and Interpretation
- Keyboard Information
- Memory Configuration, including total physical memory and installed SIMM identification
- Mouse Type and Settings
- Operating System Information, including version, DOS support, and session limits
- Model and Processor Information, including model name, processor type and speed, and BIOS date

- Parallel and Serial Port Configuration
- PCI Bus Configuration
- PCMCIA Bus Configuration
- Video System Information, including adapter type, screen resolution, and video display identification
- Printer Configuration, including data on installed printer drivers
- SCSI, ESDI, IDE/ST506, or other disk-adapter information, including devices attached, device sizes, and adapter data
- System Security Features, including power-on password and C2 features

The program also contains support for control from the command line, including textual report generation and binary history file generation and loading.

Besides the general information about your system and the parts installed within the system, there are some more advanced applications for the System Information Tool that you may find useful.

For example, if you need to know the serial number of your system's motherboard, you don't have to search for your screwdriver. You can use the System Information Tool. Select **Vital Product Data (VPD)** from the main menu. This gives you the system port part number, and even its replacement number.

In the Operating System Information you will find some information about the operating system, the window list, the CSD levels, and CONFIG.SYS file information:

- The Operating System Information shows you the version of your operating system and on which drive it is installed.
- If you have any OS/2 version installed, the Window List shows you which applications or tasks are visible in the window list and which ones are not. You will see, for example, that the PMSPOOL.EXE or the switch-to task are not visible, but you always know that they are somewhere.
- The Component CSD Levels are important to know for your service organization to find out which operating system and which CSDs or fixes have been applied to it.

- ✗ CONFIG.SYS File Information will list all files (along with the date, time, and size of the files) that are associated with a line in your CONFIG.SYS.

Summary

The BonusPak contains a suite of valuable applications, each of which offers numerous features designed to improve your productivity and make your work and home life a little easier. I have kept this chapter focused on the most popular applications, IBM Works and the IBM Information Superhighway programs. I have also offered overviews for the other programs, so you'll be aware of the tremendous variety of features available through the BonusPak.

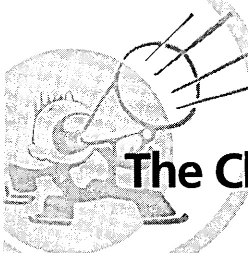
Doris Corel is a Systems Engineer with IBM Austria, on assignment in Boca Raton, Florida. She provides widespread technical information for OS/2 in the International Support Organization. She has worked with OS/2 since 1987, originally providing support to insurance, banking, and public sector customers, and later leading projects for implementing OS/2 in large LAN environments.

Chapter 13

Multimedia

The Sights! The Sounds!

Although multimedia applications often appear to be merely entertaining, or even frivolous, they nonetheless can place great demands on a PC and its operating environment. The delivery of audio or video data to the end user requires a system that not only processes large amounts of data, but does so with very precise tolerances. In other words, any lapse in multimedia delivery is painfully obvious. If someone loads a spreadsheet or opens a word processing file, delays in getting all the data loaded may be bothersome; however, when an audio-video presentation breaks up, the result is quickly considered unacceptable.



The Challenge of Delivering Multimedia Data

Multimedia presentations are so demanding that one is often quite hesitant to attempt other operations at the same time. Yet, multimedia is becoming increasingly important.

- In the home, applications such as encyclopedias, games, or a new combination of the two, called an Edutainment program, are straining the limits of the current DOS and Windows PC platforms. Since each application often has to circumvent parts of the operating environment in order to provide acceptable performance and the desired effects, users often become frustrated at having to reconfigure their machine every time they want to use a particular application.
- In the corporate environment, multimedia is quickly moving out of the category of "the $0 billion industry" (where many opportunities are cited, but actual applications are few and far between). Today, just-in-time training efforts, kiosks, and even networked "video-on-demand" are becoming more common, demonstrating the power that multimedia provides in delivering information in a much more intuitive and natural manner. As with the home market, however, existing popular PC platforms struggle to deliver the necessary performance; add the necessary subsystems for network support, and these platforms quickly break down.

Into this world comes OS/2 Multimedia support, which exploits the power of OS/2 to provide a fantastic environment for the delivery of audio-video data on your computer. In fact, an OS/2 Multimedia program is often the best demonstration of OS/2 itself, since it clearly shows the system's ability to efficiently handle multiple streams of data. A user can now play back audio, or even motion video (a "movie" requiring only a standard display adapter and sound card) data, and still perform other tasks, such as starting an application, with minimal disruption to the multimedia presentation. This power is especially evident when compared to other PC operating environments.

"But, wait," you may counter. "If I'm playing a software motion video file, why would I *want* to do something else? Wouldn't I want to just devote my attention to the movie?" This is a fair question, but potentially limiting, for two reasons:

1. Even if you have no desire to do anything else but watch the movie, you still need a powerful operating environment to respond to the demands of audio-video playback; otherwise, the movie is perceived as low quality.
2. The beauty of audio is that you can do some other activity while listening; why not take advantage of that capability?

Developer's Desktop

For example, in the OS/2 development lab in Boca Raton, using standard IBM equipment and a reasonably high-speed local area network, a developer can start a video of, say, a long but important executive presentation. The developer can then perform other activities, while listening to the video for the "good parts." When a particularly interesting topic is raised in the video, the developer can devote full attention to the video. This is a real-world example of multimedia in action, and OS/2 can deliver this capability efficiently, reliably, and effectively.

A brief history of OS/2 Multimedia

Uncensored

OS/2 Multimedia was first released as the Multimedia Presentation Manager/2 (MMPM/2) product. In mid-1992, while the development team for OS/2 Version 2.0 was working phenomenal hours to deliver their landmark 32-bit operating system, a smaller but equally dedicated group of folks was burning the midnight oil to create a set of multimedia extensions that was worthy of running on the OS/2 base. MMPM/2 1.0 was released as a separate product in mid-1992, at the same time as OS/2 2.0. This release utilized a single, generic media player for playback of audio and MIDI files, Compact Disc audio, CD-XA, and laserdiscs. It also included a small application, called the Volume Control, which allowed users to simultaneously reduce the volume of all supported multimedia devices that were playing their media. Additionally, a Multimedia Setup application was provided, allowing users to alter a few settings in the MMPM/2 system.

The next year, 1993, saw the delivery of MMPM/2 Version 1.1, but this time it was included with OS/2 Version 2.1, with the installation of multimedia support being a separate step. This release contained several new applications: a slightly refined media player, a full-function CD player, and a wave audio editor.

In addition, software motion video (SMV) was supported, with IBM supporting the playback of a new format, called Ultimotion, which provided a substantially better picture and smoother presentation than existing SMV programming. With SMV, the viewer was able to view videos without any additional hardware other than your normal video display; to hear the associated audio, all that was required was a supported sound card.

MMPM/2 1.1 also provided its first (and, up to now, only) Workplace Shell object, called System Sounds. This object, which enhanced the existing Sound object in the OS/2 System Setup folder, allowed a user-selectable audio file to be played when certain events occurred (drag and drop, warnings, errors,and so on). Along with these enhancements was support for the use of REXX to control multimedia files and devices (see the section *At Your CMD: Multimedia in REXX* for additional information). Finally, a few Lotus 1-2-3 and Microsoft Excel macros were provided, which allowed a user to perform basic audio annotation of cells in a given spreadsheet.

At this point, the terms MMPM/2 and OS/2 Multimedia could be used interchangeably, since this support was only available via the OS/2 product, and not through a separate package. The installation, however, was a separate step, apart from the initial installation of OS/2. In the latter part of 1993, the Ultimedia Video IN/2 product was released. Video IN provided subsystems and additional utilities, allowing the viewer to *capture* software motion video, using a variety of popular video capture devices.

In October 1994, OS/2 Warp went "golden" and, again, additional multimedia features were included. OS/2 Warp continued to integrate MMPM/2 into the base system and provided improved multimedia device recognition during its installation. In addition, subsystem support was added to support a number of video capture adapters, Kodak Photo-CD CD-ROMs, and MPEG file playback, using Sigma Designs' Reel Magic hardware. Moreover, the OS/2 BonusPak provided additional multimedia support:

✗ Video IN, previously available as a separate set of software motion video capture device drivers and utilities, with support for additional video capture hardware.

Note

Video capture hardware is not strictly necessary to utilize Video IN; see the section on Recording Software Motion Video Using Video IN for more details.

✗ The Multimedia Viewer, which included special Workplace Shell folders and browsers, as well as support for additional audio and image formats.

Note that the IBM WebExplorer, an Internet World Wide Web browser application that can be obtained from IBM, utilizes the Multimedia Viewer "applets" as the default "viewer" programs for audio, video, and image data.

Uncensored

As a result of these changes, today more multimedia support than ever is included in the OS/2 Warp package. Over the last three years, many bugs have been fixed, installation has improved, and support for new audio and video devices is being added each week, commensurate with OS/2 Warp's growing popularity. The resulting OS/2 Multimedia platform is robust, tested, and "ready for prime-time!"

OS/2 Warp: A solid platform for robust Multimedia

A technical treatise is not required here; besides, it's not nearly as interesting as actually playing with the applications! It is worth noting, however, that OS/2 Warp provides a wonderful set of features that provide a stable and powerful multimedia platform. These features include:

- **Preemptive multitasking** — This simply means that the operating system decides which of the currently running programs gets to use the processor. Contrast this feature with the design of other popular operating environments/systems such as Microsoft Windows and Apple's System 7, which operate on a scheme called *cooperative multitasking.* In cooperative multitasking the application decides when to give up the processor.

  This distinction may not seem important today since, in reality, you may not care to have other programs running while you're using your multimedia application. However, how many times have you clicked the mouse, or pressed a key, to stop a particular bit of multimedia playback, only to be forced to wait until the application decides to respond to your input?

  Obviously, the application can be written to handle this situation, but it is more complicated in a non-OS/2 Warp environment, and such complexity can lead to a more expensive and less reliable application. Moreover, as time marches on and your computer becomes increasingly connected to your home, your cable company, and so on, preemptive multitasking will become a crucial element in ensuring that your system delivers the multimedia information you desire, while performing other important tasks as well.

- **Multiple threads of execution** — Somewhat related to preemptive multitasking, this feature means that MMPM/2 can ensure that streams of audio and video data are given the appropriate (higher) priority in the system as individual tasks. Other tasks will still run, but the multimedia data will flow essentially uninterrupted to the output device(s).

- **Robust device sharing among applications** — The concept of allowing multiple applications to have simultaneous access to a given resource (for example, a CD-ROM device for audio compact discs or a sound card for wave audio playback) is foreign in a DOS environment, which is essentially single-tasking.

In Windows, such capabilities are possible, but in practice Windows multimedia applications generally do not do a very good job of sharing the devices they use. On the one hand, this is understandable since current user habits tend to allow a single application to consume all resources. On the other hand, wouldn't it be handy to view, say, an instructional video and, at the same time, have it paused automatically in order for a background task to provide some audio indication of status or some other notification?

With OS/2 Warp and its associated multimedia components, that platform exists today. Well-written OS/2 Warp applications can utilize multimedia capabilities with a small amount of additional programming, and users can benefit from a truly robust platform and start (dare it be said?) enjoying their time on a computer, even when performing otherwise mundane tasks.

A Tour of OS/2 Warp Multimedia

Once installed, the various OS/2 Warp Multimedia applications and capabilities are just a few mouse clicks away. This chapter discusses the extent to which multimedia is supported by OS/2 Warp, explores the Multimedia folder, looks at the BonusPak Multimedia Viewer and Video IN applications, and describes some of the on-line information available to OS/2 Warp multimedia users.

In terms of *what* is supported, OS/2 Warp Multimedia recognizes, "out of the box," more multimedia file and device types than many other operating environments. Included in the base support, either through the initial installation or via the BonusPak, are a wide variety of digital audio file types, as well as software motion video .AVI files, compressed using either IBM or Intel algorithms; all you need is a supported sound card to play back any of these multimedia files. Moreover, a number of video *capture* devices are supported by OS/2 Warp; these devices let you create your own software motion video .AVI files. Some video capture device support can be installed via OS/2 base installation; other capture card support is available via Video IN installation on the BonusPak.

Another type of video, called MPEG (Moving Picture Experts Group), is also supported. MPEG videos usually provide a large picture area, but with smooth video and CD-quality audio; however, an additional MPEG hardware adapter is required.

In the image arena, OS/2 Multimedia recognizes many of the popular image formats, including Kodak's Photo CD on CD-ROMs. Table 13-1 summarizes the different file types supported by OS/2 Warp.

Table 13-1. Summary of Supported Multimedia Files

Description	Typical File Extension(s)	Installed Via
Audio Files		
Digital wave audio	WAV	OS/2 Warp Base
Creative Labs voice	VOC	OS/2 Warp Base
IBM Audio Visual Connection (AVC) ADPCM	_AU	OS/2 Warp Base
Apple Interchange File Format (AIFF) wave audio	AIF	BonusPak
Interchange File Format (originally popular on Commodore Amiga computers)	IFF	BonusPak
Unix/NeXT/Sun SND format	AU, SND	BonusPak
Video Files		
Audio Video Interleaved (Note 1)	AVI	OS/2 Warp Base
Autodesk Animator animation (Note 2)	FLI, FLC	OS/2 Warp Base
Moving Picture Experts Group (MPEG, Note 3)	MPG, VBS	OS/2 Warp Base
Image Files		
IBM AVC Still Video Image	_IM	OS/2 Warp Base
IBM M-Motion Still Video Image	VID	OS/2 Warp Base
IBM OS/2 1.3 Bitmap	BMP	OS/2 Warp Base
IBM OS/2 2.0 Bitmap	BMP	OS/2 Warp Base
Microsoft Windows Bitmap	BMP	OS/2 Warp Base
Kodak Photo CD	PCD	OS/2 Warp Base
Microsoft Windows Device Independent Bitmap	DIB, RDI	OS/2 Warp Base

continued

Table 13-1. Summary of Supported Multimedia Files, continued

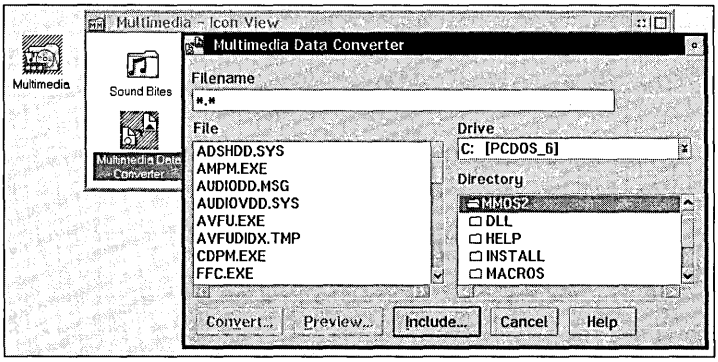
Description	Typical File Extension(s)	Installed Via
Graphics Interchange Format	GIF	BonusPak
PC Paintbrush	PCX	BonusPak
Targa (Note 4)	TGA	BonusPak
TIFF format (Note 5)	TIF	BonusPak

Table Notes:

1. *An AVI file can contain one of a number of different compression/decompression (codec) algorithms. The codecs supported by OS/2 Warp are Intel's Indeo 2.1, Indeo 3.1, Indeo 3.2 (playback only), and IBM's Ultimotion. Microsoft's Video 1 and SuperMac's Cinepak are not supported by OS/2 Warp, but can be obtained via shareware on the Internet (see the section on Additional Multimedia Support and Applications).*
2. *FLC and FLI animation files are supported for playback only.*
3. *MPEG files require a supported hardware adapter.*
4. *OS/2 Warp supports compressed and uncompressed TARGA image files.*
5. *OS/2 Warp supports FAX compressed and Intel and Motorola (compressed and uncompressed) TIFF image files.*

The Multimedia Data Converter application, located in the Multimedia folder, can convert supported audio and image files from one format to another. This application is also handy in that you can see which audio and image file formats are supported by selecting the **Include...** button in the main window (see Figure 13-1).

Figure 13-1. Multimedia Folder With Multimedia Data Converter



Filename: *.*

File:
ADSHDD.SYS
AMPM.EXE
AUDIODD.MSG
AUDIOVDD.SYS
AVFU.EXE
AVFUDIDX.TMP
CDPM.EXE
FFC.EXE

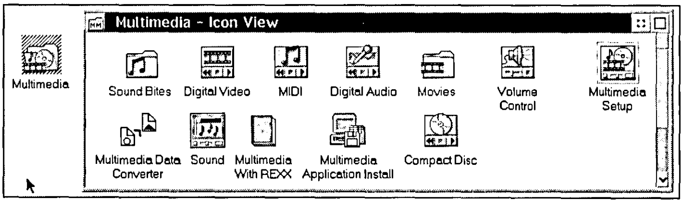
Drive: C: [PCDOS_6]

Directory:
MMOS2
DLL
HELP
INSTALL
MACROS

Multimedia folder

After OS/2 Multimedia is installed, the Multimedia folder resides right on your desktop. A sample desktop with the open Multimedia folder is shown in Figure 13-2.

Figure 13-2. Partial Desktop Showing the Multimedia Icon With an Open Multimedia Folder



A number of handy objects are created in the Multimedia folder:

- Two other folders, Sound Bites and Movies, contain various audio and software motion video files. You can double-click directly on these files to start the appropriate OS/2 Multimedia player/editor application.
- The Multimedia Data Converter allows you to convert certain audio and image files from one format to another.
- A Volume Control application acts as a master audio volume dial, allowing you to control, from one spot, the volume levels of any active multimedia player/editor.
- A Multimedia With REXX on-line information book, which allows you to easily utilize multimedia in command files. See the section entitled At Your CMD: Multimedia in REXX for details on using REXX to control multimedia files and devices.
- A Multimedia Application Install program allows you to install other programs that have provided MMPM/2 installation "scripts."

Warning

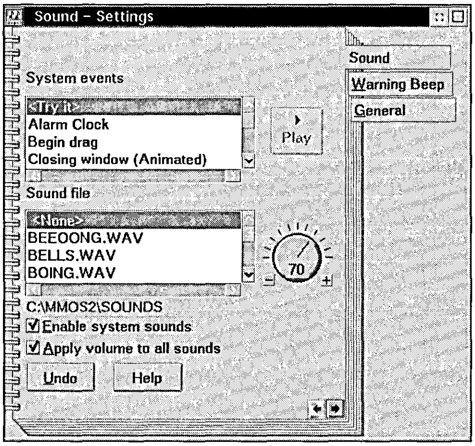
An important clarification is in order here: if you wish to add or change the OS/2 Multimedia support in your system (for instance, if you change audio hardware), then you must utilize OS/2's Selective Install program, which is found in the System Setup folder. The Multimedia Application Install (MINSTALL.EXE) program was originally used to install MMPM/2 when it was a separate package; now it is relegated to installing third-party applications, including the OS/2 Warp BonusPak Video IN and Multimedia Viewer components (see the corresponding sections in this chapter for additional information). You should use the Multimedia Application Install program only if an application specifically says to do so.

Other objects in the Multimedia folder are described in more detail in the following sections.

System sounds: more than just beeps

After OS/2 Multimedia is installed, the Sound object in the System Setup folder is updated to incorporate some multimedia System Sounds. These system sounds allow you to play back selected .WAV files that are associated to some system event, such as a warning alarm (for example, when you clicked on a grayed-out button). A shadow of the Sound object is also created in the Multimedia folder. Figure 13-3 shows the Sound object opened to its Settings view. By default, system sounds are enabled, and the initial directory of available sounds is the \MMOS2\SOUNDS directory on the drive on which you installed OS/2 Multimedia.

Figure 13-3. OS/2 Sound Object Settings View


Sound - Settings

System events
- <Try it>
- Alarm Clock
- Begin drag
- Closing window (Animated)

Sound file
- <None>
- BEEOONG.WAV
- BELLS.WAV
- BOING.WAV

C:\MMOS2\SOUNDS

☑ Enable system sounds

☑ Apply volume to all sounds

Setting a system sound is easy: simply select the desired event in the System events list, then select the desired sound from the Sound file list. To listen to the desired sound, press the **Play** button.

Hint: you can also just double-click the mouse pointer on the name of the sound file in the list. If you would like to "audition" the sound before assigning it to a particular event, then select <Try it> from the event list, then select the desired sound file and play it.

If you have some .WAV files that you would like to use for system sounds, there are two ways to assign the files to an event. The first way is to utilize the simple drive and directory navigation capability found at the end of the Sound file list. Double-click on the appropriate entries to change to the desired path containing your sound files. The drive and directory are displayed immediately below the Sound file list.

The other way to select a wave file for a particular event is to select the desired event, open the appropriate folder containing the file, and simply drag and drop the desired file onto the Sound object.

The <None> entry in the Sound file list allows you to suppress playing a .WAV file for a given event. If you select this item, or cause an event alarm to sound when the Sound object cannot play an audio file, then the appropriate default system action occurs, which is normally a speaker beep or just no sound at all.

Note

Two system events, Opening window and Closing window, include the word *(Animated)* in their event description. A system sound will be generated only if you have enabled Window Animation (see Chapter 7) and the application itself has been programmed to utilize animated windows. In other words, if a window is not set up for animation, then no system sound will be heard when it opens or closes.

System sounds are fairly entertaining. In addition, as more programs utilize certain OS/2 features causing an event alarm to sound, system sounds can be useful, too. For example, some corporate installations use system sounds to provide a telephone number to call if a certain event (such as a severe error) occurs. However, in some cases, one may find that system sounds "get old" after a while and, after all, they do consume a small amount of system resources.

Note

The Sound object has the capacity to "disable" system sounds, but much of the support program is still loaded into the system, even though no sounds will be heard. Fortunately, you can remove the system sounds feature, as well as "re-enable" it later.

If you would like to disable system sounds and ensure that no relevant code is loaded on subsequent reboots, then simply do the following:

1 Open an OS/2 Window or an OS/2 Full Screen Command Prompt.

2 Change to the \MMOS2\INSTALL directory on the appropriate drive.

3 Enter the following REXX command at an OS/2 Command Prompt:

```
DINSTSND
```

4 Shutdown your system and reboot. You will find that the Sound object has now reverted to its nonmultimedia appearance.

Later, if you would like to utilize the system sounds again, then switch to the \MMOS2\INSTALL directory, as in the previous steps, but enter the following REXX command at an OS/2 Command Prompt:

```
INSTSND d:
```

where d: is the drive containing the MMOS2 directory.

The CD player

If your system has a CD-ROM device, then the Compact Disc object, shown in Figure 13-4, is installed in the Multimedia folder. This application (CDPM.EXE) has many of the features normally found on standard stereo CD players, with the goal of incorporating this function in as compact a window as possible.

The 70% Solution

An interesting, behind-the-scenes story about the default volume level for system sounds demonstrates how small details can be overlooked and why it's important to have many people test the system in real-world configurations before shipping the product. Toward the end of the testing phase of OS/2 Warp, many managers and staff on the development team were installing OS/2 Warp everytime it was "built" (that is, every time it was revised because of reported defects). This included installing the OS/2 Multimedia support on a number of IBM ThinkPad machines, which contained the appropriate audio hardware.

As it turned out, a number of the multimedia developers' personal test machines either had System Sounds disabled (because of requirements brought on by "debugging" problems) or were using older, less popular hardware. At the time, the default System Sounds volume was fairly high, about 85 to 100 percent. On the project management teams' ThinkPads, which were more sensitive to a high-volume level than the developer's hardware, this volume caused people to literally jump out of their seats when the "start up" sound was played during OS/2 system initialization. Needless to say, a fair amount of "quick focus" was placed on the problem, and the default System Sounds volume was reduced to a level that was acceptable to the most customers.

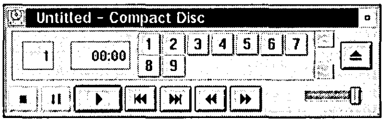


Figure 13-4.
CD Player in Its Default View

While not a Workplace Shell object itself, the CD player provides a pop-up menu, similar in look and feel to that of a real Workplace Shell object. To display this menu, simply press mouse button 2 anywhere in the window (see Figure 13-5). You can also display this menu by single-clicking mouse button 1 on the system menu in the upper-left corner of the CD player window.

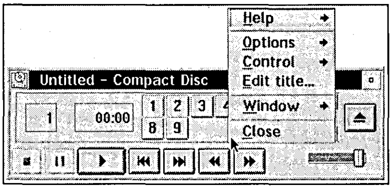


Figure 13-5.
CD Player With Open Pop-up Menu

The OS/2 CD player can begin playing the disc as soon as it is inserted into the drive; select **Options Automatic play** from the pop-up menu. You can also give your CD a name by selecting **Edit title...** from the context menu. The title you give a CD will be remembered by the CD player, even after removing the CD from the drive. Try it!

Uncensored

Other features include random playback (Options Shuffle tracks in the pop-up menu) and program repeat (Options Repeat). With the shuffle option, there is a small bug (perpetually rationalized as a feature in the halls of OS/2 Multimedia development) where the current CD track is always first in the random list!

Another slightly obscure, but quite handy, feature of the CD player is that it can suppress tracks from being played. To suppress a track, press Shift and click on the track number to be disabled. An example of the CD player interface, after some tracks have been disabled, is shown in Figure 13-6. To re-enable the track for playback, simply click on the track button again.

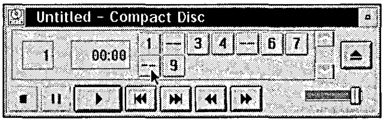
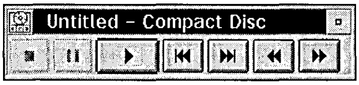


Figure 13-6.
CD Player With Disabled Tracks

The CD player was designed to be as compact as possible and yet still provide a fair number of features and functions. Often, of course, you may just want to open the CD player and have it occupy as little space as possible, especially after you have set up the options according to your preferences. To alter the size and position of the CD player, the CD player provides two capabilities. First, the application has sizeable borders, which allow you to shrink the entire window to a very small "footprint." Second, two menu options are provided in the pop-up menu, Window Default size and Window Compact size, which change the size of the window to preset dimensions. The default size is the minimum size of the window, which still displays all possible buttons and sliders, while the compact size reduces the window size to an area that just contains the CD control buttons, as shown in Figure 13-7.

Figure 13-7. CD Player Shown in Its Compact Size



Also note that, at all times, the border of the CD player can be sized however you wish, allowing you to set (and save) a very small "footprint" of this application.

Finally, here's a fun little CD player trick that demonstrates the underlying power and flexibility of the OS/2 Multimedia support:

1 Open the **Compact Disc** object in the **Multimedia** folder. Start playing your CD.

2 Start another CD player, either by copying the **Compact Disc** object, or entering **cdpm** from an OS/2 Command Prompt.

3 In the second CD player, select a different track and start playback.

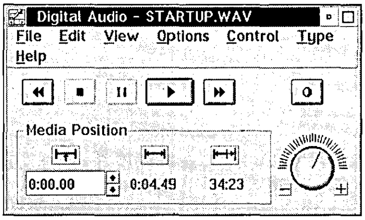
4 Click back and forth on each CD player, and note how quickly each player switches to the correct position in the CD for its particular window.

The *Digital Audio* object: "Wee-Pum"

"Wee-Pum" is how the Digital Audio Editor application is affectionately pronounced by members of the OS/2 Multimedia development group; the name of the program is WEPM.EXE (for Wave Editor, Presentation Manager). This application provides you with a very nice set of basic wave-editing features.

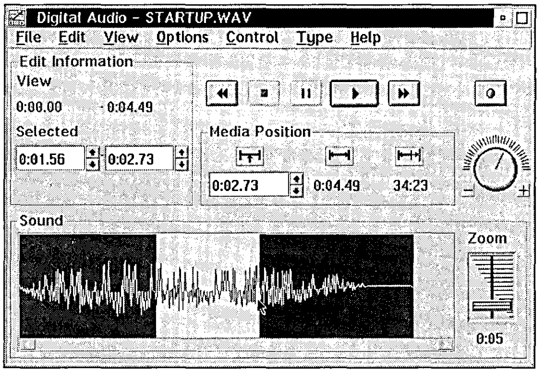
Figure 13-8 shows the Digital Audio application with its "default" (non-editor) view. This view can be changed by selecting **View Editor** from the menu bar; the resulting user interface is shown in Figure 13-9. The editor view displays a graphical representation of the audio file, and lets you select a range of this audio on which to perform various operations.

Figure 13-8. Digital Audio Wave Editor Application in Its Player/Recorder View


Digital Audio - STARTUP.WAV
File Edit View Options Control Type Help

Media Position
0:00.00 0:04.49 34:23

Figure 13-9. Digital Audio Wave Editor Application in Its Editor View With a Range Selected


Digital Audio - STARTUP.WAV
File Edit View Options Control Type Help
Sound
Zoom

Edit Information
View
0:00.00 0:04.49
Selected
0:01.56 0:02.73
Media Position
0:02.73 0:04.49 34:23
Zoom 0:05

After installation, the Digital Audio application is automatically associated with files having extensions of WAV, IFF, AIF, AU, and SND or file types of Digital Audio. You can, therefore, double-click on one of these files directly, and the Digital Audio application will be opened. Another way to load a file for manipulation, aside from the obvious and comfortable File Open selection, is to open the folder containing the audio file you are editing and simply drag the file onto an open Digital Audio application.

A few features of the Digital Audio editor are especially noteworthy. One is the application's capacity to perform a number of digital effects upon a selected portion of the file being edited. These options can be found in the bottom section of the Edit menu (see Figure 13-10). These features came about when two OS/2 Multimedia developers, Ron Jones and David Killian, decided that some additional audio effects simply *had* to be included in the Digital Audio editor. Ron and David, who were both "into" signal processing anyway, convinced the management team of the worth of the effects code, and literally worked day and night, in addition to their normal assignments, to ensure that the audio effects were coded, tested, and ready to go into the product.

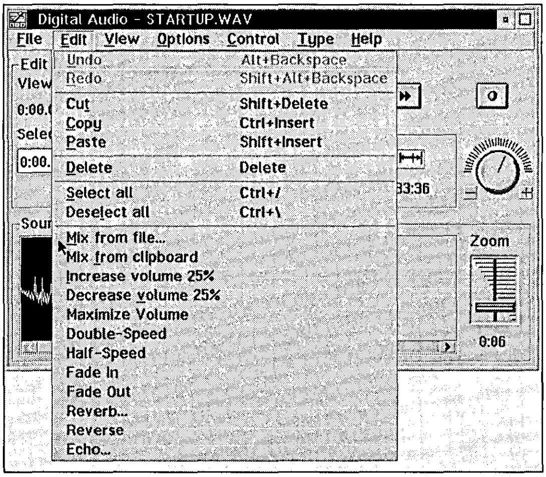


Figure 13-10. Digital Audio Wave Editor With Edit Menu Displayed

The audio effects available with the Digital Audio editor are as follows (remember, a range of audio data must be selected in order to use these options):

- **Increase volume 25 percent** — Slightly raises the volume of the selected range.
- **Decrease volume 25 percent** — Slightly lowers the volume of the selected range.
- **Maximize Volume** — Sets the volume level of the selected range to the maximum possible value.
- **Double-Speed** — Causes the selected range to play back at twice normal speed. The pitch of the playback will be raised.
- **Half-Speed** — Causes the selected range to play back at 1/2 the normal speed. The pitch of the playback will be lowered.
- **Fade In** — Alters the volume of the selected range of data such that the beginning of the range plays softly, with the volume rising to the selected level.

- **Fade Out** — The complement of Fade In, where the audio at the beginning of the selected range is at the normal level and drops to a very low level by the end of the selection.
- **Reverb...** — Provides a slight "doubling" of the original selected audio. The user can adjust the amount of delay between the original sound and the duplicated sound; the volume of this "secondary" audio can also be adjusted. This audio effect provides you with audio that sounds more like it was recorded in a large room. Contrast this effect with **Echo...**, below.
- **Reverse** — Alters the selected range such that the selection is played backwards. Now, at long last, you can settle arguments about controversial rock'n'roll musical passages,which sound like they are played in reverse; simply record the passage in question, select it for editing, reverse the audio, and settle the bet!
- **Echo...** — A "doubling" of the sounds in the selected range. User-selectable delay and volume settings let you vary the "amount" of echo that you wish to create. This feature is very similar to the **Reverb...** option; the main difference is that with echo, the duplicated sounds are more distinct and separate than the main sounds.

Some noise may be introduced if you repeatedly choose **Double-speed** and then "back out" the changes by selecting **Half-speed**, or vice versa. Also, you can select or deselect the entire audio file via the Ctrl-/ and Ctrl-\ key combinations, respectively.

You can perform as many Undo and Redo operations on the file as you desire, to allow you to back out or re-apply, respectively, different changes to the file that you may have made in the course of editing.

Another interesting feature of the Digital Audio editor is its capacity to easily mix, or combine, one file with another. Similarly, you can mix in any audio data that has been copied or cut to the clipboard; in fact, this data will be mixed in even if the format of the data (such as sample size, samples per second) differs from that of the currently loaded file. See the PLAY.CMD and RECORD.CMD sections for a few tidbits about digital audio data characteristics. All you have to do is select all or part of an existing wave audio file, then select **Edit Mix from file...** or **Edit Mix from clipboard....** The results can be quite satisfying!

One last hint when using the Digital Audio editor: when recording, it's important to know exactly where the recorded data is stored for a new file. See the section entitled Multimedia Setup for more information about the Multimedia work path and how to change it.

The Media Player: power in a plain wrapper

The unassuming look of the OS/2 Media Player belies the power of the underlying OS/2 Multimedia platform. Nonetheless, any playable multimedia file or device type supported by OS/2 Multimedia (MMPM/2) can be manipulated with this demure little application.

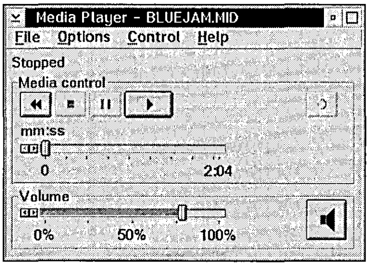


Figure 13-11. Media Player Playing a MIDI File

The Media Player user interface was designed to handle the playback of almost any kind of media in a generic fashion. Of course, the trade-off is the lack of any controls that are specific to a given type of device (for instance, step forward/backward buttons, which would be useful when playing compact discs). However, as time passes, and more device-specific interfaces are provided, the Media Player can be further relegated to a role of providing an interface for files and devices which may not have a specific player interface available.

In OS/2 Warp, the Media Player provides an interface for controlling software motion video (SMV) and Musical Instrument Digital Interface (MIDI) files. When an .AVI file is being played, a separate "default playback window" is created by the underlying subsystem; the Media Player is actually unaware that this window has been created (refer to Figure 13-12)! You can start the individual player from the Multimedia folder or, in the case of file-based media, drag and drop the desired file onto the player object. In addition, you can just double-click on a .FLC, .FLI, .AVI, .MID, or (with the appropriate hardware and installation) .MPG file and the Media Player will start automatically.

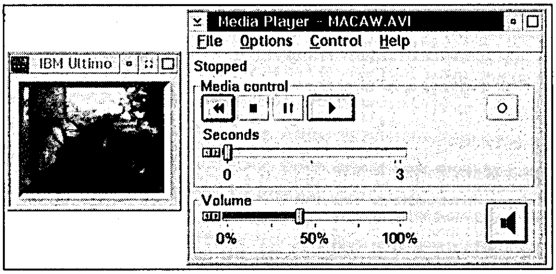


Figure 13-12. The Media Player Playing MACAW.AVI

When using the Media Player as a Digital Video application, it's interesting to note a few capabilities of the default video window, which is opened when the .AVI file is loaded by the Media Player. Figure 13-13 shows the system menu for a default video window. By clicking on this window and selecting the system menu (or pressing one of the key combinations), you can alter the size of the video or obtain information about the movie, including frame rate, normal frame size, data transfer rates, and codec type.

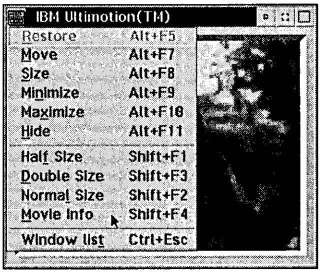


Figure 13-13. The System Menu for the Media Player's Default Video Window

A few Media Player command line options are documented in the Multimedia on-line information, located in the Information folder.

However, other undocumented command line options exist. For completeness, the entire set of command line options is shown here ([ ] indicate optional parameters and are not specified in the actual command):

```
mppm [device] [filename] [/s[c]] [/a] [/L > outfile]
```

The device parameter allows you to specify a particular multimedia resource name; in MMPM/2 terms, this is called the *logical device name.* Simply put, the logical device name is the name of a particular multimedia device on your system. Some examples of logical device names are waveaudio (for .WAV file playback); waveaudio01, cdaudio02, sequencer01 (for MIDI playback); and digitalvideo01 (for .AVI file playback). If the device name ends with a two-digit number, it indicates which device of that type to use, out of multiple devices of that type. If no number is specified, then the "default" device of that type is used; the default can be changed by using the Multimedia Setup utility. Generally, you will only have one each of a given type of multimedia device, although systems may occasionally have multiple sound cards or more than one CD-ROM drive.

The filename parameter indicates which multimedia file to play back if the device requires a file (a sound card requires a .WAV file or equivalent, while a CD player does not use files). If no device is specified along with the filename, the appropriate device associated with this file will be used; again, you can use the Multimedia Setup utility to change which device to use, if you have more than one in your system. The full path name of the file must be specified if the file is not in the current directory in which you are invoking the Media Player.

The /s parameter allows you to start playback of the file or device as soon as the player is ready to do so. A "companion" parameter, /sc, performs the same function but, in addition, will close the Media Player as soon as playback is complete.

The /a parameter is short for *Ad nauseam* and, as the name suggests, allows you to play a selection repeatedly.

Uncensored

This option was so named in honor of the poor folks who attended various trade shows and were forced to listen to the same audio and video over and over again.

The /a parameter is useful for specialized applications such as demonstrations. The Media Player has the following behavior with respect to the /a option:

- ✗ You can specify /a without specifying a file or device. When the user interface is opened and you specify the required device or file, pressing the **Play** button will "engage" the *Ad nauseam* feature.
- ✗ If you press the **Pause** button while playing a selection, the /a option remains in effect. Resuming the playback will repeat the selection. This allows you to select another file or device through drag and drop or by using the **File Open** or **Options Change device...** menu items.
- ✗ Pressing the **Stop** button during playback will turn off the /a option. The selection will not play repetitively. The application must be closed and re-opened with the /a option in order to play selections *Ad nauseam.*

Tip

If you experience problems when using the Media Player, or are just interested in seeing the kinds of commands that are passed from the Media Player to the OS/2 Multimedia subsystem, then specify **/ L > outfile** at the end of your **mppm...** command line, where **outfile** is the appropriately named file to contain the output of the program. Simply use the Media Player as you normally would and, upon closing the application, view the filename you specified as **outfile.**

Warning

Be a bit careful in your use of this function; the Media Player will write out a message to this file every half second, so if you are playing, say, a compact disc, you may wind up with a fairly large outfile.

Another interesting quirk with the Media Player is how it saves away the size and location of its main window. These values are stored in association with the title of the Media Player, excluding any file name, when it was last closed. The US English default title (which would, for example, be displayed when the Media Player is invoked from the command line) is Media Player. However, remember that the MPPM.EXE program is also part of objects named MIDI and Digital Video in the Multimedia folder. You can therefore save different window sizes and positions on your desktop for each of these different objects.

A curious example of this feature is to start MPPM.EXE from the command line, hide/minimize the resulting window, and close the application from the Window List. Now, you have a Media Player you can use to play files automatically (utilizing the /s option) without showing its user interface. To restore the settings of a given Media Player that has been hidden in this manner, simply open it the way you normally would, highlight it in the Window List, select **Show** from its Window List pop-up menu, and close the application while it is visible.

Tip

Another use of this flexible window placement/sizing scheme is that you can make a given Media Player window smaller and save that "look" based on the application's title. You can size the window down so that it just shows the control buttons (you can also "de-select" **Options Show status line** to make the window even more compact). Then, when you close the application, future uses of this particular title will result in a much more compact Media Player interface.

When the window is sized smaller than the normal size, scroll bars will appear, allowing you to scroll around the dialog without resizing! If you wish to reset the Media Player window to its regular size, select **Default Size** from the system menu (see Figure 13-14).

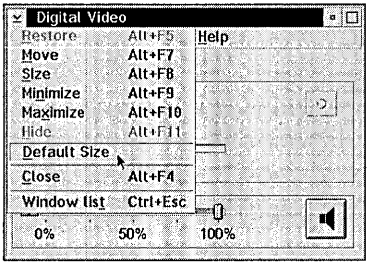


Figure 13-14. The Default Size Option for the Media Player

One last item that is not particularly obvious is the Media Player's support for direct manipulation. You can drag and drop any supported audio, video, or MIDI file onto any of the various Media Player objects (MIDI, Digital Video) in the Multimedia folder. Once the Media Player window is opened, you can drag/drop a file onto that window.

Since the system won't let you drop a file onto a button or other control in the window, it's best to drop the file onto the area between the Play button ▶ and the Record button. ●

Multimedia setup

The Multimedia Setup object in the Multimedia folder starts the program STPM.EXE. This program allows you to alter a few key settings within the OS/2 Multimedia system. In reality, you will rarely need to change these settings, but a short tutorial about certain features is appropriate in case you do need to modify the default configuration.

When a new Digital Audio or Digital Video file is created, or when any editing is performed upon an existing file, OS/2 Multimedia interrogates a setting called the Multimedia work path and places in that directory a temporary file named, for example, 00000000.001. The default multimedia work path is \MMOS2 on the appropriate drive. You could experience disconcerting error messages if this drive happens to be either low in disk space or contains the OS/2 swapper file (refer to the SWAPPATH statement in your CONFIG.SYS file for the location of your SWAPPER.DAT file). In the case of a default installation, the latter situation would be true, as both the multimedia work path *and* the OS/2 swapper file reside on the boot drive (usually C:).

If you anticipate recording digital audio files of any significant length, you should ensure that you set the multimedia work path to a disk partition that contains plenty of free space.

Another handy setting involves your CD-ROM device. The drive letter for this device is always determined after the system has recognized all other disk drives and assigned their respective drive letters. If you add a hard drive, then your CD-ROM's drive letter will automatically change. While OS/2 is able to handle and assign a new drive letter for the CD-ROM device, the OS/2 Multimedia subsystem is not quite as intelligent. For this reason, the Multimedia Setup notebook contains a Compact Disc page, with an associated **Drive Letter** page. This page allows you to specify the new drive letter that OS/2 has associated with your CD-ROM device and also allows your CD Player to function properly again if you add additional hard disks to your system.

One other feature added for your convenience involves multimedia file types and extensions. For each multimedia device in your system that uses files, the Multimedia Setup application provides two settings pages: a Names page and an Association page.

- ✗ The Names page is really a holdover from earlier versions of MMPM/2, when the Media Player application was the only program available to play any kind of multimedia device, and is of limited value in the latest versions of OS/2 Multimedia.
- ✗ The Association page, however, allows you to specify new or different file extensions or file types that should be recognized by OS/2 Multimedia. You can,therefore, name a file as you wish and simply specify the desired file extension or file type, and the file will be handled correctly by OS/2.

  Again, this feature may not be so useful for most normal configurations, but it does provide some additional flexibility for more adventurous users or unique systems.

Finally, here's a hidden treasure: If you have installed one of the video capture adapters (or one of a similar type) supported by OS/2 Multimedia, then Multimedia Setup provides a quick monitoring facility. Open the Multimedia Setup notebook, select the Digital Video 2 tab, and then select the Defaults tab. There, you can select buttons to adjust settings on your video capture card, monitor the incoming video, or even, in the case of supported WinTV hardware, change the TV tuner channels.

But, wait, there's more: OS/2 Multimedia in the BonusPak

The multimedia applications and platform found in the base OS/2 Warp system are nicely supplemented by various additions to the BonusPak. Included in the BonusPak are the following items:

- The **Multimedia Viewer** set of applications, which provides an enhanced, "multimedia-file-aware" folder, as well as support for additional image and audio file formats.
- The **Ultimedia Video IN/2** product, which allows you to perform extended editing of software motion video (SMV) files and, with the appropriate video capture hardware, create your own high-quality, smooth running SMV files.
- Additional images and SMV movies for your entertainment.

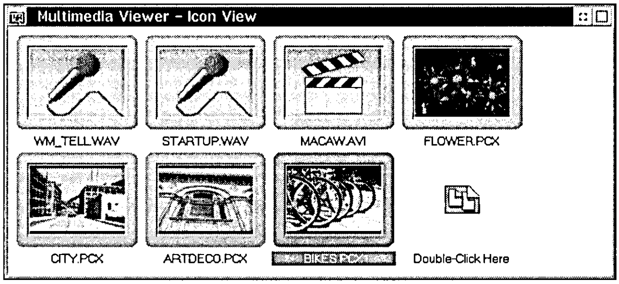
The Multimedia viewer

The Multimedia Viewer is a set of Workplace Shell extensions that provides a new type of folder, which is more aware of multimedia file types and displays them in a slightly different view. This folder type mimics a photographic light table, on which a photographer places slides for reviewing.

The advantage, of course, is that with this type of folder, pictorial representations of an image file can be viewed without necessarily viewing the file itself. This is significant if you are viewing a Kodak Photo CD disc, where a single image can consume many megabytes of disk space (or memory, when processed by an application). With the Multimedia Viewer light table, there will be some initial time spent processing the image, but when the folder is subsequently opened, the time required to display the images is much quicker (and remember, the image does not have to be displayed via an application). Moreover, light table references will show the image even if the media containing the original image is not available! This is especially convenient with image files on removable media, such as diskettes or CD-ROMs.

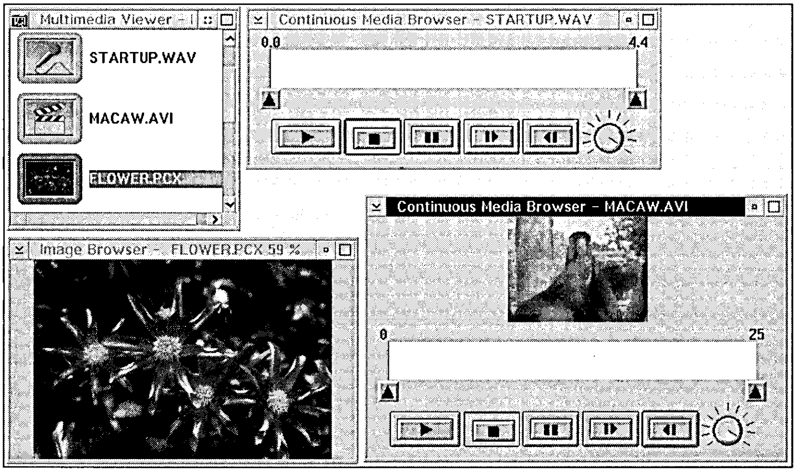
When the Multimedia Viewer is installed, the pop-up menu of any data file on your system has a new menu item added, called **Create LT Reference...**, where LT stands for Light Table. This option lets you add a special kind of object to a Multimedia Viewer folder. An LT Reference is very similar to a shadow object (refer to Chapter 5); however, when an LT Reference object is added to a Multimedia Viewer folder, a high-quality "thumbnail" image is created and used as the object's icon. Each object of this type has a border that resembles a photographic slide, as shown in Figure 13-15. Of course, any file can be copied, moved, or "shadowed" into a Multimedia Viewer folder, and they will be represented by the same icon as any other folder, with one exception: Multimedia Viewer files that are copied or moved into the folder will appear as a "slide" object.

Figure 13-15. Open Multimedia Viewer Folder



A Multimedia Viewer slide object has a few features worth knowing about. First, if you double-click mouse button 1 on the slide portion of the object's icon, you will invoke the Multimedia Viewer's browser program, which displays the file's contents (if the file is an image) or provides controls to play the file (if the file contains audio or video data). See Figure 13-16 for examples of the various browsers. If you double-click on the central portion of the object icon's ("inside the slide"), you will get the default view for that object. You can get a pop-up menu for the object in the normal fashion by single-clicking mouse button 2 on any part of the object's icon.

Figure 13-16. Multimedia Viewer Folder With Various Browsers Opened


Multimedia Viewer - I
STARTUP.WAV
MACAW.AVI
FLOWER.PCX
Continuous Media Browser - STARTUP.WAV
0.0
4.4
Image Browser - FLOWER.PCX 59 %
Continuous Media Browser - MACAW.AVI
0
25

The audio, image, and video browsers supplied with the Multimedia Viewer are named AB.EXE, IB.EXE, and VB.EXE, respectively. These browsers can be invoked from an OS/2 command line; simply provide the appropriate filename as a command line argument. If you do not provide a filename to process, a file open dialog box will be displayed.

Note

These Multimedia Viewer browser programs are used by the IBM WebExplorer application as the default viewers for various audio, image, and video files.

Recording software motion video using Video IN

Formerly available only as a separate product, the Ultimedia Video IN/2 product is now included in the OS/2 Warp BonusPak. With this set of tools, you can manipulate .AVI (Audio Video Interleaved) files and perform basic functions such as cut/paste, audio-video synchronization, bitmap capture and display and, of course, recording. While certainly no match for a dedicated multimedia authoring suite, Video IN still represents a decent value in helping you get started with software motion video recording and editing.

Included with Video IN is the capacity to control certain videodisc players. Specifically, you can control, from within the Video IN Recorder (see below), the following commercial Pioneer videodisc models:

✘ LD-V4200

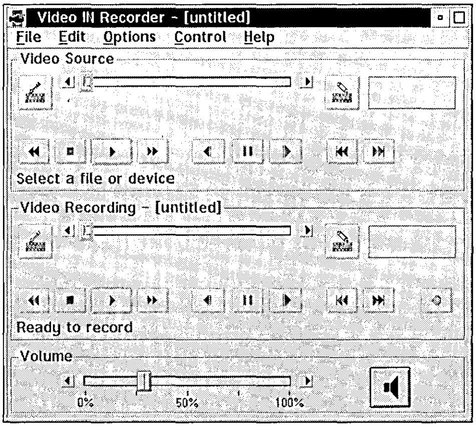
✘ LD-V4300D

✘ LD-V4400

✘ LD-V8000

The significance of this laserdisc support is that you can do *frame-step recording*, in which a high-quality image is captured on a frame-by-frame basis. Frame-step recording provides the highest-quality movie images and the highest frame rates, since the system can take the time necessary to capture each image. Contrast this with *real-time recording*, in which the system captures images, on a continuous basis, as best as possible; system load and the capabilities of the capture hardware limit the number of frames that can be contiguously captured using this method. If you don't have the configuration for frame-step recording, don't be discouraged; real-time video recording using Video IN provides excellent results in either case.

The workhorse program of this package is the Video IN Recorder or SWVR.EXE shown in Figure 13-17. With the Video IN Recorder, you can easily record a new SMV movie, using a variety of available compression/decompression (codec) algorithms such as Intel's Indeo and IBM's own Ultimotion. The Ultimotion video format provides for superior data reduction (up to 18:1), while maintaining excellent picture quality and smooth frame rates. In addition, you can use the Video IN Recorder to perform basic editing features such as marking, cutting, copying, and pasting, as well as creating a series of still image bitmaps from your video input source.



Select a file or device

Ready to record

Figure 13-17. The Video IN Recorder Application

Although the idea of creating software motion video files is a daunting one, it really could not be any easier to record SMV movies than with the Video IN Recorder. In fact, the trickiest part is getting the hardware installed and ensuring that you have enough hard disk space to handle your video production! Table 13-2 shows the resources you will need to record an uncompressed .AVI file with good audio and video.

Table 13-2. Approximate Disk Space Requirements for Uncompressed Video

.AVI Component	Characteristics	Bytes/Second	Bytes/Minute
Audio	16-bit, 11.025 KHz, Monophonic	22,050	1,323,000
Video	15 frames/sec, "64,000" color, 320x240 pixels	2,304,000	138,240,000
Total		2,326,050	139,563,000

Of course, Table 13-2 is only for uncompressed video capture; in reality, your disk space requirements may be much lower for each video you record. Besides, if you've ever used a video camera for home movies, you'll understand that 60 seconds of video can seem to take forever, depending on the subject matter!

Resource requirements aside, recording a video using a video capture adapter is quite straightforward:

1 Open the **Video IN Recorder.**

2 Select **File New** to create the file into which you will record.

3 Use **Options Device selections...** to select the desired video capture device.

4 Select the type of recording you wish to perform, using **Options Record Setup.** In the resulting **Record Setup** settings dialog box, select the audio and video parameters according to your preferences.

5 Press the **Record** button in the Video Recording section of the application and start the playback of your source. Press the **Stop** button when desired.

6 Mark, cut, copy, and paste as desired until you have the desired video clip.

7 Save the file to the desired directory, using **File Save as....**

That's all there is to it!

Another feature of the Video IN Recorder is its capacity to record from one .AVI file to another. With this capability, you can take portions of an existing SMV movie and use it to build another file. In addition, when recording from a file, you can set the codec of the new file to one *different* from the source movie. This is especially useful in demonstrating the advantages of the Ultimotion codec. Unfortunately, the Ultimotion codec has not been adopted as quickly as one would hope; however, this patented algorithm has a number of advantages. One advantage is Ultimotion's ability to provide excellent compression with minimal loss of picture quality. To see this, you may wish to utilize the "record from file" capability of the Video IN Recorder to convert an existing Indeo .AVI into an Ultimotion .AVI file. To do this, follow these simple steps:

1 Open the **Video IN Recorder** application.

2 Select **File New** to create the file into which you will record.

3 Select **Options Record setup Frame Step**. Under the Quality tab of the **Record Setup** settings dialog box, select **Ultimotion Asymmetrical Compression** in the Compression Type list box.

4 Load the Indeo source file, using **File Open Source File.**

5 Press the **Record** button. ◉

These steps are all that is required to record from one file to another. After the recording is completed, you can play either the original or the new file alternately and view the results for yourself.

The purpose of this demonstration is not to "pick on" the Indeo codec format, which is fine in its own right. The point is that Ultimotion is also very good, and the record from file feature can be very handy.

Note

Moreover, note that you can also copy from the Ultimotion format to a file using an Indeo codec; in fact, you can use any codec that has been developed for OS/2 Multimedia (see the section entitled Additional Multimedia Support and Applications for information on other codecs). Bear in mind that all codecs are "lossy"; that is, they compress the video streams, but some of the original data is lost. Going from one "lossy" codec to another may further degrade the picture a bit. However, the degradation is fairly small, given the quality of the codecs that exist today.

Tip

Here are some more hints involving the Video IN Recorder:

- When recording, make sure your volume is set to the appropriate level, which is probably 100 percent. When recording a video, the incoming audio is adjusted according to the current setting of the volume slider. If you have turned this setting down to achieve a comfortable listening level for video playback, the resulting volume of a recorded video may be too low.

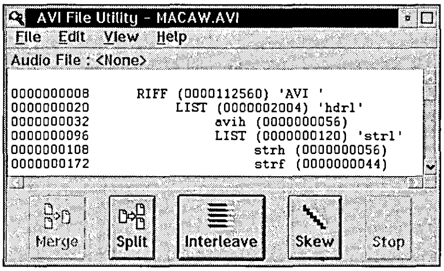
- It is possible to perform frame-step recording using the real-time compression algorithms. The resulting picture will appear "warmer," that is, it will appear less "blocky." However, a small bug in the system may result in a high data rate requirement (about 500 KB/sec). This bug is being addressed by the OS/2 Multimedia video team.

- Refer to the section entitled Additional Multimedia Support and Applications for an excellent Video IN user's guide.

Another utility that is provided with the Video IN product is called the AVI File Utility (AVFU.EXE). This simple program is shown in Figure 13-18 and gives you the following capabilities:

- ✗ You can obtain technical information about a given .AVI file.
- ✗ You can split off the audio portion of the video to create a silent movie.
- ✗ You can add audio to a silent movie (no audio can be present in the file prior to attempting this). This capability is particularly helpful if you want to replace the soundtrack of a movie with a different audio selection. Simply remove the existing audio and add the desired data.
- ✗ You can adjust the audio-video "interleave," which will, in many cases, improve the synchronization between audio and video. This capability is especially important if you have videos of "talking heads," where it can be quite bothersome if the speaker's lips do not match the accompanying audio.

Figure 13-18.
The AVI File Utility



Audio File : <None>

```
0000000008      RIFF (0000112560) 'AVI '
0000000020           LIST (0000002004) 'hdrl'
0000000032                avih (0000000056)
0000000096                LIST (0000000120) 'strl'
0000000108                     strh (0000000056)
0000000172                     strf (0000000044)
```

Let's say you want to remove the audio portion of the MACAW.AVI file shipped with OS/2 Warp. (There are more than a few OS/2 Multimedia developers who, after months of testing using this video, would also like to shoot the parrot with something other than a camera!) Then, you want to add another bit of audio, for instance, the JET.WAV file in \MMOS2\SOUNDS. Assuming you have already installed the Video IN support from the OS/2 Warp BonusPak, follow these steps:

1. Open the AVI File Utility in the Multimedia folder.
2. Load the \MOVIES\MACAW.AVI file by selecting **File Open.**
3. Select the **Split** button to split off the audio; if you wish, you can save this audio by selecting **File Save Audio File....**

4 Select the desired .WAV file to be merged, via **File Open Audio File....** You will be asked whether you would like to save or discard the audio that you earlier split off from the movie.

5 Select the **Merge** button to add the selected audio to the video file.

6 Save the file as whatever name you wish and close the application.

You can now play the file using the Digital Video application.

Uncensored

Adding the Video IN utilities to your system will not make you an overnight multimedia authoring sensation — that is not the intent. The goal of this support is to enable your system for high-quality video capture, provide easy-to-use user interfaces that perform basic functions and demonstrate the capabilities of OS/2 Multimedia in recording SMV, and to stimulate an interest in using OS/2 as a multimedia authoring platform.

Software motion video capture was, until now, the exclusive domain of professional content creators who possessed the requisite mountains of hardware and software necessary to create such movies. Now, as consumer machines provide a powerful but affordable level of technology, OS/2 provides the capability to turn home movies on your computer into reality. Moreover, OS/2 Multimedia does this with solid reliability and without requiring you to close all your applications in order to manipulate your digital video.

Developer's Desktop

In fact, the examples listed here were performed while other operations (including the writing of this chapter) were being performed!

Other goodies!

In the past, it has been said by many experts in the field, from Nicholas Negroponte at the MIT Media Lab to Bill Malone of Tele-Communications Inc., that in the battle of technology vs. content, content wins. In that spirit, no tour of the BonusPak would be complete without mention of OS/2's game attempt to provide additional content.

Beside the venerable MACAW.AVI in your \MMOS2\MOVIES directory and the variety of wave audio and MIDI files found in \MMOS2\SOUNDS, the BonusPak CD-ROM provides two additional directories for your entertainment:

✗ An IMAGES directory, containing files in a variety of image formats, including PCX, Targa, TIFF, and GIF.

- A MOVIES directory, containing two other directories: SNGLSPIN, in which there are two files, each of which is a fun commercial parody by one of our senior planners, who also happened to have a knack for writing (and starring in!) these productions. The directory is named SNGLSPIN because the data rate required to play these movies with acceptable quality is low enough so that the movie can be played directly off of a single-spin CD-ROM.

  The other directory, named DBLSPIN, contains files that are of slightly higher quality but require at least a double-spin CD-ROM device in order to provide the audio-video data at an acceptable rate. Of course, you can always copy any of these files to your hard disk, which is a much faster device (in terms of delivering file data to the system when requested).

  Uncensored

  The DBLSPIN directory contains two Autodesk Animator .FLC files, which show some beautiful spinning OS/2 logos; associated with these files is an image file containing a "screen shot" of one of the frames of this animation. The .AVI files in the DBLSPIN directory were created by Brad Baron, a member of the OS/2 Service and Support team. Brad took it upon himself to take videos of various underwater locations in the South Florida area, capture the videos, and get them on the CD-ROM for your enjoyment. Ah, the rigors of OS/2 in the tropics!

On-line information

Various on-line vehicles exist to help you understand more about the various aspects of OS/2 Multimedia. The following areas contain many tips and techniques for better utilizing the applications that are provided with OS/2 Warp:

- In the Information folder, there is a Multimedia object (\OS2\BOOK\MULTIMBK.INF on the boot drive), which contains help on various topics, such as playing wave audio, MIDI and video files, using the Setup utility, and installing WIN-OS/2 sound support.
- The Tutorial application, which provides clear, step-by-step instructions on how to use the most popular OS/2 Multimedia features. Simply open the Tutorial object and select the button entitled **Multimedia and Games**.
- On-line help for the various multimedia applications. In general, the on-line help information is very extensive, and provides a wealth of tips. Of course, some of the help panels provide only a small amount of extra information (for example,

help for the Play button in the media player) because not much additional information is required. On balance, however, selecting **Help General help** from one of the OS/2 Multimedia applications will provide you with a very good starting point from which you can learn the features of the particular program.

At Your CMD: Multimedia in REXX

REXX is a great batch file command language, whose popularity increases by the year. Far more powerful than the old .BAT file commands (which are still available in non-REXX OS/2 .CMD files), REXX provides a simple way to perform powerful operations in OS/2, from creating various Workplace Shell folders and program objects to automating repetitive tasks normally performed on an OS/2 command line. It is only natural, then, to enhance the power of REXX by adding multimedia extensions.

REXX Multimedia in OS/2 Warp consists of two command files, PLAY.CMD and RECORD.CMD, as well as REXX extensions and documentation that allow you to write your own REXX file to control multimedia devices and files. The Multimedia With REXX on-line information provides an introduction to the Media Control Interface (MCI) string commands used to manipulate multimedia objects; it also describes the supplied .CMD files and how to send MCI string commands to the system and interpret the response.

PLAY.CMD and RECORD.CMD

The PLAY.CMD REXX file, located in the \MMOS2 directory, allows you to play any supported, playable multimedia device or file, including wave audio, MIDI, software motion video, or compact discs. For example, if you were to add support for additional files or devices, then the PLAY.CMD, like the Media Player, would be able to play it. As always, there is an exception to this general rule in that .AVS files (hardware assisted video files played using an Intel Action Media II adapter) cannot be played via PLAY.CMD; however, this support is not widely prevalent.

The PLAY command has the following keyword syntax:

```
PLAY FILE=filename DEV=devicename TIMEFMT=timeformat FROM=fromposition
TO=toposition
```

The Multimedia With REXX information provides a decent description of the command line parameters, but the following points are worth noting:

- All keywords are optional, except as noted below.
- For devices that use files (that is, sound cards), FILE=filename (where filename is the appropriate multimedia file) must be specified. In this case, DEV=devname is optional, but could be specified. For example, if you had two sound cards in your system, you could specify DEV=waveaudio02 to ensure that the file is played on the second installed adapter.
- For devices that do not use files (that is, a CD-ROM device playing compact disc digital audio), DEV=devname (where devname is the appropriate OS/2 Multimedia device name, such as cdaudio) must be specified.
- The TIMEFMT= keyword is mentioned in the documentation, but could be clarified a bit. This keyword is short for "time format," and allows you to specify the unit of time that the system should use when manipulating the object. The default time format used to control an OS/2 Multimedia device is somewhat arcane. The unit is called the MMTIME, and it equals 1/3000 of a second.

  This measure is admittedly rather strange, but when you consider that, in the video industry, video is displayed at 30 frames per second (fps), then 1 MMTIME would equal $^{1}/_{100}$ of a frame. It's not much of an explanation, but at least it seems more like a round number!

  Other time formats depend upon the type of multimedia object being manipulated. For instance, the time format for a wave audio device, could be set to milliseconds, or just ms. A CD audio device can utilize a time format of tmsf (for tracks:minutes:seconds:frames), while a digital video file can use a time format of hmsf (for hours:minutes:seconds:frames).
- The FROM= and TO= keywords allow you to specify the starting and ending times, respectively, of the object being played. If specified, the fromposition and toposition values must be in the appropriate time format.

Here's a fun example of using a different time format, which also shows the power of OS/2 Multimedia in managing different "open instances" of a given multimedia device. This example is similar to the one given in the section entitled The CD Player, where two players were operating at the same time, each managing a different section of the compact disc. This time, you replace one of the CD Players with an invocation of the PLAY command. Assuming you have inserted a music CD into your CD-ROM drive, which has a 4th track at least 40 seconds long, follow these steps:

1 Open the **Compact Disc** application.

2 Start playing the CD.

3 Open an OS/2 window.

4 In the OS/2 window, enter the following command:

```
play dev=cdaudio timefmt=tmsf from=4:00:30 to=4:00:40
```

This PLAY command sets the time format to tracks:minutes:seconds:frames, and instructs OS/2 Multimedia to play the fourth track, from the 30th to the 40th second. You should hear the CD Player be suspended, the fourth track playing for 10 seconds, then the CD Player resuming.

If you wish to play an .AVI file via PLAY.CMD, you must first invoke the PMREXX program. SMV file playback requires that PLAY.CMD run in a Presentation Manager environment.

The RECORD command performs a simple recording operation, using the default wave audio device (your sound card). The microphone input your sound adapter is set as the input device. All you need to do is enter

```
RECORD filename
```

where filename is the appropriate name of the resulting wave audio file.

Note

Digital audio files all have at least three characteristics, which can be queried or set. The higher a given value is, the more hard disk space is required to store the digitized audio data:

1 The sample rate, which is the number of times (per second) that the audio hardware measures an incoming audio signal and digitizes it to some value. Typical values include 11025, 22050, and 44100 (the latter being the value used for most compact discs).

2 The sample size, which is the number of digital bits used to represent the digitized value. Typical values are 8 and 16.

3 The number of channels, which can be 1 (monophonic) or 2 (stereo).

If you are feeling adventurous, you may want to experiment with the RECORD command, as follows:

1 Make a copy of the RECORD.CMD file as a backup.

2 Insert the following commands at around line 65 of RECORD.CMD (U.S. version):

```
MacRC = SendString("set rexxalias bitspersample 16 wait")
if MacRC <> 0 then signal ErrExit
MacRC = SendString("set rexxalias samplespersecond 22050 wait")
if MacRC <> 0 then signal ErrExit
```

If you have a Creative Labs Sound Blaster 16 or AWE32 sound card, which supports certain types of audio compression, and you have installed the Multimedia Viewer audio extensions, then you might also add the following lines after the preceding ones:

```
MacRC = SendString("set rexxalias format tag mulaw wait")
if MacRC <> 0 then signal ErrExit
```

3 Save the RECORD.CMD file and run it from an OS/2 Command Prompt, following the instructions.

After inserting the RECORD command, you will have explicitly set the wave audio device to use 16 bits when digitizing each of its 22,050 samples per second, and the resulting data will be compressed using a format called "mulaw." Try it and see if it works for you. Remember, make a copy of your original RECORD.CMD!

Programming REXX Multimedia

The REXX Multimedia extensions are fairly powerful and the on-line documentation does a decent job of outlining some of the basic concepts and commands. However, there are far more commands (or, more specifically, device-specific keywords for the commands described) than just the ones shown in the documentation. For a complete list of possible Media Control Interface string commands, you should order the OS/2 Warp Multimedia Programming Reference, document number G25H-7114, from IBM Publications (1-800-879-2755).

There are a few additional points that are not described in the various documentation on the subject of REXX Multimedia.

To help illustrate the power of REXX Multimedia, a command file, VIDBOUNC.CMD, is provided on the CD-ROM accompanying this book. VIDBOUNC is a fun little REXX file that has the following command line syntax (the brackets indicate optional parameters and are not included in the command):

```
pmrexx VIDBOUNC [filename] [v=XX]
```

where filename is a particular .AVI file you would like to play (the default is the beloved/dreaded MACAW.AVI file), and v=XX is an indication to set the volume to XX%.

Once invoked, VIDBOUNC will ask you to verify the resolution of your screen; if your resolution is not 1024 × 768, then you can change it. Refer to the System object in the System Setup folder for information on your display's screen dimensions. VIDBOUNC will then proceed to play the SMV movie 6 times: once in each corner and centered on the screen, at its normal size and playback rate, and finally centered on the screen, but "pel-doubled" (that is, the horizontal and video dimensions are twice their normal value), and played quickly. Playing the video quickly will mute the sound, so it will be a fast, silent movie.

For readers of a slightly more technical bent, the VIDBOUNC.CMD file demonstrates the following important, but obscure (or even invisible) points about REXX Multimedia support in OS/2:

- To play an SMV file via REXX, you must invoke the .CMD file via the PMREXX program, which runs the command file in an OS/2 Presentation Manager environment. This is required by the OS/2 Multimedia system when playing SMV files.
- The following line opens the selected file (SendMCICommand is a local routine that calls the mciRxSendString REXX function):

  ```
  cmdRC = SendMCICommand('OPEN' strip(filename) 'ALIAS' alias 'WAIT')
  ```

- Three points are worth noting about this command:

  1. OS/2 Multimedia will handle the recognition of the file and will determine which multimedia device to open on behalf of the file.
  2. An ALIAS keyword has been specified, so that for future commands an alias string can be specified to identify this open instance of a digital video device, as opposed to specifying the file name each time.

3 OS/2 Multimedia provides for powerful multimedia device management, which allows for device sharing.

Normally, an application would allow a given multimedia device to be shared via the SHAREABLE keyword. However, for various development reasons, OS/2 Multimedia does not provide support for sharing devices at the REXX level because it is difficult for a REXX file to handle multiple "threads" of execution at once; a REXX program generally processes instructions sequentially, with exceptions occurring only for situations such as a programming error or when the user presses **Ctrl-Break**. The SHAREABLE keyword is therefore *not* included in the OPEN command in a REXX file, which results in the desired device being opened for exclusive use by the REXX session. Other applications using the device will be suspended until the REXX session releases the device, usually via the CLOSE command.

✘ Related to the discussion of exclusive device ownership is the concept of notification. A regular OS/2 application can receive OS/2 Multimedia-specific notifications while it performs other tasks. One such notification would be an indication that the program is losing the use of the device. Another example of notification is the specification of the NOTIFY keyword on the command, most often, done on the PLAY command. In REXX, however, no facility currently exists for providing multimedia notifications to the command file. Therefore, the WAIT command is used almost all the time, as demonstrated by the following example from VIDBOUNC:

```
call SendMCICommand 'PLAY' alias 'WAIT'
```

✘ In some cases, the return code from a mciRxSendString command will be greater than 65536 which, when displayed, can be quite confusing, since most OS/2 Multimedia errors start at 5000 or so.

For some MCI command errors, a value specifying the open instance of the device is added to the return code. In programming terms, the device ID is placed in the high-order two bytes of a four-byte return value, with the low-order two bytes containing the actual error code. The following REXX code, located in VIDBOUNC.CMD's SendMCICommand routine, handles this case by converting the returned value to hexadecimal, retaining only the right most 4 hex digits and converting the value back to decimal:

```
if rc > 65536 then rc = X2D(right(D2X(rc),4))
```

- ✗ The size of an SMV movie, as well as its corresponding default video window, can be determined quite easily. First, the following STATUS commands, starting at line 139 of VIDBOUNC.CMD, allow the program to understand the actual width and height, respectively, of the SMV movie:

```
if SendMCICommand('STATUS' alias 'HORIZONTAL VIDEO EXTENT WAIT') = 0 then
  vidx = mciResults
else vidx = 0
if SendMCICommand('STATUS' alias 'VERTICAL VIDEO EXTENT WAIT') = 0 then
  vidy = mciResults
```

Next, the actual dimensions of the default movie can be quickly determined by using the WHERE command. Here is an excerpt from line 181 of VIDBOUNC.CMD:

```
if SendMCICommand('WHERE' alias 'WINDOW WAIT') = 0 then
do
 parse value mciResults with winx winy winx2 winy2
 fudgex = winx2 - winx - vidx
 fudgey = winy2 - winy - vidy
end
```

The WHERE command returns the starting x, starting y, ending x, and ending y coordinates (winx, winy, winx2, and winy2). These values can then be used to determine "fudge factors" for each x and y dimension, which is actually the amount of space required for the default window border and controls. To double the size of the window, then, you can simply double each video extent (vidx and vidy), add the fudge factors (which remain constant) and center the resulting values on the screen, using the PUT command. Here's the related REXX code from VIDBOUNC, starting at line 208:

```
    x=(screenx-(vidx*2)-fudgex)/2  /* Starting x = width doubled, centered  */
    x2= x + vidx*2 + fudgex        /* Ending x = Starting x + double-width  */
                                   /* + borders                             */
    y=(screeny-(vidy*2)-fudgey)/2  /* Starting y = height doubled, centered */
    y2= y + vidy*2 + fudgey        /* Ending y = Starting y + double-width  */
                                   /* + borders                             */
    if SendMCICommand('PUT' alias 'WINDOW AT' x y x2 y2 'MOVE SIZE
WAIT') = 0 then
:
```

If you think about it, this REXX file is roughly 130 lines of code. However, you are manipulating a multimedia file type in a way that, even today, would generally require *thousands* of lines of real programming code to achieve the same effect. Clearly, the power of OS/2 and its associated Multimedia and REXX support bring a whole new level of excitement and utility to the personal computer!

Fun Stuff Out There

Out in the real world, OS/2 Multimedia, despite its power and flexibility, remains very much an "unsung hero" when compared with the stupendous number of DOS and Windows multimedia applications. Happily, this situation is changing for the better, as more people become familiar with OS/2's multimedia capabilities. Similarly, application developers, who have long understood the advantages that OS/2 provides to mission-critical or demanding PC environments, are beginning to realize the benefits of writing consumer-oriented multimedia applications on an OS/2 platform.

The next few sections describe just a few additional "goodies" that are readily available through various outlets.

OS/2 games

One particularly impressive OS/2 game is Galactic Civilizations which, by all accounts, is a multi-threaded, multimedia-enabled, thoroughly engrossing strategy game. The goal of "GalCiv" is to colonize as much of outer space as possible; along the way, the player is required to make all kinds of decisions, ranging from whether to spend money on military equipment, social programs, or additional colonization, to determining which aliens should be approached as allies. This game appeared in an Internet list called the "Internet Top 100," which unscientifically keeps track of the most popular games. This list is watched closely by game developers. Galactic Civilizations made it to the number 1 spot in just a few weeks, eclipsing such impressive programs as Id's Doom game and LucasArts Entertainment Company's Rebel Assault.

For further information about obtaining Galactic Civilizations, as well as a number of other interesting, multimedia-enabled applications, contact Advanced Idea Machines, at 801-572-6353.

In addition to games for sale, OS/2 multimedia-enabled shareware is becoming increasingly popular. A great place to browse, if you have Internet access, is the FTP site ftp-os2.cdrom.com (login as anonymous). This site has numerous applications, including fun shareware programs such as ROIDS (a throwback to the old Asteroids game, with fine performance and digital audio support).

Additional multimedia support and applications

A wealth of information about OS/2 Multimedia is available from a variety of sources. Here are just a few miscellaneous items worth investigating or obtaining:

- In the BonusPak, the Video IN file README.VIN describes a User's Guide for the Video IN applications. This guide, which was not available for shipping on OS/2 Warp, contains a complete set of information about the use of the Video IN utilities. The document also has a very nice table of supported video capture cards which, though now a bit dated, still contains useful information for comparison. To obtain the Video IN User's guide, refer to the VIDEOIN\README.VIN file on the appropriate OS/2 BonusPak CD-ROM or diskette.

  Be sure to contact your favorite hardware vendor and inquire about the existence of OS/2 support for your preferred multimedia hardware.

- One of the most common user complaints is OS/2 Multimedia's inability to play certain software motion video .AVI files. The usual cause of this problem is that the video data within the file has been compressed using an algorithm that is not recognized by the base OS/2 Multimedia support. Two popular, but unsupported, codecs are Microsoft's Video 1 and SuperMac's Cinepak compression schemes.

  The good news, however, is that an enterprising German student, Andreas Portele, has created an OS/2-compliant set of codecs for these video formats and has placed them on the Internet. To obtain these codecs, FTP to ftp-os2.cdrom.com as anonymous and switch to the pub/os2/incoming directory, and do a "binary get" of the file apcodec3.zip (the location may change in the future, so you may have to look around this FTP site if you do not find the file immediately). You can also point your World Wide Web browser to **http://www.rbg.informatki.th-darmstadt.de/~portele**.

Finally, a number of forums exists in which OS/2 Multimedia is discussed, including:

- The OS/2 BBS; call IBM for details.
- The CompuServe Information Service. You can ask questions in various sections of the OS2USER, OS2SUPPORT (preferred), or OS2DF1 (the developer's forum, where various OS/2 Multimedia developers monitor messages in section 12 - MMPM/2).
- The Internet news group comp.os.os2.multimedia.

Paul Rogers is an Advisory Programmer with IBM Boca Raton. Paul has been involved in OS/2 Multimedia development since 1991. He has been with IBM since 1979, and has worked on OS/2 development projects since 1986. Paul worked on the keyboard device driver for OS/2 Version 1.0.

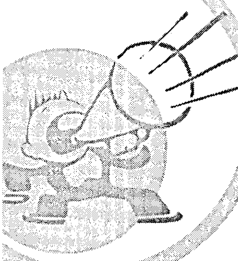
Chapter 14

Printing

Follow the Paper Trail!

This chapter will describe the OS/2 Warp printing subsystem, including a detailed description of its user interface, the Printer object. The Printer object is used to control your printer. It allows you to customize the settings for your printer and manage your print jobs. You'll learn how to create a Printer object, send jobs to the printer, and manage those print jobs.

Introduction to the Printing Subsystem



When you print a data file, it is sent to a Printer object and becomes a print *job*. You can think of a *Printer object* as a queue which displays an ordered list of jobs. *Network Printer objects* can also be created, allowing you to send jobs to devices attached to other systems.

The *spooler* is the underlying component which manages the printing of both local and remote jobs. It is the spooler's responsibility to coordinate the sharing of output devices (printers, plotters, fax machines, and so on) between applications. Use of the spooler ensures that jobs from different applications will not accidentally be intermixed, which can ruin the output. For example, if the OS/2 Warp spooler is running, and you print a job under WIN-OS/2 and a job under DOS, both print jobs will be handled by the OS/2 Warp spooler. A major advantage of the spooler is that you can still use your applications while printing a job, which is not possible using other operating systems. The Spooler object, located in the System Setup folder (see Chapter 7), allows you to customize settings, such as where spooled jobs are stored on the fixed disk, and how quickly spooled jobs are sent to the printer.

All other printing subsystem objects are accessed through the Printer object. A Printer object uses a *queue driver* to perform any hardware-independent processing on the print jobs and puts them into a format that the printer driver can understand. *Printer drivers* provide hardware-specific information that is needed in order to use the printer. The *port driver* provides support for the transmission of data through the serial or parallel port connected to the device.

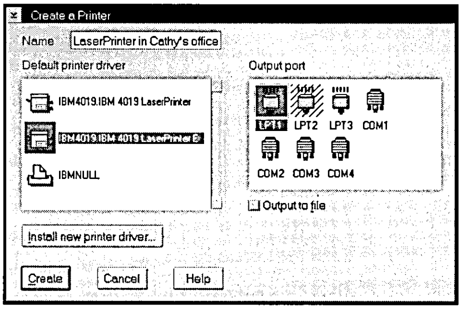
Creating a Printer Object

In this section you will learn how to create a local or network Printer object by dragging a Printer template from the Templates folder. Alternate methods of creating Printer objects will also be discussed. At a minimum, you will need to select a printer driver and a port to create a Printer object.

Dragging a Printer template from the Templates folder

The Templates folder contains a template called Printer. The recommended way to create a Printer object is to drag this template from the Templates folder into another folder, such as the Desktop. When you drag a Printer template out of the Templates folder, the Create a Printer dialog will appear (see Figure 14-1). On this dialog, you will see the following:

- ✘ **Name** — This is your Printer object's title. You can use the default name which appears, or enter any name up to a maximum of 47 characters.
- ✘ **Default printer driver** — You will see a list of all the printer drivers currently installed. Select a printer driver that supports your model of printer. If you don't see your printer driver in the list, click on the **Install new printer driver** push button (or select Install from an existing printer driver's pop-up menu) and the Install New Printer Driver dialog (see Figure 14-2) will appear. You may also delete a printer driver object from this list by selecting Delete from the driver's pop-up menu.



Name: LaserPrinter in Cathy's office

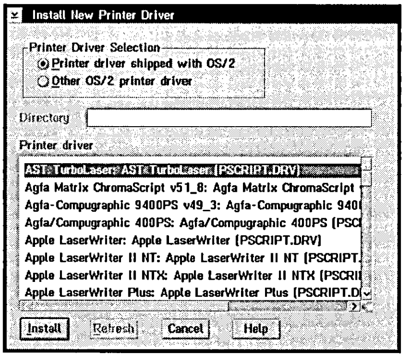
IBM4019.IBM 4019 LaserPrinter

IBM4019.IBM 4019 LaserPrinter E

IBMNULL

□ Output to file

Figure 14-1. Create a Printer Dialog



● Printer driver shipped with OS/2

○ Other OS/2 printer driver

AST TurboLaser: AST TurboLaser (PSCRIPT.DRV)
Agfa Matrix ChromaScript v51_8: Agfa Matrix ChromaScript
Agfa-Compugraphic 9400PS v49_3: Agfa-Compugraphic 940
Agfa/Compugraphic 400PS: Agfa/Compugraphic 400PS (PSC
Apple LaserWriter: Apple LaserWriter (PSCRIPT.DRV)
Apple LaserWriter II NT: Apple LaserWriter II NT (PSCRIPT.
Apple LaserWriter II NTX: Apple LaserWriter II NTX (PSCRI
Apple LaserWriter Plus: Apple LaserWriter Plus (PSCRIPT.D

Figure 14-2. Install New Printer Driver Dialog

- **Output port** — Select the port that your printer is using. You may select more than one port for printer pooling (see the section on Printer pooling, later in this chapter). To install a port not in the list, select **Install** from the pop-up menu and the Install New Port dialog will appear. After typing in the path where the new port drivers are located, press the **Refresh** push button to see all the drivers in that path. Select a port driver in the Output port list box and press the **Install** push button to install the driver. You may also delete a port object from this list by selecting Delete from the port's pop-up menu. To change the port's settings, select Settings (see Figures 14-15 and 14-16, later in this chapter) from the pop-up menu. You can select Redirection on the port's pop-up menu to redirect the output to another port (see the section on Redirection, later in this chapter).

- **Output to file checkbox** — If you want print jobs sent to this Printer object to be stored in a file on your fixed disk, select this option. When you send a print job, a dialog will be displayed asking you to enter a filename. A printer-specific file (see the Understanding print job types section) is created and can be copied to another system and printed if that system uses the same type of printer.

- ✗ **Install new printer driver** — Select this push button to display the Install New Printer Driver dialog (see Figure 14-2). You can install printer drivers shipped with OS/2, or other OS/2 printer drivers from any directory. Any new printer drivers installed through the Install New Printer Driver dialog will be displayed in the Default printer driver list after you return from the dialog.

When you have filled in all of the above information, press the **Create** push button to complete the installation of your Printer object. After installing a Printer object, if you are using OS/2 Warp with WIN-OS/2 and there is an equivalent WIN-OS/2 printer driver in \OS2\MDOS\WINOS2\DRVMAP.INF (or in \WINDOWS\SYSTEM\DRVMAP.INF if you are using OS/2 Warp), you will be asked if you want to create an equivalent WIN-OS/2 printer. After you are prompted for a diskette containing the WIN-OS/2 or Windows printer drivers, the Windows printer driver will be installed and your WIN.INI file will automatically be updated.

After you have successfully created your Printer object, the icon you see in your folder will depend on the type of printer driver you are using. For example, an IBM4019 LaserPrinter will have a laser printer icon, and an IBM4201 Proprinter will have a dot matrix printer icon.

Installing a new printer driver

The Install New Printer Driver dialog (see Figure 14-2) contains the following:

- ✗ **Printer Driver Selection** — Click on the **Printer driver shipped with OS/2** radio button to display the list of printer drivers that are included with OS/2. The list of printer drivers is read from the \OS2\INSTALL\PRDESC.LST file. Otherwise, select the **Other OS/2 printer driver** radio button if you received a driver from another source, such as the manufacturer of your printer or a bulletin board system. Selecting **Other OS/2 printer driver** enables you to type the location of the printer driver file (*.DRV) in the Directory field.
- ✗ **Directory** — Type the path where your printer driver file (*.DRV) is located. This field is enabled only when Other OS/2 printer driver is selected. Press the **Refresh** push button to find all printer drivers in that directory.
- ✗ **Printer driver** — If you selected **Printer driver shipped with OS/2**, all printer drivers included with OS/2 will be displayed. Otherwise, if you selected **Other OS/2 printer driver**, specified a directory, and selected **Refresh**, all printer drivers

located in that directory will be displayed. After selecting one or more printer drivers from this list, press the **Install** push button. If the printer driver location is not known, you will be asked to enter the path where the printer driver exists. If another version of the printer driver has already been installed, you will be asked if you want to replace the old printer driver file with the new version. When the install is complete, you will receive a message telling you that it has been successful. Press the **OK** push button to return to the Create a Printer dialog.

Other ways to create a Printer object

Here are some alternate ways to create a Printer object:

- Select a printer on the System Configuration dialog when installing your system (see Chapter 1).
- Select a printer from **Selective Install** in the System Setup folder (see Chapter 1).
- Select the **Create another** or **Copy** option from an existing printer's pop-up menu.
- Drag a printer driver to a folder using the following steps:
  1. Open the OS/2 System folder, then the Drives folder.
  2. Open a folder containing printer drivers (*.DRV files).
  3. Double-click on a printer driver to open an Icon view containing the supported printer models.
  4. Drag a printer driver to a folder to automatically install a printer.

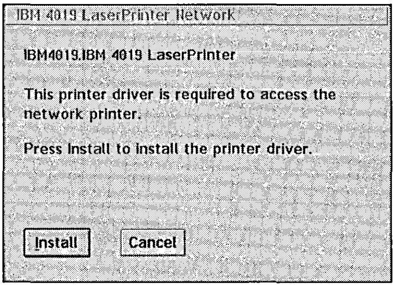
Unsupported Printers

If you don't see a printer driver that supports your printer in the list, you may want to contact the manufacturer of your printer to see if they have an OS/2 printer driver available, or check the bulletin board systems to try to find one. Another choice is to use a printer driver that your printer can emulate, and switch your printer into that mode. Otherwise, you can try using the IBMNULL printer driver.

Creating a Network Printer object

A Network (remote) Printer object represents a printer resource located on another computer or server. When a network designed for OS/2 is installed, the Network folder will appear on your Desktop, and you will see a Network Printer template in your Templates folder. You can drag the Network Printer template from your Templates folder to another folder to create a Network Printer object. You can also access network printers through the Network folder. The Network folder contains network groups; opening a network group folder will show the servers that are available on that network. To view the network resources (printers and shares) that you have access to, open a server's folder.

If the necessary printer driver is not installed locally, you will be prompted to install the driver when you access a network printer for the first time. For example, when you drag a file to a Network Printer object, drag a Network Printer template from the Templates folder, or drag a network printer out of the Network folder (see Figure 14-3).


IBM 4019 LaserPrinter Network

IBM4019.IBM 4019 LaserPrinter

This printer driver is required to access the network printer.

Press Install to install the printer driver.

Figure 14-3. Installing a Network Printer

Using the Printer Object

On a Printer object's pop-up menu, you can open a view to display print job objects. The print job's icon will change depending on the state of the job (see Figure 14-4).

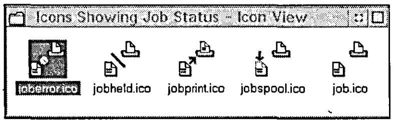
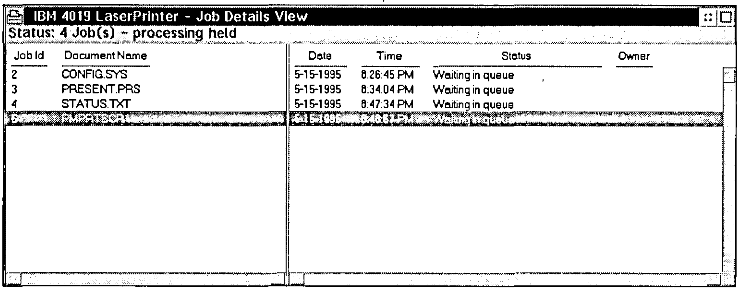


Figure 14-4. Job Status Icons

If you open the Printer object in Details view, you will see the details of each job, including the Job ID, Document Name, Date, Time, Status, and Owner (see Figure 14-5).



IBM 4019 LaserPrinter - Job Details View

Status: 4 Job(s) - processing held

Job Id	Document Name	Date	Time	Status	Owner
2	CONFIG.SYS	5-15-1995	8:26:45 PM	Waiting in queue	
3	PRESENT.PRS	5-15-1995	8:34:04 PM	Waiting in queue	
4	STATUS.TXT	5-15-1995	8:47:34 PM	Waiting in queue	
5	PMPRTSCR	5-15-1995	8:45:57 PM	Waiting in queue	

Figure 14-5. Printer Details View

Both views of the Printer object contain a status line, showing the number of jobs in the queue and the status of the queue (for example, processing held). Besides opening a Printer object in Icon view or Details view, other Printer object-related options on the pop-up menu include:

- ✗ **Settings** — Select Settings to display the Printer object's Settings Notebook (see Figures 14-9 through 14-18, later in this chapter).
- ✗ **Change status** — You can hold print jobs and print them at a later time by changing the printer's status to Hold. To print the held jobs, select Release. This option is useful when you need to take your printer off-line for a period of time.
- ✗ **Set default** — You can use this option to specify your default printer. When you select the Print option on a data file's pop-up menu, the file will be printed on the default printer. You can also set the application default printer when creating a Printer object by using REXX and the APPDEFAULT= setup string.
- ✗ **Delete all jobs** — You will see this option if this Printer object contains at least one print job. This allows you to delete all print jobs at once.

There are four menu items that can be added to a Network Printer object's pop-up menu:

- ✗ **Access another** — Accesses another network printer. When you select this item, a dialog will appear asking for information about the network printer you are trying to access. You will need to know the resource name of your network printer and where it is located (network and server name).
- ✗ **Assign port** — Assigns a port to this network printer. All jobs sent to that port will now be sent to the network printer. This option can be used if you have applications that print to a port. Assigning a port to a network printer allows the application's output to be sent to the network printer.

- **Login** — Logs into the network. Use this option if your network requires you to log in to access the network printer.
- **Logout** — Logs out of the network. Use this option for security reasons if you are leaving your computer and don't want another person to use the network printer while you are away.

Print Job objects contain the following pop-up menu items:

- **Open (click on the arrow to the right and select) Job content** — Uses the PICVIEW application to display the contents of the print job.
- **Settings** — Displays the print job's Settings Notebook (see Figures 14-6 through 14-8).
- **Copy** — Prints multiple copies of the print job.
- **Delete** — Deletes the print job.
- **Change status** — Allows you to hold a job or release it to print it at a later time.
- **Print next** — Allows you to change the ordering of your jobs by selecting which one should be the next to print.
- **Start again** — If a job is currently printing, you can start it over again. This option is useful, for instance, if your forms weren't loaded correctly the first time.

Selecting print job options

The Print Job object Settings Notebook contains three pages: printing options, submission data, and queue options.

Printing options page

The Printing options page (see Figure 14-6) contains the following fields:

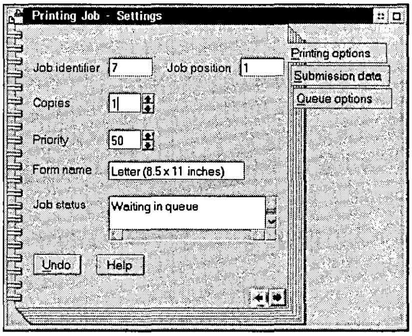
- **Job identifier** — The job ID number.
- **Job position** — The position of the job in the queue.
- **Copies** — The number of copies to be printed. The number of copies can range from 1 to 999.

- **Priority** — The priority of this job. Jobs with higher numbers will print before jobs with lower numbers. Jobs with equal priorities will print in the order in which they were submitted. The default priority is 50, but it can be changed to anywhere from 1 to 99 on local printers. On network printers, the user can specify a priority from 1 to 50; a network administrator can specify a priority from 1 to 99.

- **Form name** — The form that was specified to print on when job was submitted.

If the form name doesn't match the Printer object's printer properties, the Job status field will indicate Forms mismatch, and the job will not print until the printer properties are changed to match the form name. See the section on the Printer Driver page for information on how to change the printer properties.

- **Job status** — The status of the job (printing, waiting in queue, out of paper, forms mismatch, and so on).



Printing Job - Settings

Job identifier: 7
Job position: 1
Copies: 1
Priority: 50
Form name: Letter (8.5 x 11 inches)
Job status: Waiting in queue

Figure 14-6. Printing Job - Printing Options Page

Submission data page

The Submission data page (see Figure 14-7) contains the following fields:

- **Date** — The date the job was submitted.

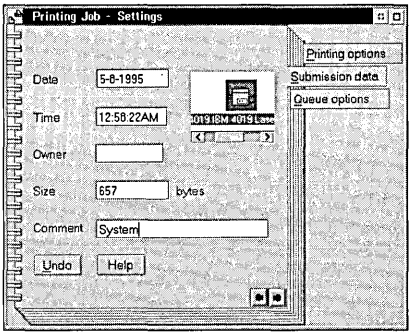
- **Time** — The time the job was submitted.

- **Printer driver object** — The printer driver used to print this job. If you select Settings on the printer driver's pop-up menu, the printer driver's job properties will be displayed.

- **Owner** — The owner of the job.

- **Size** — The size of the job in bytes.
- **Comment** — Any comment relating to the job that has been entered. This is the only field you can change.

You may want to change the comment to "Rough Draft" or "Final Version" so you'll recognize the jobs when you view them through the Printer object.



Date 5-8-1995

Time 12:58:22AM

Owner

Size 657 bytes

Comment System

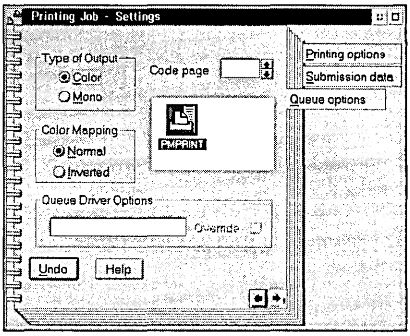
Figure 14-7. Printing Job - Submission Data Page

Queue options page

The Queue options page (see Figure 14-8) will only be displayed if the print job was spooled using standard output data format (see the Understanding print job types section). This is because the settings on the Queue options page are only relevant for metafiles played through the queue driver. The Queue options page contains the following fields:

- **Type of Output** — Select Color or Mono.
- **Code page** — Select your country's code page.
- **Color Mapping** — Select Normal or Inverted.
- **Queue driver object** — The Queue driver to be used for this print job.
- **Queue Driver Options** — You can't change the information in this field, but you can select the **Override** option to ignore the queue driver options. For example, if the field contained **COP=3** to print 3 copies of the job, or **COL=C** to specify color output, select **Override** to ignore the options, and the default options will be used.

Figure 14-8. Printing Job - Queue Options Page



Type of Output
● Color
○ Mono

Code page

Color Mapping
● Normal
○ Inverted

Queue Driver Options
Override □

Printing a job

There are several ways you can print a data file:

- Drag and drop a Data File object onto a Printer object or an open view of a Printer object.
- Select the **Print** option on a data file's pop-up menu. If you haven't set the data file's type to either Plain Text or Printer-specific (on the Type page of the data file's Settings Notebook), you will be asked if the file is Plain Text (standard output data) or Printer-specific (raw data). See the Understanding print job types section for an explanation of both options.
- Print a data file from an application.
- Use the Print Screen key. You can print an item on the Desktop by selecting that item and pressing the **Print Screen** key on your keyboard. The print job will be sent to your default printer. If you place the mouse pointer on the Desktop and press the Print Screen key, you will print the full Desktop. If you press the Print Screen key from an OS/2 full-screen session, your print job will go to LPT1. See the discussion of the System Object in Chapter 7 for information on how to enable or disable the Print Screen key.

- Use the **PRINT** command from an OS/2 or DOS command prompt. By default, the PRINT command prints to the LPT1 port. You can add parameters to the PRINT command to change the default settings:

Parameter	***Function***
/D	Use this option to specify a different printer port. For example, PRINT /D:LPT2 MY.DOC will print MY.DOC on the printer attached to the LPT2 port.
/B	Use this option if your file is binary or contains Ctrl-Z characters. Otherwise, they will be interpreted as end of file characters. For example, PRINT /D:LPT2 /B MY.DOC will print the entire file MY.DOC even if it contains Ctrl-Z characters.
/C	Use this option to cancel the job currently printing on the selected printer. For example, PRINT /D:LPT2 /C will cancel the job currently printing to LPT2 before beginning your print job.
/T	Use this option to cancel all print jobs, including the current print job. For example, PRINT /D:LPT2 /T will cancel all jobs printing to LPT2.

- Use the **COPY** command from the command line. For example, COPY FILE.BIN /B LPT1 copies a binary file to the printer attached to the LPT1 port. COPY FILE COM1 copies a file to a plotter attached to the COM1 port.

Understanding print job types

Print jobs that are held in the Printer object are known as *spool files.* These files consist of actual data and also contain information on how to print the job. OS/2 Warp supports print jobs in two formats:

- **Standard output data** — Standard output data is spooled as a Presentation Manager metafile (in packed binary format). Using the standard format produces smaller files than using raw data, which saves disk space and reduces network traffic when printing to a network printer. The PICVIEW application can view these jobs; PICVIEW is started when you click on the arrow to the right of

Open and then select the Job content option on the job's pop-up menu. For network printers that do not support Presentation Manager, the system will automatically convert the print job to raw data. If you select the Printer-specific format option on the Printer object's Settings Notebook's Queue options page, all standard print jobs printed to that Printer object will be converted to raw data.

- ✗ **Raw data** — Raw data is the actual printer commands used to print the job. Raw data for a PostScript printer consists of actual PostScript commands. Raw data for an HP LaserJet printer consists of actual Printer Command Language (PCL) commands. The System Editor can be used to view the content of a job formatted using raw data. The System Editor is called when you click on the arrow to the right of Open and then select the Job content option on the job's pop-up menu.

Using the Printer Object's Settings Notebook

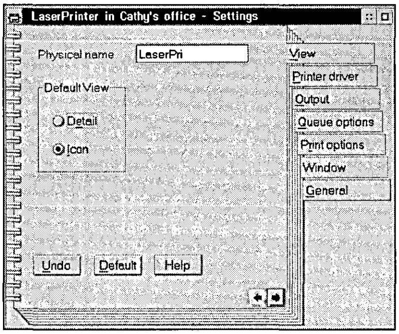
You can use the Printer object's Settings Notebook to select options for either your local printer or a network printer.

Selecting local printer options

The local Printer object's Settings Notebook contains several pages that allow you to configure the printer the way you want it.

View page

The Physical name field (see Figure 14-9) contains an eight-character name that the spooler derives from the print object's title in order to create a directory under the \SPOOL path. For example, if the physical name is IBM4019L, spooled print jobs for this Printer object will be stored in the \SPOOL\IBM4019L directory. When printing from some applications, you may see this eight-character queue name instead of the print object's title. The queue name can also be set if you create a Printer object using REXX and the QUEUENAME= setup string.



LaserPrinter in Cathy's office - Settings

Physical name: LaserPri

Default View

○ Detail

● Icon

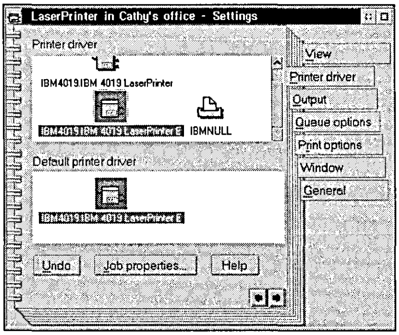
Figure 14-9.
Printer Object - View Page

The View page allows you to change the Printer object's default open view to either Detail or Icon view. Changes on this page take effect when you either close the Settings Notebook or switch to another page in the Notebook. Opening the Printer object in Details view shows the job's ID, name, date and time submitted, and status. Opening the Printer object in Icon view shows job objects, each with an icon representing its state (spooling, waiting, printing, etc.).

Printer driver page

The Printer driver page (see Figure 14-10) contains two fields, the Printer driver container and the Default printer driver container. The Printer driver container shows all printer driver objects that are currently installed on the system. The selected driver is the one assigned to this Printer object.

More than one printer driver can be selected for each Printer object. But only one printer driver can be selected in the Default printer driver container. The printer driver can also be set using REXX and the PRINTDRIVER= setup string.



LaserPrinter in Cathy's office - Settings

Printer driver

IBM4019.IBM 4019 LaserPrinter

IBM4019.IBM 4019 LaserPrinter E

IBMNULL

Default printer driver

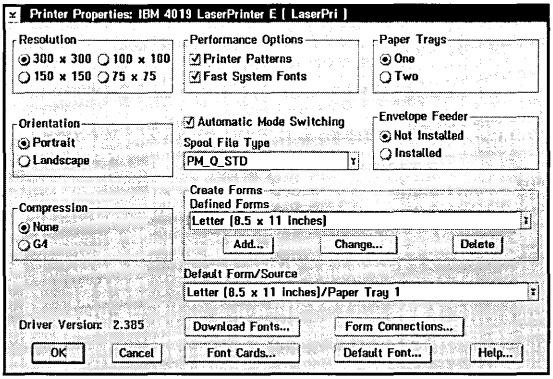
IBM4019.IBM 4019 LaserPrinter E

Figure 14-10.
Printer Object - Printer Driver Page

The pop-up menu of a printer driver object contains:

- ✗ **Settings** — When you select the Settings option on a printer driver object, the printer properties dialog is displayed for that printer driver. The printer properties define the physical printer configuration (see Figure 14-11 for an example of a printer driver's printer properties).

Figure 14-11. Printer Driver Printer Properties


Printer Properties: IBM 4019 LaserPrinter E (LaserPri)

Resolution
● 300 x 300 ○ 100 x 100
○ 150 x 150 ○ 75 x 75

Performance Options
☑ Printer Patterns
☑ Fast System Fonts

Paper Trays
● One
○ Two

Orientation
● Portrait
○ Landscape

☑ Automatic Mode Switching

Spool File Type: PM_Q_STD

Envelope Feeder
● Not Installed
○ Installed

Compression
● None
○ G4

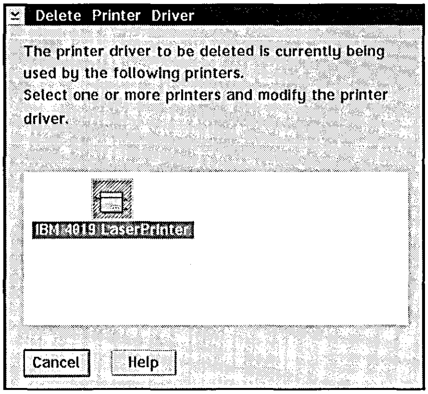
Create Forms
Defined Forms: Letter (8.5 x 11 Inches)

Default Form/Source: Letter (8.5 x 11 Inches)/Paper Tray 1

Driver Version: 2.385

- ✗ **Delete** — Select this option to delete this printer driver object. If any Printer objects (including the one you are currently changing the settings on) have this printer driver selected, you will be prompted with the Delete Printer Driver dialog (see Figure 14-12). This dialog lists all Printer objects that are using the printer driver, giving you a chance to select another printer driver for those printer objects, or to cancel the delete. You can select Settings on the pop-up menu of a printer object and go to the Printer driver page to deselect the printer driver you are trying to delete. Press the **Cancel** push button when you are finished changing the settings. If there aren't any Printer objects using the printer driver and all printer drivers contained in the printer driver file have been deleted, you will also be asked if you want to delete the actual printer driver file (*.DRV) from the fixed disk.

Figure 14-12. Delete Printer Driver Dialog



Delete Printer Driver

The printer driver to be deleted is currently being used by the following printers.
Select one or more printers and modify the printer driver.

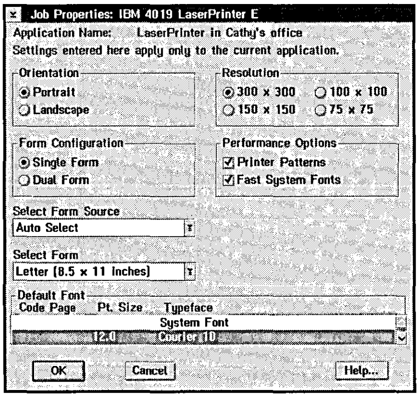
IBM 4019 LaserPrinter

- ✗ **Install** — Select this option to install another printer driver object.

Tip

To change the printer driver's job properties, make sure the driver is selected, and then click on the **Job properties** push button. The job properties define settings based on a print job (see Figure 14-13 for an example of a printer driver's job properties). This option can also be set using REXX and the JOBPROPERTIES= setup string. In order for the changes to take effect, you must switch to another page or close the Settings Notebook.

Figure 14-13. Printer Driver Job Properties



Job Properties: IBM 4019 LaserPrinter E

Application Name: LaserPrinter in Cathy's office
Settings entered here apply only to the current application.

Orientation
● Portrait
○ Landscape

Resolution
● 300 x 300 ○ 100 x 100
○ 150 x 150 ○ 75 x 75

Form Configuration
● Single Form
○ Dual Form

Performance Options
☑ Printer Patterns
☑ Fast System Fonts

Select Form Source
Auto Select

Select Form
Letter (8.5 x 11 inches)

Default Font

Code Page	Pt. Size	Typeface
		System Font
	12.0	Courier 10

OS/2 Warp provides support for these 14 types of printers:

- IBMNULL
- Epson for generic Epson-compatible printers
- Hpdipm for HP DeskJet printers
- IBM42xx for IBM 42xx printers
- IBM52012 for IBM 52012 printers
- IBM52xx for IBM 52xx printers
- IBMPCL3 for printers that support the IBMPCL3 emulation
- IBM4019 for IBM 4019/4029 printers
- IBMPCL5 for printers that support the IBMPCL5 emulation
- LaserJet for LaserJet printers
- PostScript for PostScript printers
- Smgxpget for the PaintJet printers
- Plotters for the plotter devices
- OMNI for Epson, DeskJet and PaintJet printers

Output page

The Output page (see Figure 14-14) allows you to assign a port to your Printer object. The selected port is the one currently assigned to this Printer object. Multiple ports may be selected. The port name can also be set using REXX and the PORTNAME= setup string. If a port has hash marks through it, another Printer object is using that port.

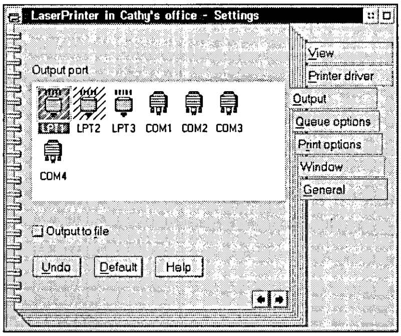


Figure 14-14. Printer Object - Output Page.

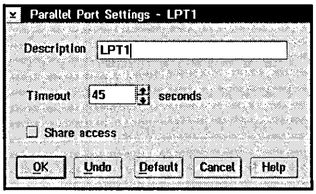
Printer Off-line Message

If you receive an error message telling you that your printer is off-line or out of paper, the problem may be solved by increasing the timeout value. You should also check to make sure you have a good printer cable. If this doesn't solve the problem, try booting your machine with the Installation diskette and then insert Diskette 1. Press **Esc** to get to a command prompt and use the **COPY** or **PRINT** command to print to the port. If you still can't print, and you are printing using interrupts, you may have hardware conflicts (see the section New polling support, later in this chapter). Check any cards (such as a sound card) or devices in your CONFIG.SYS to see what interrupt they are using.

The pop-up menu for a Port Driver object contains:

- **Settings** — Selecting Settings displays the port object's settings. When you select the Settings option on a Port object, a Settings dialog for the port is displayed.
- **Timeout** — For parallel ports (LPT1-LPT3), you can change the timeout value (see Figure 14-15). This is the amount of time that the system waits before reporting a printing delay as an error. The default timeout value is 45 seconds; however, you may need to customize the timeout value if you are using certain printers. For matrix printers, set it to 10 seconds or less. If you are using a PostScript printer and you get an error message telling you to make sure the printer is switched on, online, and loaded with paper, set the timeout value to 120 to 300 seconds or more.

Figure 14-15. Parallel Port Settings Dialog



Parallel Port Settings - LPT1

Description LPT1

Timeout 45 seconds

□ Share access

For serial ports (COM1-COM4), selecting Settings will display the serial port's Settings dialog (see Figure 14-16). The default settings work with most printers or plotters, but if your manufacturer's documentation recommends certain values, you can change the settings. On this dialog, you have the following options:

- ✘ **Timeout** — You can also change the timeout value for COM ports. This is the amount of time that the system waits before reporting a printing delay as an error.
- ✘ **Baud Rate** — The baud rate represents the speed at which information travels over a communication line. The higher the baud rate, the faster information travels. The default baud rate is 9600. Click on the up or down arrows to change the baud rate. Valid baud rates are 110, 150, 300, 600, 1200, 1800, 2400, 3600, 4800, 7200, 9600, 14400, 19200, 38400 and 57600. You must select the same baud rate that your printer uses.
- ✘ **Word Length** — The default word length is 8 bits. This means that both the transmitter and receiver of information interpret each group of 8 bits to represent one character.
- ✘ **Parity** — Select the type of parity checking your printer or plotter requires. The default parity is None. Select a parity other than None if your manufacturer's documentation says the printer ignores parity, but expects it anyway. For example, selecting Odd parity means that every character represented by an even number will have a bit added to it so that it becomes an odd number. The receiver checks to make sure it receives only odd values.
- ✘ **Stop Bits** — The default value is 1. In transmitting data, the stop bit is a signal at the end of a character that notifies the receiver that one character has ended and another is about to begin. If your manufacturer's documentation specifies a different signal length for stop bits, you change the value.
- ✘ **Handshake** — The default handshake is Hardware. This describes the communication protocol between your printer or plotter and OS/2 Warp. Hardware means that communication is accomplished by electronic signaling, and requires a cable specified by the manufacturer. If communication is accomplished by XON/XOFF, select None.

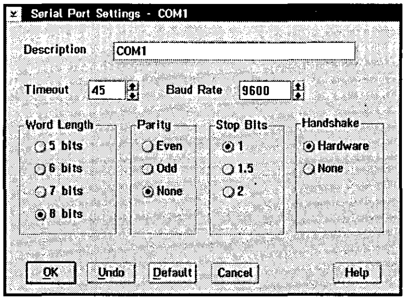
The pop-up menu for a Port Driver object also contains

- ✘ **Install** — Use this option to install another Port object.
- ✘ **Redirection** — Use this option to redirect jobs to another printer port. For example, if your Printer object currently has COM1 selected, you can display the pop-up menu for LPT1 and select Redirection. Print jobs sent to LPT1 will now be printed on COM1. See the section on Redirection, later in this chapter, for more information.

Share Access to a Port

Tip

The Share access option on the parallel port's Settings dialog allows DOS applications to share the same parallel port. If your DOS application tries to access the port during initialization, and the port is in use, an error message will be displayed. If you turn share access on, this error will not occur. If share access is on, and you see output from different applications being intermixed, you should turn share access off.



Serial Port Settings - COM1

Description: COM1

Timeout: 45

Baud Rate: 9600

Word Length: ○ 5 bits ○ 6 bits ○ 7 bits ● 8 bits

Parity: ○ Even ○ Odd ● None

Stop Bits: ● 1 ○ 1.5 ○ 2

Handshake: ● Hardware ○ None

Figure 14-16. Serial Port Settings Dialog

On the Output page, you can select Output to file if you want your print jobs to be sent to a file on the fixed disk, instead of to the printer. When a job is sent to a file, you will be prompted to enter the filename. This option can also be set using REXX and the OUTPUTTOFILE= setup string.

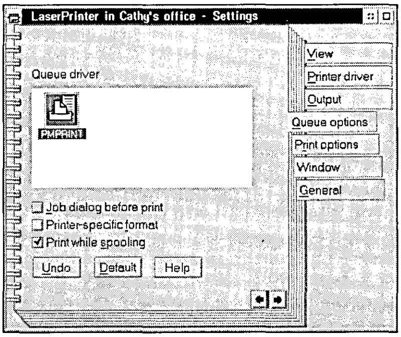
The spooler supports multiple printers simultaneously. A Printer object can be set up to use multiple printers at the same time (by selecting more than one port), in order to make the best use of available resources when printing jobs; this is called *printer pooling,* (see the section on Printer pooling, later in this chapter).

The spooler also supports *printer sharing,* meaning that multiple Printer objects can send jobs to the same physical device. For example, you can create one Printer object that prints urgent jobs, and another Printer object that prints large jobs, both directing the jobs to the same port (see the section on Printer sharing).

Queue options

The Queue Options page (see Figure 14-17) allows you to specify which queue driver this Printer object should use. PMPRINT is the default, but to install another driver, such as PMPLOT for plotter devices, select Install on the queue driver's pop-up menu.

The queue driver can also be set by using REXX and the QUEUEDRIVER= setup string. You can also choose the Job dialog before print option to display the job properties dialog before printing every print job; this option can also be set using REXX and the JOBDIALOGBEFOREPRINT= setup string. Choose Printer-specific format if you always want to print raw data; this option can also be set using REXX and the PRINTERSPECIFICFORMAT= setup string. Choose Print while spooling to start printing the job right away, even if the job hasn't finished spooling; this option can also be set using REXX and the PRINTWHILESPOOLING= setup string.


LaserPrinter in Cathy's office - Settings

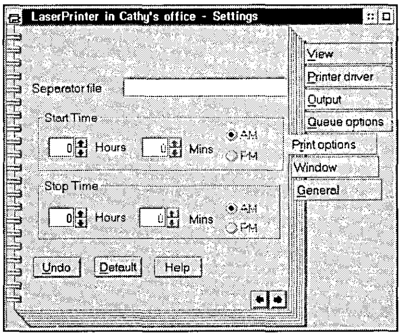
Queue driver: PMPRINT

□ Job dialog before print
□ Printer-specific format
☑ Print while spooling

Figure 14-17. Printer Object - Queue Options Page

Print options page

You can specify a separator file name on the Print options page (see Figure 14-18). The separator page will be printed before every print job. The separator file name can also be specified using REXX and the SEPARATOR= setup string. In the \OS2 directory, there are two sample separator files, PSCRIPT.SEP and SAMPLE.SEP.


LaserPrinter in Cathy's office - Settings

Separator file

Start Time: 0 Hours 0 Mins ● AM ○ PM

Stop Time: 0 Hours 0 Mins ● AM ○ PM

Figure 14-18. Printer Object - Print Options Page

Separator files can use the following codes:

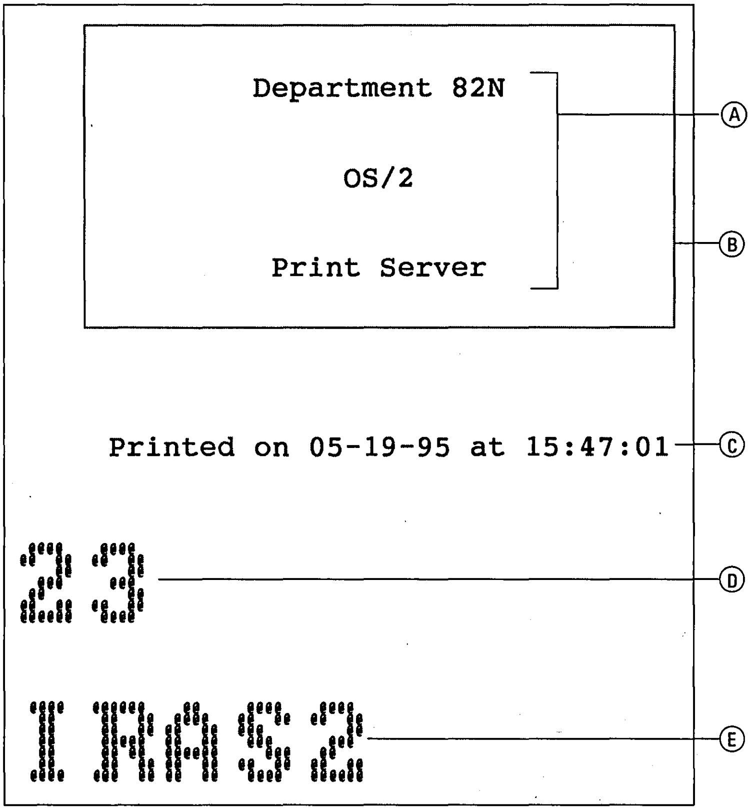
- @Ltext — Prints text
- @0 — Ends the line
- @D — Prints the job's creation date
- @T — Prints the job's creation time
- @N — Prints name of person or machine ID of job creator
- @J — Prints job name
- @I — Prints job ID
- @Q — Prints queue description and queue name
- @H*nn* — Sets printer-specific control sequence, where *nn* is a hexadecimal number
- @F*filename* — Prints the file. The default location for this file is in the \SPOOL directory
- @W*nn* — Sets page width, where *nn* is a number between 1 and 132
- @*n* — Skips *n* number of lines, where *n* is a number between 0 and 9
- @B — Creates block characters
- @S — Creates single-width block characters (use after @B)
- @M — Creates double-width block characters (use after @B)
- @U — Turns off block-character printing
- @E — Ejects a page. Starts a new page or ends the separator page

See Figure 14-19 for an example of separator page codes for an IBM 4019 Laser Printer. The separator page will print from the 200-sheet top tray, which contains colored paper. The normal output is printed from the 500-sheet bottom tray, which contains white paper. Also see Figure 14-20, which is an actual separator page generated by the code in Figure 14-19.

Figure 14-19. Separator Page Code

```
@
@H1B@L[K@H03@H00@H00@H31@H01
@H1B@L[F@H03@H00@H03@H01@H02
@H1B@L[@H40@H04@H00@H00@H00@H22@H02
@5@L             Department 82N@0
@1@L                  OS/2     @0
@1@L              Print Server @0
@H1B@L[j@H01@H00@H00@0
@H1B@L[Q@H06@H00@H02@H04@H00@HE0@H01@H8E
@H1B@L[i@H07@H00@H01@H00@H07@HA0@H00@H04@H01
@H1B@L[i@H07@H00@H01@H00@H00@H04@H03@HF6@H01
@H1B@L[Q@H06@H00@H00@H04@H07@H9C@H00@H00
@H1B@L[i@H07@H00@H01@H00@H00@H04@H03@HF6@H01
@H1B@L[Q@H06@H00@H03@H04@HF8@H64@H03@HF2
@H1B@L[i@H07@H00@H01@H00@H07@HA0@H00@H04@H01@0
@H1B@L[j@H01@H00@H03@0
@3@L     Printed on @D@L at @T@0
@H1B@L[@H40@H04@H00@H00@H00@H11@H01
@H1B@LE@H1B@LG@H1B@L0
@4@B@I@0
@4@N@0
@U
@H1B@LF@H1B@LH@H1B@LA@H0C@H1B@L2
@H1B@L[K@H04@H00@H00@H31@H01@E
```

Figure 14-20. Sample Separator Page



Department 82N

OS/2

Print Server

Printed on 05-19-95 at 15:47:01

23

IRAS2

The following describes what each line of separator page code accomplishes:

1 @ is the control character.

2 The printer is reset with the PPDS data stream.

The same separator page code is used for all modes, and the mode will be switched back to the correct mode in line 24. This allows us to have one set of separator page codes for PPDS mode, PCL mode, Plotter mode, and PostScript mode.

3 The second tray is selected.

Note

For this laser printer, tray renumbering is active, so tray 1 and tray 2 are switched. This means that the command to select tray 2 actually selects the top tray, which contains the colored paper.

4 Double height and double width characters are set.

5 - 7 The heading is printed (see A in Figure 14-20). The @5 at the beginning of line 5 spaces down 5 lines. The @0 at end of each line terminates the line on the printer (that is, it spaces to the next line). The @1 at the beginning of lines 6 and 7 spaces down 1 line before printing the text.

8 - 16 A box is drawn around the heading (see B in Figure 14-20).

17 The job's date and time are printed (see C in Figure 14-20).

18 Single height and single width characters are set.

19 Emphasized text, double strike and 1/8 inch line spacing are specified.

20 Block characters are created and the job ID is printed (see D in Figure 14-20).

21 The submitter's name is printed (see E in Figure 14-20).

22 Block characters are turned off.

23 Emphasized text and double strike are turned off. 1/6 inch line spacing is set.

24 The printer is reset. The separator page is printed and tray 1 is selected. The PPDS data stream is selected. The @E is required to terminate the separator page. Since we don't want to print a blank page, the Set Initial Conditions command (which resets the printer) specifies an extra byte of data, which absorbs the form feed command generated by the @E code.

Note

The same separator page code is used for all modes except for the character before the @E on the last line, which selects the data stream to be used for the print job. It is changed to:

- ✗ @H01 — Selects PPDS data stream
- ✗ @H02 — Selects PCL data stream
- ✗ @H03 — Selects Plotter data stream
- ✗ @H08 — Selects PostScript data stream

On the Print options page, you can also specify when this Printer object can print jobs. You can start and stop printing at certain times. If both the hours and minutes are set to 0, the Printer object will print at all times. If you specify a start time, you must also specify a stop time. The queue start time and stop time can be set using REXX and the QSTARTTIME= and QSTOPTIME= setup strings.

Selecting network printer options

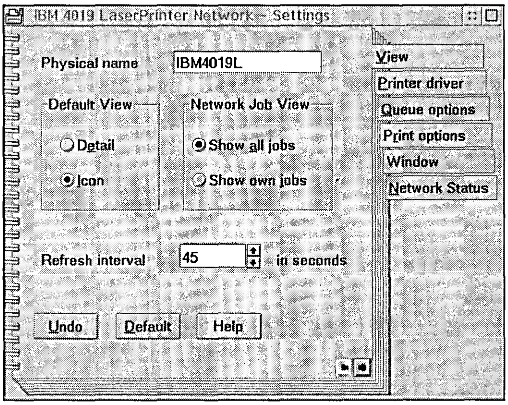
The Network Printer object Settings Notebook contains the same pages as the local Printer object, with the following exceptions.

View page

The View page (see Figure 14-21) allows you to change the Printer object's default open view to either Detail or Icon view. Changes on this page become effective when you either close the Settings Notebook or switch to another page. For network printers, the Network Job View and Refresh interval fields are displayed. Choose **Show all jobs** to see all jobs waiting to print on this network printer, or choose **Show own jobs** to see only the jobs that you have submitted to this network printer.

The Refresh interval shows how often this network printer's status is updated. By default, information is updated every 45 seconds. Selecting 0 means the network printer's job status will not be updated automatically; select Refresh on the pop-up menu to refresh at any time. Changing the refresh interval to more often than 45 seconds is not recommended, since other system tasks will slow down.

Figure 14-21. Network Printer Object View Page



IBM 4019 LaserPrinter Network - Settings

Physical name: IBM4019L

Default View: ○ Detail ● Icon

Network Job View: ● Show all jobs ○ Show own jobs

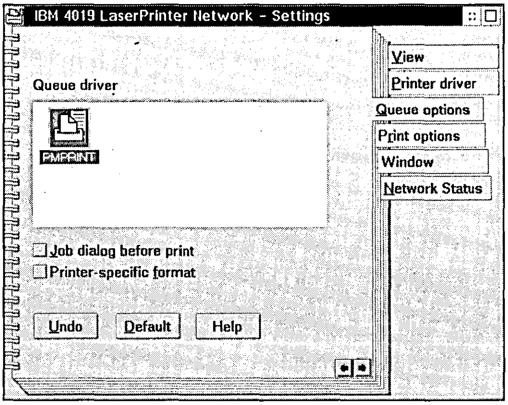
Refresh interval: 45 in seconds

Queue options page

The Queue options page (see Figure 14-22) allows you to specify which queue driver this Printer object should use. PMPRINT is the default, but to install another driver, such as PMPLOT for plotter devices, select Install on the queue driver's pop-up menu.

You can also choose the **Job dialog before print** option to display the job properties dialog before printing every print job. Choose **Printer-specific format** if you always want to print raw data. For local printers, choose **Print while spooling** to start printing the job right away, even if the job hasn't finished spooling. **Print while spooling** is not available for Network Printer objects.

Figure 14-22. Network Printer Object Queue Options Page



Queue driver: PMPRINT

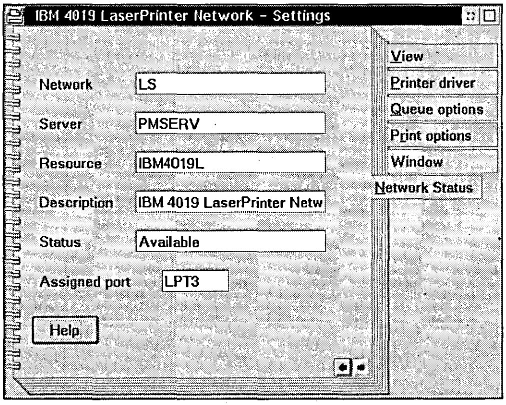
□ Job dialog before print

□ Printer-specific format

Network Status page

The Network Printer object adds the Network Status page (see Figure 14-23) to the Settings Notebook. This page displays the network printer's network, server, resource name, description, and status (available, not available, and so on). You can also see if a port has been assigned to the network printer.

Figure 14-23. Network Printer Object Network Status Page



Network: LS

Server: PMSERV

Resource: IBM4019L

Description: IBM 4019 LaserPrinter Netw

Status: Available

Assigned port: LPT3

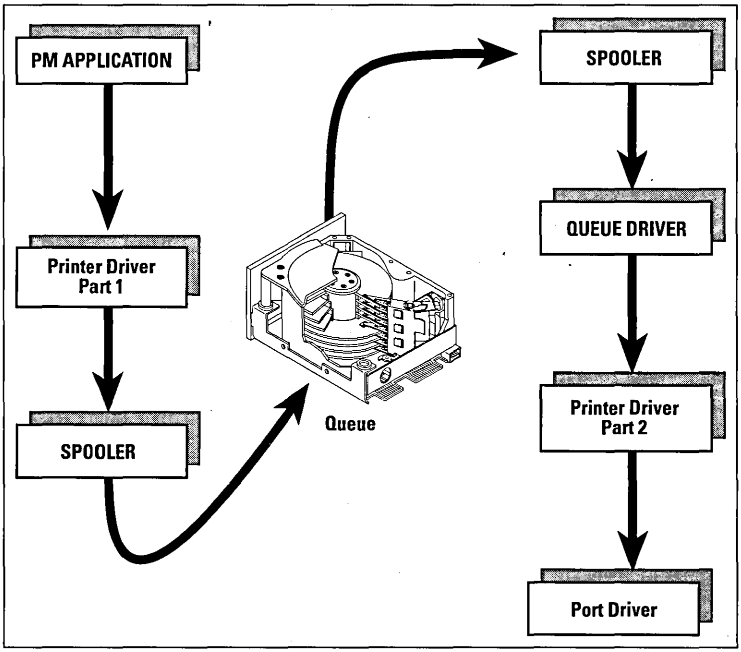
Advanced Topics

This section will describe the data flow in the printing subsystem, along with other advanced topics such as printer sharing, printer pooling, and redirection. The new support for printing using polling in OS/2 Warp will also be described.

Printer data flow

The flow of data from a Presentation Manager application to the serial or parallel printer is shown in Figure 14-24. The PM application's print data is sent through a first pass of the printer driver in order to produce a metafile in the spooler's queue. The spooler's queue processor (PMPRINT, for example) processes the jobs on the queue, making sure the job properties match the printer properties. (If the properties, such as forms, do not match, the job will be held in the queue with a "Forms mismatch" error until you change the properties to match.) The job will then go through a second pass of the printer driver to generate printer-specific data which will be printed to the port.

Figure 14-24. Printer Data Flow in the Printing Subsystem



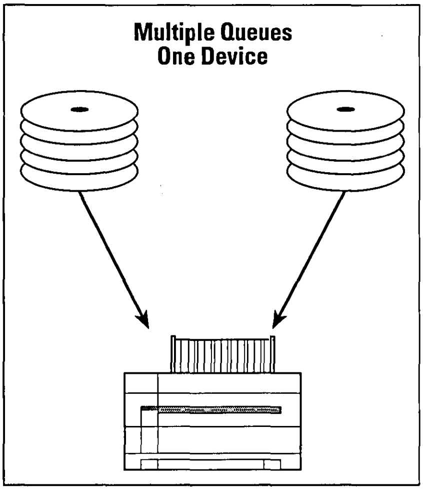
Printer sharing

Shhh! Tip

Printer sharing (see Figure 14-25) is the term used when multiple Printer objects are connected to a single printer. The advantage of printer sharing is that one Printer object can be set up with one set of printer properties, and another Printer object can be set up with a different set of printer properties.

For example, you could have one Printer object defined to print in landscape mode and another Printer object defined to print in portrait mode, both using the same LPT1 port and therefore the same printer. Printing a job to the first Printer object will print the job in landscape mode. Printing a job to the second Printer object will print the same job in portrait mode. This eliminates the necessity of changing the printer properties on one Printer object from landscape to portrait.

Figure 14-25.
Printer Sharing



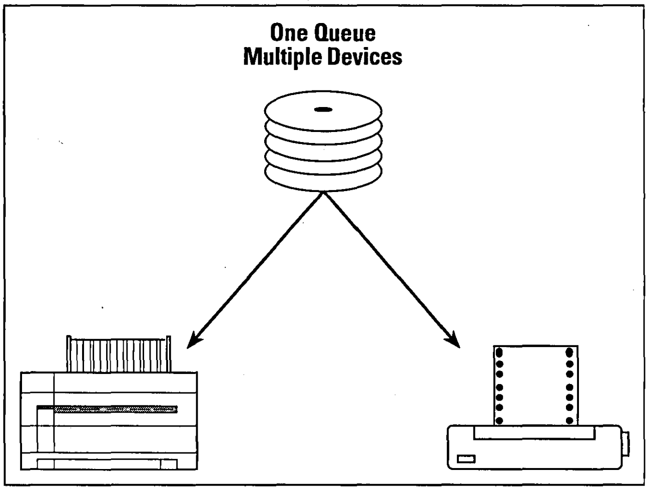
Printer pooling

Printer pooling (see Figure 14-26) is the term used when a Printer object has more than one port selected on the Output page of its Settings notebook. Print jobs sent to this Printer object can be printed to any of the selected ports for load balancing. The spooler will print the next job on the first available port, increasing the number of jobs that can print at any one time.

Shhh! Tip

This is a nice feature for a network administrator who may have multiple printers available.

Figure 14-26.
Printer Pooling



Note

When you pool printers, each printer must use the same printer driver

Redirection

When printing jobs from the command line, or from a non-Presentation Manager program, you may want to redirect your jobs from an LPT port to a COM port:

1. Select a COM port for the Printer object to use on the Output page of the Settings Notebook.
2. Close the Notebook to save the settings.
3. Open the Output page again, and display the pop-up menu for the LPT port. You will see the Redirection option.
4. Select the arrow to the right of Redirection to select the COM port that is associated to this Printer object.

Print jobs printed to the LPT port will now be redirected to the COM port.

Tip

Redirection can also be done from the command line by using the SPOOL command, SPOOL /D:LPT1 /O:COM1, where LPT1 is the input device and COM1 is the output device. Then, to print a job to the COM1 port, type **PRINT MY.DOC / D:LPT1**, and the print job will be redirected. SPOOL /Q will show you which ports are redirected.

New polling support

The following parallel protocols are available to transfer data:

- Character hardware interrupt driven
- Block hardware interrupt driven
- Polling

Note

Prior to OS/2 Warp, only the first two methods, character hardware interrupt driven and block hardware interrupt driven, were used for printing. Both of these methods use the parallel port's hardware-generated interrupt abilities. However, there are some printers which do not support interrupt sharing.

Uncensored

In OS/2 Warp, support was added for polling to fix this problem, and to also improve the performance of some printers. With the use of polling, the interrupt method for printing is no longer necessary. Polling makes another interrupt available for use (interrupts 5 and/or 7 are now available for other devices, such as sound cards) and allows cables that don't work with hardware interrupts to work correctly. The OS/2 Warp printer drivers are now polling enabled.

Uncensored

The PRINTXX.SYS drivers now have the option of using polling. Depending on the system and printer being used, performance may be dramatically improved. For 486 66MHz systems using fast laser printers, performance has been increased by a factor of five. For 486 66MHz systems, the top speed is approximately 70,000 characters per second. But with polling, since more instructions are executed for the transmission of characters, the processor loses time that could be used elsewhere. Because of this, OS/2 Warp gives system tasks run time when a polling write request is taking place by yielding when other threads of the same or higher priority are ready. In CONFIG.SYS, you can specify the /IRQ argument by using **BASEDEV=PRINTXX.SYS /IRQ**.

The following command-line arguments are used for PRINT01.SYS:

- **(None)** — The default for the PRINT01.SYS driver is to use polling (for LPT1-LPT3).
- **/IRQ** — The driver will print using the interrupt method. Printing (for LPT1-LPT3) will use either INT5, INT7 or timer interrupts.

The following command line arguments are used for PRINT02.SYS:

- **(None)** — The default for the PRINT02.SYS driver is to use polling unless the block hardware interrupt driven option is available for the device.
- **/IRQ** — The driver will print using the interrupt method. Printing (for LPT1-LPT3) will use INT7 and will use the block hardware interrupt driven option if it is available for the device.

If you want to force polling on an interrupt-driven port, you can edit CONFIG.SYS and replace BASEDEV=PRINT02.SYS with **BASEDEV=PRINT01.SYS.**

When using the polling parallel driver, you should always have the spooler enabled. If you notice that background jobs are running slowly, you can use the spooler to lower the print priority (see Chapter 7). If your printer is off-line, and you want to end the session that is printing, cancel the job and wait for the printer timeout. Then shut down your system. If you have problems using the polling driver, you can switch back to using interrupts by using the /IRQ option, or by installing OS/2 2.1 level drivers or below.

Summary

This chapter has described the OS/2 Warp printing subsystem, giving specific details on the Printer object. You've seen how to create a local or network Printer object and how to use it to display and manipulate jobs. You know the different ways to create a print job, and the difference between standard output and raw data. You've seen how to change the settings of a Printer object (including its printer driver, port, and queue driver). Advanced topics have also been discussed to give you a better idea of how the printing subsystem works, including printer sharing, printer pooling, redirection, and the new polling method of printing.

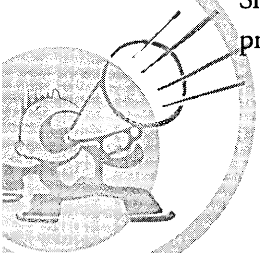
Cathy Bloomfield is a Senior Associate Programmer at IBM Boca Raton. She has been with the OS/2 Development Team since 1989, working on the Printer Manager, Spooler, Print Object, Network Object, and Workplace Shell.

Chapter 15

Command Line Interface

Secrets of the C:\ Prompt

This chapter covers many aspects of using OS/2 Warp from its command line. I will discuss what the commands are and how to use them. Because most commands operate on files and take arguments, I will also explore file access and command syntax. The information presented in this chapter is valuable outside the scope of the command line, because it will help you to work with various aspects of the Workplace Shell, such as creating program objects, which require an understanding of filenames, program names, and program parameters.



File access

In this section, I will explore the basic components of accessing files and programs on the various file systems that are available to you as a user of the OS/2 Warp operating system. While some of the material may seem like a review of more basic material, it is important that there be a good understanding of this foundation, which is built upon in later sections of this chapter.

Partitions and drives

Most preloaded computers ship with an unpartitioned hard drive, but you can choose to divide up your hard disk into as many as four partitions of equal or varying sizes.

There are two types of partitions:

- Primary partitions on the same hard disk share a drive letter. Only one primary partition on the hard disk can be seen by the operating system at one time. You might wish to do this if you are using the Boot Manager to switch between operating systems, and you do not want data that is visible under one operating system to be seen by the others for fear of having that data accidentally corrupted.
- Extended partitions can be further subdivided into logical drives. These drives are not separate physical disks, but you can assign a drive letter to them. Logical drives in an extended partition are seen by all operating systems that you boot from the Boot Manager. The potential problem with extended partitions is that if you ever add a new physical drive such as another hard disk or a CD-ROM drive, the drive letters associated with your logical drives will all shift up to the next letter.

File systems

OS/2 Warp ships with two file systems, the FAT and HPFS file systems. FAT stands for File Allocation Table, while HPFS stands for High Performance File System.

Path and filenames

All the programs and data that you store on your computer are described by an assigned filename. Table 15-1 describes the set of rules that govern what you can name your files.

Table 15-1. File Type Comparison

File Type	Rules Governing Filenames
FAT	The name may be up to eight characters long followed optionally by a period and up to a three-character extension. The following characters cannot be used as part of the filename: `\ / : * ? " . < > | , + = [ ] ;` Additionally, all characters that are below the number 1F as specified using hexadecimal notation are not valid filename characters.
HPFS	The name may be up to 254 characters long. The case you use when creating the filename is preserved, but file searches are not case sensitive. You may use as many periods as you would like in a filename. The file extension is those letters that occur after the last period in the name. The following characters cannot be used as part of the filename: `\ / : * ? " . < > |` Additionally, all characters that are below the number 1F as specified using hexadecimal notation are not valid filename characters.

The space character is a perfectly valid filename character for both file systems. Neither system allows you to use device names as filenames. The operating system will first examine the filename to see if it is a valid device name. If it is, it will use it without regard for any drive or path information that may be used with the filename. Given the following example, the file CONFIG.SYS is copied to the LPT1 device even though the user was attempting to create a file with the name LPT1.

```
COPY C:\CONFIG.SYS C:\LPT1.TXT
```

The following are some of the device names provided standard in OS/2 Warp:

KBD$
PRN
NUL
COM1
COM2
COM3
COM4
CLOCK$
LPT1
LPT2
LPT3
CON
SCREEN$
POINTER$
MOUSE$

Other device names may be provided by application developers; those additional device names would also be invalid as filenames.

You can create and use directories to store your files on your hard disk. Directory names follow all the same rules as filenames for the FAT and the HPFS file systems.

When you want to tell a program or a command where your file is, if it is not located in the current directory on the current drive, you have to include the path name as well. Each directory name in a path must be separated by a backslash (\ ). You may find that some programs and commands also accept the forward slash (/), but that is purely hit or miss.

OS/2 Warp recognizes two shortcuts for indicating the directory path. A single period (.) is taken to mean the current directory, and two periods (..) are taken to mean the parent directory. You should be able to use these shortcuts wherever a path can be specified. For example,

```
COPY . A:
```

will copy all files from the current directory to the A: drive. While:

```
COPY FILE.TXT ..
```

will copy the file FILE.TXT to the parent directory. There are two types of filenames: fully qualified filename and qualified filename. A fully qualified filename includes the complete designation of a file's name starting with its drive letter, listing all of its directory names, and ending with its filename. Following are two examples of fully qualified filenames:

```
C:\OS2\PMSHELL.EXE
C:\CONFIG.SYS
```

If you do not want to type out the fully qualified filename, you can use a relative path designation, sometimes referred to as a qualified filename, even if the file is not in the current directory. Following are a few examples of qualified filenames:

```
..\dir2\file.ext
```

The above example demonstrates a file that exists in a directory called dir2, which is in the parent directory of the current directory.

```
subdir\file.ext
```

This example demonstrates a file that exists in a directory called subdir, which is a subdirectory of the current directory.

```
a:file.ext
```

This example demonstrates a file that exists in the current directory of drive A:.

Because the FAT and HPFS file system support the concept of files and directories within directories, they are referred to as hierarchical file systems.

Extended attributes

In addition to the traditional information that is stored within a file, OS/2 Warp supports the concept of extended attributes. These are pieces of information that are stored separately from the file itself, although the operating system maintains the relationship between the file's name and its extended attributes. This allows application developers to keep bookkeeping information with the file so that if the file gets moved around by the user, the extended attributes get moved along with it. For example, the Workplace Shell uses the following extended attributes to store customizations made by the user to a file or directory:

.LONGNAME	The descriptive title of the file object
.ICON	The customized ICON that is to be used when displaying the icon for the file object
.TYPE	The list of types that are displayed on the file object's type page
.CLASSINFO	The place that most other settings information that describes the Workplace Shell object is kept.

Command symbols

In order for the command line to be of utility to the power user, a set of command line operators was provided so that you could do some amount of automation of your tasks without even having to write batch files. In this section, I will describe what these special characters are and how they can be put to good use.

Grouping symbols

At the OS/2 command line, you have the ability to execute more than one command on a single command line or on a single batch file statement. The following symbols help you do that.

&

The & character is used to unconditionally execute the commands that appear on either side of the & character.

Example: The following example attempts to copy both files to the A: drive without regard if one or the other succeeded or failed.

```
COPY chapt15.doc a:&COPY book.dot a:
```

&&

The double && characters are used to execute the second command or program only if the first one was successful. Success is determined by the command or program returning a zero return code back to the operating system.

Example: In the following example we can rewrite the previous example as follows:

```
COPY chapt15.doc a:&&COPY book.dot a:
```

This command will copy the file book.dot only if the file chapt16.doc was successfully copied.

||

The two vertical bars indicate that the command on the right will only be executed if the command on the left failed. Failure is the command or program returning a value other than zero back to the operating system.

Example: The following is an example of how to display a message if a command fails:

```
COPY chapt15.doc a:||echo The copy failed
```

()

You need to use parentheses when you want to group a set of commands into a single command or if you want to change the priority in which command symbols are interpreted.

For instance, if you want to execute multiple commands and have the combined output redirected to the same place, as if it was a single command, you can use the following command syntax:

```
(COPY chapt15.doc A:&DIR A:) > log.txt
```

Some commands such as the IF command take a single command as an argument. You can use parentheses to enter multiple commands:

```
IF "%1" = "YES" (COPY chapt15.doc A:&DIR A:)
```

Piping and redirection

You use piping and redirection to control the flow of input and output from commands and programs.

|

A single vertical bar is the piping operator. It indicates that the command on the left of it should have its output directed to the input of the program on the right of it. You can put several of these on a single command line, and it forms a "pipe" for the data.

Some examples of using the pipe operator are

```
TYPE config.sys | more
DIR /F | sort
TYPE config.sys | FIND /I "path" | SORT
```

>, >>, <, 2>, 2>&1

Redirection operators allow you to control the input and output of your command from or to an alternate source or target, which is usually a file but can be a device. Table 15-2 explains each of the redirection symbols.

Table 15-2. Redirection Operators

Symbol	Description
>	This allows the output of a command to be sent to either a file or a device, as the following examples demonstrate: DIR > file.out DIR > lpt1
2>	Same as > except that it redirects the output written to standard error instead of standard output.

continued

Table 15-2. Redirection Operators, continued

Symbol	Description
>>	This allows the output of a command to be appended to the end of an existing file. If the file does not exist, then it behaves just like the > operator. TYPE config.sys >> file.out
<	This allows the command to take its input from a file or a device. Most of the OS/2 commands described in this chapter will take their responses to their prompts from a redirected input file. In this way you can write batch files that automate the execution of these commands using response files. SORT < file.txt DEL * < Y.TXT CMD < XYZ.CMD
2>&1	This sends the output from standard error to the same location as the output from standard output. You can use it as follows: DIR A: > output.txt 2>&1 DIR A: 2>&1 I more

Taking characters literally

Now that I have introduced all these special characters' symbols, I have also introduced the problem of how you can use the character as the character and not as a special symbol. There are two ways of doing this on the command line.

^

The first way is by using the ^ symbol to indicate that the next character is to be taken literally. So if I want to use the & character in a filename or as part of a command, I could do the following:

```
MKDIR A^&B
```

This will create a directory called A&B. If I want to simply echo the & character out, I would use the following echo command:

```
ECHO ^&
```

To use the ^ character, just specify two of them, such as ^^.

""

I can also use quotes to surround the character or name that uses the special character. This is used most frequently for creating filenames with spaces in them. The following examples illustrate these points:

```
COPY XYZ.TXT "file with spaces"
MKDIR "A&B"
```

Executing commands silently

Using the @ symbol at the beginning of a command causes that command to be executed silently. Under normal circumstances, when a command is used in a command file, on an IF statement or with the FOR command, the command is echoed back out to the display. Using the @ symbol before the command prevents the command itself from being echoed onto the display. To see an example of this, try the following two commands to see the difference. The first example is without the @ symbol, and the second example uses the @ symbol:

```
FOR %I in ( one two three ) do echo %I.
FOR %I in ( one two three ) do @echo %I.
```

Environment variable substitution

You can use environment variable substitution at both the command line and inside of command files. To have the value of an environment variable expanded out, enclose the name of that environment variable in percent characters "%". The following examples show how I use this:

```
ECHO %temp%
SET PATH=C:\UTIL;%path%
```

The first example just displays the value of the environment variable "temp." The second one uses the value of the "path" environment variable to build a new one with a new path added at the beginning.

Commands

This section of the chapter is a reference for all the internal commands and a majority of the external commands.

Many of the commands support the use of wildcard characters. The two wildcard characters in OS/2 are * and ?. The question mark character is used to match a character in the same position as the question mark. The asterisk is used to match as many characters as can be matched. A single asterisk matches all filenames.

Note

Most commands take a /? parameter, which will cause the command to display a basic help message. Usually this is a one-line description of the command and a one-line description for each of its arguments and parameters.

Internal commands

Internal commands are those commands that are built into the protect mode command processor, namely CMD.EXE. You cannot find a program file that represents them. To execute an internal command from a Workplace Shell Program object, you need to specify it as a parameter to the CMD.EXE program. See the CMD command later in this chapter for more details.

All of the internal commands treat the following symbols as white space characters in addition to the space character unless the command indicates that it treats the symbol as significant:

- Comma “,”
- Equal sign “=”
- Semicolon “;”

So in the following example the two statements are identical:

```
COPY A B
COPY A,B
```

CALL

Syntax: `CALL command command-arguments`

Description: The CALL command is used to chain one batch file to another. This means that if you want to call another batch file from within a batch file and have the processing return back to the first batch file, you need to precede the name of the nested batch file with the call command. If you do not do this, the control will just be passed to the second batch file, and the statements following in the first batch file will not be processed.

Example: In the following example the statement nested.cmd results in what is called a *chained* batch file. This means that control is passed to nested.cmd and it does not return. This results in the statement echo line 3 never getting executed. Nor would any lines after it.

```
echo line 1
nested.cmd
echo line 3
```

Example: In the following example the problem is fixed by using the CALL command to indicate that control should be returned back to the original batch file after nested.cmd is done executing. This results in echo line 3 and all statements that might exist after it being executed.

Even though the CALL command is designed for working with batch files, it actually allows any command to be used following it.

```
echo line 1
CALL nested.cmd
echo line 3
```

Note

Do not confuse this CALL command with the REXX CALL command. When using the CALL command for nested batch file within a REXX command file, you have to enclose the CALL command in quotes, as in "CALL". In this way the REXX interpreter sees it as a command to be passed on instead of a command to be interpreted by REXX.

CD, CHDIR

Syntax:

```
CD X:
CHDIR X:
```

Description: In this form of the CD or CHDIR command, the current directory of the drive you specify is displayed. If you leave off the drive designator, then the current directory of the current drive is displayed.

Syntax:

```
CD path
CHDIR path
```

Description: In this form of the CD or CHDIR command, the current directory is changed to the directory specified in the path argument. If the path argument includes a drive specification, then the change only affects the drive specified; otherwise, if the path argument does not include a drive specification, then the change only effects the current drive.

So if the current prompt is [C:\] and you type **CD D:\DATA,** the current directory of the D: drive is changed to the \DATA directory. You will not see your prompt change because your CD command was not for your current drive. The current drive is the drive reflected in your command prompt. But if you next type **CD D:**, you will confirm that the D: directory was in fact changed to the \DATA directory. If you had done the same thing on your current drive, you would see your prompt change to [C:\DATA].

The CD command only takes one argument; however, any text you type after the first path argument is ignored and treated as a comment text.

CHCP

Syntax: `CHCP`

Description: Use this form of the CHCP command to see what code pages have been prepared for the system. A code page is a characters set that is used to define how characters should be displayed on the screen or when they are printed. They are specific to a given country and language. Let's work through the following example.

You installed OS/2 Warp on a machine you purchased here in the United States. The keyboard looks like the typical keyboard you would find on most machines in the United States. You also purchased your copy of OS/2 Warp in the United States. I'll assume that the information on the box is all written in English. Under this scenerio, your country code is 001 (U.S.A.) and your code pages are 437 (American English) and 850 (European English) with 437 being the default.

When you type what looks like to you as the letter A on your keyboard, all that does is send the number 65 to the computer and to OS/2 Warp. Notice that it does not send the letter A, just a number. Then OS/2 Warp routes the number 65 to its video subsystem to display it back to you. The video subsystem needs to know how to display the number 65 on the screen as a character. To do that, it looks up the number 65 in the current code page table, which is 437, for instructions on what symbol to draw. Under code page 437, it will find the instructions for drawing the A symbol. If you change the code page, the same number 65 that you input as the letter A may come out looking completely different, depending upon what the drawing instructions stored in the new code page are for that number.

Syntax: `CHCP nnn`

Description: Use this form of the CHCP command to change between the prepared code pages for the system. First use the CHCP with no arguments to determine what prepared code pages are available. Then you can use CHCP followed by the number of one of the prepared code pages to set the session code page to the other prepared code page.

CLS

Syntax: `CLS`

Description: Clears the screen. Even though this command takes no arguments, any text you type after the command name is ignored and treated as a comment text.

COPY

Syntax: `COPY [/A or/B]path or device [/A or/B][+][,,][path or device] [/A or/B][/V][/F]`

Description: The COPY command is one of the most complex of the internal OS/2 commands. It can have many forms, depending on the parameters used on the command line, thus making it several commands in one:

- /A — For all path and device arguments that directly precede this switch and for all device and path arguments that follow, copy in ASCII mode.

For a source path or device, ASCII mode means that the source will copy up to but not including any embedded CTRL-Z characters that might be contained in the file.

For a target path or device, ASCII mode means that the target file or device will have a final CTRL-Z added as the last character written to the file or device.

- ✗ **/B** — For all path and device arguments that directly precede this switch and for all device and path arguments that follow, copy in BINARY mode.

  For a source path or device, BINARY mode means that the source will copy all characters in the file or all characters coming in from a device. This can be troublesome on a device since the copy command will have no means of knowing when it should stop reading from a source device.

  For a target path or device, BINARY mode means that the target file or device will not have a final CTRL-Z added as the last character written to the file or device.

- ✗ **Path** — This can be any path specification. On a source path, it must include some path specification in addition to the drive letter, should you choose to specify a drive letter. On a target path, you can just specify a drive letter alone.

  Note

  Don't forget that you can use "." to designate the current directory and ".."to represent the parent directory of the current directory.

  If you choose to specify a filename portion of the path argument, you may also include wildcards in both the source and/or target path arguments.

  A target path argument is optional.

- ✗ **Device** — These are OS/2 device names, like LPT1 or COM1. The colon at the end of the device name is not required. It is actually better if you do not specify it.

  A target device argument is optional.

- ✗ **+ (plus)** — This can be used to separate source path or device names. It indicates that the multiple inputs from the combined sources should be combined at the target.

- **„ (double commas)** — When used in combination with a single "+" argument, this argument causes the date and time of the target file to be updated to the current date and time. If you do not specify a target, the date and time of the source file is updated to the current date and time.
- /V — This parameter verifies that the target file was written correctly by reading the target back and comparing it against the source.
- /F — This parameter will make the copy operation fail if one or more of the source files contain extended attributes and the target drive does not support extended attributes.

The best way to see various ways that the copy command can be used in combination with its arguments is to work through several examples:

Example: In the following example it is assumed that the current drive and directory are not on the D: drive.

```
COPY FILE1.TXT FILE2.TXT
COPY FILE1.TXT D:
COPY D:FILE1.TXT
```

In all three instances a single file is being copied to a target file. In the first form, it is FILE2.TXT in the current drive and directory. In the second form, it is copied to file FILE1.TXT in the current directory on the D: drive. In the third form, it is copied to the file FILE1.TXT in the current drive and directory. In all of these examples, the copy defaults to a binary copy at both the source and the target.

Example: The following example is very similar to the preceding example.

```
COPY *.CMD *.BAT
COPY . D:
COPY D:*.*
COPY FILE1.TXT D:*.*
```

The end result is that for every match of the wild card at the source, an individual file copy will take place to a different file at the target. It is as if you specified several individual COPY commands, where there is an individually named source and target file. This can be contrasted with an append mode copy, where several source files combine into a single target file. This form will also result in a binary mode copy. These examples show that if you do not indicate a filename portion of the argument for the source and target, then a filename of *.* is used.

Example: The following is an example of multiple source files being copied to a single target file.

```
COPY *.TXT BIGTXT.ALL
COPY FILE1.TXT + FILE2.TXT FILE3.TXT
```

In the first command, all files in the current drive and directory that match *.TXT are copied to the file BIGTXT.ALL. It will not matter if even only one file is matched, the fact that it is a wildcard source and a single target distinguishes it from COPY FILE1.TXT D:*.*. Multiple source specification matching copied to a single target file will result in ASCII mode copy at both the source and the target. This means that the data from the source is only copied up to but not including the first CTRL-Z character in the file. A CTRL-Z character is written to the target file even if the source file does not contain any CTRL-Z characters.

Copying files to a target device forces the default copy mode to be ASCII mode for the source files and BINARY mode for the target. Source files will be copied up to but not including the first CTRL-Z character of the file. A final CTRL-Z character will not be written out to the device, since it is in binary mode.

Example: The following is an example of a file being copied to a target device.

```
COPY FILE1.TXT LPT1
COPY *.TXT LPT1
COPY FILE1.TXT + FILE2.TXT LPT1
```

Sometimes this default is not desirable if you have a file that contains embedded control characters that are meaningful to your printer. This would be the case if you configured your OS/2 or Windows printer to print to a file. When you then went to copy this printer-specific file to the device for that printer, you would want the entire file sent to the printer. So use the /B option after the COPY command to force the COPY command into binary mode, as illustrated in the following example:

```
COPY /B FILE1.PS LPT1
```

There may be a time when you want to copy a file but you want to change the date and time stamp of the target file to be the current date and time. In that case, you can use the form of COPY command shown below:

```
COPY /B FILE1.TXT +,,
COPY FILE1.TXT +,,
COPY FILE1.TXT +,, FILE2.TXT
COPY D:*.TXT +,,
```

This form of copy is a combination of touch mode and combine mode copies. These forms of the COPY command default to being ASCII mode copy at the source and target. The COPY /B FILE1.TXT +,, forces it to be a BINARY touch, which means that only the file date and time are affected. Without the /B, the file contents would also be truncated after the first CTRL-Z character in the file.

A binary copy can be done via the Workplace Shell by using the **Copy...** menu choice on a data file object.

DATE

Syntax: `DATE`

Description: Entering the DATE command with no parameters causes the current date to be displayed. You are then prompted to enter a new date. If you choose to not change the date, simply press **Enter** on the keyboard.

Syntax: `DATE mm-dd-yy`

Description: This form of the DATE command allows you to change the current date without displaying it first. The date will simply be changed as specified unless the date you entered is not valid. You can use a dash, a period, or a slash to separate the date components. Additionally, in non-U.S. versions of OS/2 Warp, where a different date separator is used, the date separators appropriate for the given country may also be used. In other countries, the order of the numbers may be different, depending upon those countries' conventions.

DEL, ERASE

Syntax:

```
DEL path [/P][/N][/F]
```

```
ERASE path [/P][/N][/F]
```

Description: The DEL and the ERASE commands, which are synonyms for each other, are used to erase files from the file system.

Note

For the path argument, if you only specify a path name, then a filename of * is assumed. A filename of * causes all files in the directory to be deleted. You cannot just specify a drive letter. Remember that you can use a path specification of "." to mean the current directory.

Example: In the following example, the "." character is used to represent the current directory first on the current drive and then on drive D:. Since "." is shorthand for the current directory path and since if you specify a path, a filename of * gets added, it becomes a short hand way of specifying DEL *.

```
DEL .
DEL D:.
```

Following is a description of the various switches you can use with the DEL and ERASE commands:

- /P — Requests that you be prompted before each individual file is deleted. Unlike the DOS counterpart of this command, you have to press **Enter** before your response will be accepted.
- /N — Suppresses the "Are you sure?" message when you specify that you want all the files in a directory deleted. This is good for batch files where you know that you want to clean up all the files in a directory and you do not want to have to handle redirecting in a response to the "Are you sure?" prompt.
- /F — This permanently deletes the file such that it cannot be deleted with the UNDELETE command.

You can specify as many path arguments as you can enter at the command line. The DEL and ERASE commands will attempt to complete as many of them as it can. It will issue an error message for those that it could not process and then go on to the next path parameter.

DETACH

Syntax: `DETACH command command-arguments`

Description: Just like for the CALL command, you can precede any valid command and its arguments with the DETACH command to have that command run in the background with no associated window or keyboard. When you run a command in this manner, you need to be sure that any input that the program may require is redirected in from a file or other input source. Similarly, if you want to capture the output of the program, you will need to redirect the output to a file.

The commands that can be detached must be valid OS/2 mode internal or external commands. They cannot be DOS or Windows mode programs, nor can they be Presentation Manager programs.

DIR

Syntax: `DIR path [/W or /F][/P][/N][/A][/B][/O][/R][/S][/L]`

Description: The seemingly simple task of getting a directory listing of files and directories is a complex command filled with various options to help you use its output to perform various tasks.

You can specify any combinations of the three components of the path argument, made up of either the driver letter, a path name, a filename, or any proper combination of the three. Whenever a filename is not specified, a filename of * is assumed. If you want to actually get a directory of a path within that path's directory, add the * at the end. For example, typing

```
DIR C:\OS2\DLL
```

will give you a directory of all the files listed in the C:\OS2\DLL directory. But what if you were only looking to see the directory entry for the DLL directory within C:\OS2? In that case you would need to have typed

```
DIR C:\OS2\DLL*
```

A wealth of switches is available for the DIR command. Following is a description of these switches and how they can best be used.

- /W — Used to display the list of files, along with the directory header and footer information, in a wide format. The number of columns displayed is calculated by first finding the largest file in the directory and then using the number of characters in that filename to determine the number of columns to display.

- /F — This is used to display the list of files as a list of fully qualified drive, path, and filenames, with no directory header or footer information. This is most useful as input to another program that wants to act on that list of files. For example, the following might be used inside of a REXX command file:

  ```
  /**/
  'dir /f c:\os2 | rxqueue'
  do while queued() <> 0
    pull filename
    say filename
  end
  ```

- /P — Used to pause the output of the DIR command so that you have a chance to view a screenful of information before allowing the output to continue. The DIR command counts the number of screen lines currently available for the mode that you are in rather than just assuming a standard number like 25 lines.

- /N — Forces the directory output to be displayed in OS/2 format as opposed to the DOS-compatible format. This is the default for all cases except for local FAT partitions.

- /A — When used by itself, this causes all files to be shown even if their hidden or system attributes are turned on. You can use a combination of the following letters after the /A, with no intervening spaces, to only have files displayed that match the corresponding attributes:

  - H — Only shows files that are hidden.
  - -H — Only shows files that are not hidden.
  - S — Only shows files that have the system attribute enabled.
  - -S — Only shows files that do not have the system attribute enabled.

  - D — Only shows directories. This prevents normal files from being presented.
  - -D — Only shows files and not directories.
  - A — Only shows files that need to be backed up. In other words, they have not had their archive bit turned off.
  - -A — Only shows files that have been backed up and/or had their archived bit turned off will be displayed.
  - R — Only shows files that are marked as read only.
  - -R — Only shows files that are not marked as read only.

- ✘ /B — Similar to the /F command, it displays just the filename information, except without the drive letter and the path. Also, the directory header and footer information is suppressed.
- ✘ /O — Used to sort the directory entries in the output. When specified without any of the additional options, it first displays all the directory entries sorted by their filenames, and then displays all the files sorted by their filenames. You may also use any combination of the following after the letter O to modify the default sorting rules:
  - N — Sorts all entries, both directories and files interspersed, by the filename portion. For a directory entry, its filename is still the last component portion of its name.
  - -N — Same as N above, except that it is in reverse order.
  - E — Sorts all entries, both directories and files interspersed, by the extension portion of its filename. For a directory entry, its filename is still the last component portion of its name. For HPFS drives where the extension is not limited to three characters, the extension is still all characters after the last period in the filename.
  - -E — Same as E above, except in reverse order.

  - D — Sorts all entries in their date and time order, from oldest to newest.
  - -D — Same as D above, except in reverse order. I often use the DIR /O-D command to determine which files I have created that day or that are most recently changed.
  - S — Sorts the directory entries by the number of bytes of their file size, with the smallest ones being displayed first.
  - -S — Same as S above, except in reverse order.
  - G — Simply causes all directories in the order that they are found to be grouped before files.
  - -G — Simply causes all files in the order that they are found to be grouped before directories.

- ✘ /R — Causes the long filename extended attribute to be displayed with the directory entry. Amazingly, this parameter is only valid in DOS mode and not in OS/2 mode.
- ✘ /S — Causes all subdirectories to be searched, starting at the specified directory for the files specified. If you would like to search an entire disk for a filename, you need to either have the current directory be the root directory or to specify the root directory as part of the path argument.

  For example, DIR * /S causes all files on the current drive to be displayed, whereas DIR \FILE1.TXT would display entries for FILE1.TXT wherever it might exist on the drive. Unfortunately, you have to do this for every drive you want to search.
- ✘ /L — Forces the output to be displayed in lowercase letters.

You can specify as many path parameters as can fit on the command line. If consecutive path arguments resolve to the same drive and directory, then they are combined within the same directory header and footer information. Otherwise, a new directory header and footer information is generated each time a path argument resolves to a different drive or directory.

DPATH

Syntax: `DPATH`

Description: Displays the current value of the DPATH environment variable setting. You can also view the contents of all the environment variables by using the SET command.

Syntax: `DPATH ;`

Description: Sets the value of the DPATH environment variable to empty. This is exactly equivalent to SET DPATH=.

Syntax: `DPATH path1[;path2;..;pathn]`

Description: Sets the DPATH environment variable to the list of paths specified. Each path component needs to be separated by a semicolon.

The DPATH environment variable can be used by commands and other applications to find data files. It is not guaranteed that all programs use it, so you need to check the documentation for specific programs. But, for example, the TYPE command and input redirection will search the DPATH environment variable. To test this out, issue TYPE CONFIG.SYS from the command line while the prompt is not displaying the root directory of the boot drive.

ECHO

Syntax: `ECHO any text string`

Description: Use the ECHO command to display text. The ECHO command is used mostly in batch files, but its use is not limited to just batch files.

ENDLOCAL

Syntax: `ENDLOCAL`

Description: Use the ENDLOCAL command in combination with the SETLOCAL command to be able to save and restore a local environment that consists of

- ✗ Your current drive and directory.
- ✗ The value of your environment variables. For more information about environment variables, see the SET command.

EXIT

Syntax: `EXIT`

Description: The EXIT command ends the current command processor and returns control back to the program that started that command processor. If the command processor being ended was the first one in that session or window, then that session or window is ended or closed.

EXTPROC

Syntax: `EXTPROC command command-arguments`

Description: Use EXTPROC as the first line of a batch file to instruct the protect mode command processor that you want this batch file processed by the alternate command processor that you specified as the first argument after the EXTPROC command.

Tip

Check your alternate command processor's documentation to see if it supports the use of the EXTPROC command, because it has to know to ignore that line within the batch file. To just see the effect of the EXTPROC command, create the following TEST.CMD file:

```
extproc e test.cmd
line 1
line 2
line 3
```

Now run test.cmd from the command line, and you will find that your are editing the file test.cmd.

FOR

Syntax:

```
FOR %variable IN ( argument list ) DO command argument
FOR %%variable IN ( argument list ) DO command argument
```

Description: The first form of the FOR command is for use at the command line. The second form is for use within batch files.

The argument list can be any set of separated text. If any of the text contains either of the wildcard characters, * or ?, then it is treated as a filename, and the command following the DO is iterated for every file found that matches the text with the wildcard character.

The best way to explore the power of the FOR command is through some examples:

```
FOR %I in ( 1 2 3 4 5 ) DO command > run.%I
FOR %I in (%PATH%) DO if exist %I\epm.exe pause
FOR %I in ( *.CMD ) DO FOR %J in ( param1 param2 ) do %I %J
```

In the first example, the command following the DO keyword is executed five times, and each time it is executed, the output is saved in the files run.1, run.2, run.3, run.4, and finally run.5.

In the second example, all the semicolon-separated paths that make up the PATH environment variable encompass the argument list. The command after the DO keyword checks to see if the program epm.exe exists in the directory that is part of the PATH list. If the epm.exe program is found, then the pause command is executed. This is the FOR loop that I use to find where a program that I run from the command line is located.

In the third and final example, I am showing two things. The first is that the %I variable will be substituted for each file that matches *.CMD. The second thing is that you can nest FOR commands. The %J is iterated for each %I that is found. So if there are two files in the directory that match *.CMD, the first ONE.CMD and the other TWO.CMD, the following four commands will have been executed:

```
ONE.CMD param1
ONE.CMD param2
TWO.CMD param1
TWO.CMD param2
```

The variable can be inserted anywhere after the DO keyword. It can represent a full word by itself or be embedded as part of a word.

GOTO

Syntax: `GOTO label`

Description: Use the GOTO command, only within a batch file, to transfer control to a different part of the batch file. Let's look at the following example batch file:

```
command
if errorlevel 1 goto error
echo No error occurred
goto end
:error
echo An error occurred
:end
```

In the example given above, a sample command is executed. If an error is detected, control is transferred to a part of the batch file for error processing. Otherwise, the part of the batch file that handles the case of No error occurred, is executed. But it must then skip over the error-processing part of the batch file.

IF

Syntax: `IF [NOT] ERRORLEVEL number command arguments`

Description: Use this form to test if the last command executed returned an error or executed successfully. Check your program or command documentation to determine the possible return codes.

There is a trick to using the ERRORLEVEL test command correctly. The ERRORLEVEL test is really a test to see if the errorlevel returned by the last program or command is greater than or equal to the number you specified on the ERRORLEVEL command.

The following example shows the only proper way to check to see if an error occurred:

```
command
if errorlevel 1 echo an error was returned
```

You might think that you can use the following command:

```
command
if errorlevel 0 echo no error was returned
```

but this would not have the desired effect if in fact the command returned an error. This is because a successful return code is 0. An error return code is any other number. Therefore, since the error return code is greater than 0, the errorlevel test of greater than or equal to would be passed and the command would be executed. That is why you must always check for the highest number errorlevel first. If you want to see if any error occurred, check against errorlevel 1. You could check for success as follows:

```
command
if not errorlevel 1 echo no error occurred
```

Syntax: `IF [NOT] string1 == string2 command arguments`

Description: This form of the IF command is used to test if two strings are equal, but ignores the case of the string. This command may be used to check for arguments to a batch file or tests the value of an environment variable. Because the value of a batch file argument may be empty, it is best to safeguard string1 and string2 by enclosing them in matched quotes. Some reference books show following string1 and string2 with a period. But this method is not as safe as enclosing them in quotes. This is because the batch file parameter or the environment variable string may contain embedded blanks. The following two examples illustrate these points.

```
IF "%1" == "-q" SET quietmode=on
IF "%QUIETMODE%" == "on" echo your commands will be silent
```

If you precede the test with the NOT keyword, then the effect will be reversed.

Syntax: `IF [NOT] EXIST path command arguments`

Description: This form of the IF command is used to test to see if a file or directory exists. For example:

```
IF EXIST z:\mail\new.txt echo New mail exists
```

tests to see if the file z:\mail\new.txt exists, and if so, a new mail message is displayed.

When I install a new version of a program in a separate directory, I sometimes use the following command to compare the directories to see what files are no longer part of the program:

```
FOR %I in ( * ) do IF EXIST c:\new\%I del %I
```

KEYS

Syntax: `KEYS`

Description: This command displays whether the enhanced keyboard editing keys are enabled or disabled. By default the KEYS command is enabled by a SET KEYS=ON statement in CONFIG.SYS. If that statement is removed, then the true default of OFF is activated.

When enhanced keyboard editing is enabled, the following key definitions are enabled at the command line:

- ✗ ESC — Clears the current command line and returns the cursor to the begin ning of the command line.
- ✗ Home — Moves the cursor to the beginning of the command line.
- ✗ End — Moves the cursor to just after the last character on the command line.
- ✗ Ins — Toggles Insert mode. When Insert mode is on, characters can be inserted in the middle of the command line. When Insert mode is off, typing characters in the middle of the command line overwrites characters.
- ✗ Del — Deletes the character under the cursor and moves all characters that were to the right of the cursor one space to the left.
- ✗ Left Arrow — Moves the cursor left one character. It will wrap up to the next line if required.
- ✗ Right Arrow — Moves the cursor right one character. It will wrap down to the next line if required.
- ✗ Up Arrow — Cycles through the previously entered commands, newest to oldest
- ✗ Down Arrow — Cycles through the previously entered commands, oldest to newest.
- ✗ Ctrl-Left Arrow — Moves the cursor to the first character of the word. If the cursor is already on the first character of a word, it moves to the first character of the previous word.
- ✗ Ctrl-Right Arrow — Moves the cursor to the first character of the next word.
- ✗ Ctrl-End — Deletes all characters under and to the right of the cursor.
- ✗ Ctrl-Home — Deletes all characters from the beginning of the command line to just left of the cursor.
- ✗ Enter — Executes the current command and adds it to the command list if any characters were typed at the command line.

Syntax:

```
KEYS ON
KEYS OFF
```

Description: Sets enhanced keys editing on or off. Turning keys off does not destroy the list of commands in the keys queue. This way, if you turn keys back on, the list of previously entered commands are still intact.

Syntax: `KEYS LIST`

Description: This command displays the list of commands in the current keys queue.

MD, MKDIR

Syntax:

```
MD path
MKDIR path
```

Description: Use the MD or the MKDIR commands to create new directories. You can specify multiple path arguments to the command, and it will attempt to create as many of the specified directories as it can. You cannot make multiple directories with a single path argument. If you are attempting to make several directories, one nested within the other, you have to make them one at a time.

MOVE

Syntax: `MOVE source-path [target-path]`

Description: Moves the directory or the files that match the source-path argument to the directory specified by the target path. The target path must be on the same drive as the source-path, and you cannot specify a drive letter in the target path. Unlike other commands, the move command does not add a filename of * to the source path if only a path specification is entered. This is so that you can actually move a directory around the hierarchy. You cannot move a directory to one of its own subdirectories.

PATH

Syntax: `PATH`

Description: Displays the current value of the PATH environment variable setting. You can also view the contents of all the environment variables by using the SET command.

Syntax: `PATH ;`

Description: Sets the value of the PATH environment variable to empty. This is exactly equivalent to SET PATH=.

Syntax: `PATH path1[;path2;..;pathn]`

Description: Sets the PATH environment variable to the list of paths specified. Each path component needs to be separated by a semicolon.

The PATH environment variable is used by the command processor and other program launchers to look for the program name when only the filename is entered. Each directory entry specified in the PATH environment variable is searched for the program name specified at the command line. If no extension is specified, then each of the following extensions is searched for in the following order in each path entry before the next path entry is searched:

1. .COM
2. .EXE
3. .CMD
4. .BAT

If one program name exists with different extensions in the same directory and you want a specific one executed, just type the extension at the command line along with the reset of the command name.

Uncensored

It may be a little-known fact that a valid program may have any name. The command processor, CMD.EXE, requires that it have an extension, so if you want to call your program COMMAND.UTL, just type **COMMAND.UTL** at the command line and it will get executed.

PAUSE

Syntax: `PAUSE`

Description: Use the PAUSE command to suspend processing until the user presses a key. This is typically used from within a batch file to ensure that the user has read something that is important before the batch file continues. But its use is not restricted to a batch file and can be used in a compound command or with the FOR command to suspend processing until the user presses a key.

PROMPT

Syntax: `PROMPT`

Description: Resets the prompt to the default.

Syntax: `PROMPT text`

Description: Sets the prompt to the value specified by the text parameter. The text parameter can be any characters you like. You may also intersperse the special characters in Table 15-3, all of which include a dollar sign.

Table 15-3. Prompt Character Settings

Parameter	Description
$_	Causes the cursor to go to the beginning of the next line
$A	The & character
$B	The I character
$C	The (character
$D	The current date
$E	An escape character. This is useful when using an ANSI sequence as part of the prompt (see Table 15-4).
$F	The) character
$G	The > character

continued

Table 15-3. Prompt Character Settings, continued

Parameter	Description
$H	Causes the cursor to backspace
$I	Displays the help line at the top of the screen
$L	The < character
$N	The default drive
$P	The current directory of the current drive
$Q	The = character
$R	The return code of the last executed program or command. This number is the highest number that would pass the IF ERRORLEVEL number test.
$S	A space character
$T	The current time
$V	The version of the current operating system level

Table 15-4 lists the ANSI escape sequences that may be used as part of the prompt command. The characters given below immediately follow the $E sequence.

Table 15-4. ANSI Escape Sequences

Escape Sequence	Description
[line;colH	Set Cursor Position Sets the cursor position to the numbers specified by line and col. For example, $E[5;3H causes the cursor position to be set to the fifth line and the third column. An alternative short form is $E[H, which moves the cursor to the home position, which is at 0, 0.
[nA	Move Cursor Up Moves the cursor up the number of lines specified by n. If the cursor is already at the top of the display, then this sequence is ignored. For example, $E[4A moves the cursor up four lines.

Escape Sequence	Description
[nB	Move Cursor Down Moves the cursor down the number of lines specified by n. If the cursor is already at the bottom of the display, then this sequence is ignored. For example, $E[4B moves the cursor down four lines.
[nC	Move Cursor Forward Moves the cursor forward the number of columns specified by n. If the cursor is already at the rightmost column of the display, then this sequence is ignored. For example, $E[4C moves the cursor right four columns.
[nD	Move Cursor Backward Moves the cursor backward the number of columns specified by n. If the cursor is already at the leftmost column of the display, then this sequence is ignored. For example, $E[4D moves the cursor left four columns.
[s	Save Cursor Position Causes the current cursor position to be saved. Use Restore Cursor Position to restore it. For example, $E[s.
[u	Restore Cursor Position Restores the cursor position that was saved using Save Cursor Position. For example, $E[u.
[2J	Erase the Screen Erases the screen and moves the cursor to the home position. For example, $E[2J.
[K	Erase Line Erases all characters under and to the right of the cursor to the end of the line. For example, $E[K.
[n;...;nm	Set Graphics Mode Sets the graphics mode specified by one or more of the following values for n. These values do not change until you change them again using this command.

continued

Table 15-4. ANSI Escape Sequences, continued

Escape Sequence	Description			
	Foreground Colors:		Background Colors:	
	30	Black	40	Black
	31	Red	41	Red
	32	Green	42	Green
	33	Yellow	43	Yellow
	34	Blue	44	Blue
	35	Magenta	45	Magenta
	36	Cyan	46	Cyan
	37	White	47	White
	Text Attributes:			
	0	All attributes off		
	1	Bold on		
	4	Underscore on (valid for monochrome display adapters only)		
	5	Blink on		
	7	Reverse Video On		
	8	Concealed on		
	The following example sets all future text to be yellow on blue:			
	$E[33;44m			
[=nh	Set Video Mode			
	Sets the video mode to one of the following values:			
	0	40 × 25 monochrome (text)		
	1	40 × 25 color (text)		
	2	80 × 25 monochrome (text)		
	3	80 × 25 color (text)		
	4	320 × 200 4-color (graphics)		

Escape Sequence	Description
	5 320 × 200 monochrome (graphics) 6 640 × 200 monochrome (graphics) 7 Enables line wrapping 13 320 × 200 color (graphics) 14 640 × 200 color (16-color graphics) 15 640 × 350 monochrome (2-color graphics) 16 640 × 350 monochrome (16-color graphics) 17 640 × 480 monochrome (2-color graphics) 18 640 × 480 color (16-color graphics) 19 320 × 200 color (256-color graphics) For example, $E[=3h sets the video mode to 80 x 25 color text.
[=nl	Reset Video Mode This is exactly the same as Set Video mode except for 7, which is defined here to mean that line wrapping is disabled. Please note that the last character is a lowercase L.
[code;string;...p	Set Keyboard String Remaps the keyboard keys as identified by code to the string identified by string. For a list of codes, see Table 15-5. The codes themselves have an embedded semicolon. You must follow the last digit of the code with a semicolon before the string. Additionally, the KEYS command must be OFF before these keyboard remappings can work. The following example sets the F2 key to insert the string "Hello World" into the keyboard buffer: $E[0;60; "Hello World"p

Table 15-5 defines the list of remappable keyboard codes for use with the Set Keyboard String ANSI sequence above. A blank entry in the table means that key combination cannot be mapped.

Table 15-5. Remappable Keyboard Codes for Use with the Set Keyboard String ANSI Sequence

Key Code	Code	SHIFT Code	CTRL Code	ALT Code
F1	0;59	0;84	0;94	0;104
F2	0;60	0;85	0;95	0;105
F3	0;61	0;86	0;96	0;106
F4	0;62	0;87	0;97	0;107
F5	0;63	0;88	0;98	0;108
F6	0;64	0;89	0;99	0;109
F7	0;65	0;90	0;100	0;110
F8	0;66	0;91	0;101	0;111
F9	0;67	0;92	0;102	0;112
F10	0;68	0;93	0;103	0;113
F11	0;133	0;135	0;137	0;139
F12	0;134	0;136	0;138	0;140
Home (numeric)	0;71	55	0;119	
Up Arrow (numeric)	0;72	56	(0;141)	
Page Up (numeric)	0;73	57	0;132	
Left Arrow (numeric)	0;75	52	0;115	
Right Arrow (numeric)	0;77	54	0;116	
End (numeric)	0;79	49	0;117	
Down Arrow (numeric)	0;80	50	(0;145)	
Page Down (numeric)	0;81	51	0;118	
Ins (numeric)	0;82	48	(0;146)	

Key Code	Code	SHIFT Code	CTRL Code	ALT Code
Del (numeric)	0;83	46	(0;147)	
Home	(224;71)	(224;71)	(224;119)	(224;151)
Up Arrow	(224;72)	(224;72)	(224;141)	(224;152)
Page Up	(224;73)	(224;73)	(224;132)	(224;153)
Left Arrow	(224;75)	(224;75)	(224;115)	(224;155)
Right Arrow	(224;77)	(224;77)	(224;116)	(224;157)
End	(224;79)	(224;79)	(224;117)	(224;159)
Down Arrow	(224;80)	(224;80)	(224;145)	(224;154)
Page Down	(224;81)	(224;81)	(224;118)	(224;161)
Insert	(224;82)	(224;82)	(224;146)	(224;162)
Delete	(224;83)	(224;83)	(224;147)	(224;163)
Print Screen			0;114	
Pause/Break			0;0	
Backspace	8	8	127	(0)
Enter	13		10	(0,28)
Tab	9	0;15	(0;148)	(0;165)
Null	0;3			
A	97	65	1	0;30
B	98	66	2	0;48
C	99	67	3	0;46
D	100	68	4	0;32
E	101	69	5	0;18
F	102	70	6	0;33

continued

Table 15-5. Remappable Keyboard Codes for Use with the Set Keyboard String ANSI Sequence

Key Code	Code	SHIFT Code	CTRL Code	ALT Code
G	103	71	7	0;34
H	104	72	8	0;35
I	105	73	9	0;23
J	106	74	10	0;36
K	107	75	11	0;37
L	108	76	12	0;38
M	109	77	13	0;50
N	110	78	14	0;49
O	111	79	15	0;24
P	112	80	16	0;25
Q	113	81	17	0;16
R	114	82	18	0;19
S	115	83	19	0;31
T	116	84	20	0;20
U	117	85	21	0;22
V	118	86	22	0;47
W	119	87	23	0;17
X	120	88	24	0;45
Y	121	89	25	0;21
Z	122	90	26	0;44
1	49	33		0;120
2	50	64	0	0;121
3	51	35		0;122

Key Code	Code	SHIFT Code	CTRL Code	ALT Code
4	52	36		0;123
5	53	37		0;124
6	54	94	30	0;125
7	55	38		0;126
8	56	42		0;126
9	57	40		0;127
0	48	41		0;129
-	45	95	31	0;130
=	61	43		0;131
[	91	123	27	0;26
]	93	125	29	0;27
\	92	124	28	0;43
;	59	58		0;39
'	39	34		0;40
,	44	60		0;51
.	46	62		0;52
/	47	63		0;53
'	96	126		(0;41)
Enter (keypad)	13		10	(0;166)
/ (keypad)	47	47	(0;142)	(0;74)
* (keypad)	42	(0;144)	(0;78)	
- (keypad)	45	45	(0;149)	(0;164)
+ (keypad)	43	43	(0;150)	(0;55)
5 (keypad)	(0;76)	53	(0;143)	

RD, RMDIR

Syntax:

```
RD path
RMDIR path
```

Description: Use the RD or the RMDIR command to remove a directory. In order for the directory to be able to be removed, both of the following must be true:

1. No files or other directories must exist under that directory.
2. It must not be the current directory in any process.

You can specify as many path arguments as will fit. It will attempt to remove as many of the specified directories as it can. If an error occurs during the processing of any of the path arguments, an error is displayed and the next argument is attempted.

REM

Syntax: `REM comment`

Description: The REM command is used primarily to comment batch files. It comments out the rest of the line. No interpretation is done on any of the characters of the comment, so you can use command symbols in the comment.

REN, RENAME

Syntax:

```
REN source-path target-file
RENAME source-path target-file
```

Description: Renames the directory or files specified by the source-path parameter to the name specified by the target-file parameter. Since the file or directory cannot be moved as part of the rename operation, it is invalid to specify a drive letter or path information on the target. If the source-path argument is the name of a directory, then the directory is renamed. It does not assume a source file of * as many of the other commands do.

You can use global characters in both the source and the target. For example:

```
REN D:\TEXT\*.TXT *.DOC
```

will rename all the files that match the search filter *.TXT and rename them with the extension *.DOC.

SET

Syntax: `SET`

Description: This version of the SET command causes all variables defined in the environment to be displayed.

Syntax: `SET X=Y`

Description: This causes the SET command to set the variable as indicated by X to be set to the value as indicated by Y. X and Y are arbitrary text strings. Check your program and command documentation for the list of environment variables it supports. You can also use environment strings as variables inside a batch file.

Tip

If you are extending a value to a certain environment variable, it is often useful to use environment variable substitution to add to the beginning or end of the environment variable that you are updating. For example, `SET PATH=C:\NEWUTILS;%PATH%` adds C:\NEWUTILS to the head of the PATH environment variable without forcing you to retype the whole thing yourself.

Syntax: `SET X=`

Description: This causes the environment variable as indicated by the letter X to be deleted from the environment. For example, `SET TEMP=` will remove the TEMP environment variable.

SETLOCAL

Syntax: `SETLOCAL`

Description: Used within batch files to preserve a copy of the local environment and the current drive and directory. This is useful when you want the user to be able to run your command file and have the user experience no change to his/her environment when your command file has finished executing. Use this command in conjunction with the ENDLOCAL command.

Example: The following example reflects a command file that acts as a front end to an editor. It points the PATH and TEMP environment variables to the editor directory and then starts the editor with the parameter passed into the batch file. By surrounding the commands with a SETLOCAL/ENDLOCAL pair, we have preserved the user's original values for PATH and TEMP without having to query and save them individually.

```
SETLOCAL
SET PATH=D:\EDITOR
SET TEMP=D:\EDITOR
EDITOR %1
ENDLOCAL
```

SHIFT

Syntax: `SHIFT`

Description: This command is used when there have been more then 10 replaceable parameters sent to a batch file. After querying the values of the first 10, you use the SHIFT command to gain access to the next parameter. The following example shows a command file that uses the SHIFT command to loop through any and all parameters passed to it.

Example: In the following example, every time you use the SHIFT command, the value in %1 is thrown away and replaced with what was in %2. Similarly, %2 is replaced with what was in %3, and so on down the line.

```
:loop
if "%1" == "" goto done
echo %1
shift
goto loop
:done
```

START

Syntax: `START ["program title"][/K or /C or /N][/F or /B][/PGM][/FS or /WIN or /PM or /DOS][/MAX or /MIN][/I] command arguments`

Description: The START command is used to start a program in another session. The default behavior of the START command with no arguments or parameters is to start a new OS/2 Windowed Command Prompt session. If you specify a command name and optionally its arguments, the START command will attempt to start that program in the session type that is most appropriate for the program. For OS/2 programs (non-PM) and DOS programs, it will execute the program by starting a new session with a command processor and passing that program as a parameter to the command processor. Following is a list of the parameters and switches that are available for this command:

- ✘ **"program title"** — If specified, this parameter must be enclosed in quotes. It is used to name the new session as its name should appear in the window list.

- ✘ /K — For OS/2 non-PM and DOS programs, this is used to indicate that the COMMAND parameter should be passed to the command processor by specifying the /K switch to the command processor. This keeps the command processor from exiting after the command has been executed. This is the default.
- ✘ /C — For OS/2 non-PM and DOS programs, this is used to indicate that the COMMAND parameter should be passed to the command processor by specifying the /C switch to the command processor. This causes the command processor to exit after the command has been executed. This is useful for executing commands and then having the session terminate when the command is finished.
- ✘ /N — Runs the command argument directly in the new session without a command processor in front of it. This is the default for PM programs.
- ✘ /F — Runs the new program in the foreground. Specifying any of the session switches (for example, /FS, /WIN, /PM, or /DOS) causes this switch to be made the default. Many PM applications come to the foreground anyway, even if this parameter has not been specified. To be sure, they are ill-mannered and ill-behaved applications.
- ✘ /B — Runs the new program in the background. This is the default except when one of the session switches has been specified.
- ✘ /PGM — Indicates that the next parameter that does not begin with a switch character is to be interpreted as the command argument. This is useful when you want to enclose the command in quotes but you do not wish to specify a program title. Without this switch, the first quoted string seen is always interpreted as the program title.
- ✘ /FS — Indicates that the program should be run in a full-screen session. This is not valid for PM applications.
- ✘ /WIN — Indicates that the program should be run in a window. This is not valid for PM applications. To indicate that you want a WIN-OS/2 Window, use this switch.

- ✗ /PM — Indicates that the program should be run in the PM session. You should never have to specify this switch, because the START command is normally able to detect that the application is a PM application.
- ✗ /DOS — Indicates that the program is a DOS mode application.
- ✗ /MAX — When the application is to be run in a window or if the application is a PM application, this indicates that the application is requested to be maximized. The PM application may choose to not honor this request.
- ✗ /MIN — When the application is to be run in a window or if the application is a PM application, this indicates that the application is requested to be minimized. The PM application may choose to not honor this request.
- ✗ /I — Indicates that the new session should inherit its environment from the Shell process instead of the current process. The environment includes all environment variables, current drive and directory, and any open file handles.

The following are some examples of the START command:

```
START
```

Starts a new OS/2 Command Prompt window.

```
START "As easy as 1,2,3" abc
```

Starts the program abc in the background and gives the session a fancy title.

```
START /WIN winfile
```

Starts the program winfile.exe in a seamless session on the PM desktop.

```
START /FS fdisk
```

Starts the fdisk program in a full-screen session.

```
START /PGM "DIR > OUT"
```

Starts the DIR command and redirects its output to the file OUT in another session. Since the command contains embedded command symbols, it needed to be enclosed in quotes. Since it was enclosed in quotes and I did not wish to specify a program title, I needed to use the /PGM switch to indicate that it was a command and not the program title.

TIME

Syntax: `TIME`

Description: Entering the TIME command with no parameters causes the current time to be displayed. You are then prompted to enter a new time. If you choose to not change the time, simply press **Enter** on the keyboard.

Syntax: `TIME hh:mm:ss`

Description: This form of the TIME command allows you to change the current time without first displaying the current time. The time will simply be changed to the time you specify unless the time you entered is not valid. The separator character used can be a colon or the valid time separator for the country.

TYPE

Syntax: `TYPE path`

Description: The TYPE command is used to display the contents of the files specified by the path parameter. You must specify a filename portion, if it does not default to *. You can specify global characters and multiple file specifications on the TYPE command. All files found will be typed. In the case of multiple file specifications and/or the use of global characters, the name of the file is displayed before the contents.

In order to help you control the output, the filename is written to handle 2 (standard error) while the file content is written to handle 1 (standard output). You can choose to redirect them to the same place or to different places, based on your intended result.

VER

Syntax: `VER`

Description: The VER command displays the current level of the operating system.

VERIFY

Syntax: `VERIFY`

Description: Displays the current value of the VERIFY setting.

Syntax: `VERIFY ON or OFF`

Description: VERIFY ON causes the system to verify that data is written to a disk correctly. It is slower than the default, which is OFF, because it reads back the data after it is written and compares it to the source.

VOL

Syntax: `VOL drive:`

Description: Displays the volume label and serial number for the drive specified. If no drive is specified, it displays it for the default drive. You can specify multiple drive letters as arguments.

External commands

External commands are program files with an extension of either .COM, .EXE, .BAT, or .CMD. See the PATH command for information about how external commands are searched for.

ANSI

Syntax: `ANSI`

Description: Displays if ANSI support is enabled for the current session. By default, ANSI support is enabled in OS/2 mode sessions.

Syntax: `ANSI ON or OFF`

Description: Used to enable (turn on) or disable (turn off) ANSI support. To use ANSI keyboard substitution, you must ensure that KEYS is set to OFF.

ATTRIB

Syntax: `ATTRIB [+R or -R][+S or -S][+H or -H][+A or -A] path [/S]`

Description: The ATTRIB command is used to change attributes on the specified file or directory as specified by the path argument. You may specify wildcard characters for the filename, in which case only those files that match the pattern will be affected. The following switches can be used with the ATTRIB command.

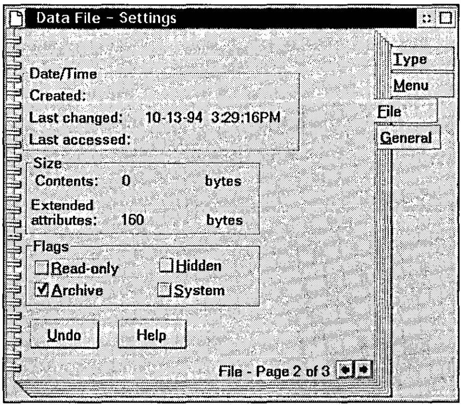
- ✗ **+R** — Enables the Read only attribute of the file.
- ✗ **-R** — Disables the Read only attribute of the file. The Read only attribute on a file is generally used to prevent a file from being accidentally deleted.

- +S — Enables the System attribute of the file.
- -S — Disables the System attribute of the file. The System attribute has no well-defined meaning. Its practical effect is to render the file hidden.
- +H — Enables the Hidden attribute of the file.
- -H — Disables the Hidden attribute of the file. The Hidden attribute on a file is generally used to prevent a file from being seen when the user goes to inspect the contents of a directory.
- +A — Enables the Archive attribute of the file.
- -A — Disables the Archive attribute of the file. The Archive attribute on a file is generally used to indicate that the file needs to be backed up. The file system automatically turns this attribute on whenever a file is created or changed.
- /S — Indicates that all subdirectories below the subdirectory as specified or implied in the path argument are to be searched for files matching the files specified in the part argument.

If none of +R, -R, +S, -S, +H, -H, +A, or -A is specified, then the ATTRIB command displays all files that match the path argument and shows the values of each of the attributes that are set. With its support of the /S switch, you can also use the ATTRIB command to look for files on a drive. For example, ATTRIB D:\BOOK.DOT /S searches for the file BOOK.DOT wherever it may exist on the D: drive.

To do the same function through the Workplace Shell, use File - Page 2 of 3 in the Settings Notebook of the data file object whose attributes you wish to manipulate, as illustrated in Figure 15-1.

Figure 15-1. Setting File Attributes Through the Settings Notebook



Data File - Settings

Date/Time
Created:
Last changed: 10-13-94 3:29:16PM
Last accessed:

Size
Contents: 0 bytes
Extended attributes: 160 bytes

Flags
☐ Read-only ☐ Hidden
☑ Archive ☐ System

File - Page 2 of 3

BACKUP

Syntax: `BACKUP source-path target-drive [/L:filename][/D:mm-dd-yy][/T:hh:mm:ss][/M][/A][/F:size][/S]`

Description: The BACKUP command is used to back up files. By most standards, the BACKUP facilities provided by the BACKUP command are rudimentary.

Developer's Desktop

I use the command mostly to transport a bunch of files between machines when I do not have access to an archiving tool such as PKZIP.

Shhh! Tip

I strongly recommend that you use a commercial backup program to protect your data, but something is better than nothing, so you can at least use this command to protect important data files and programs.

The target of a BACKUP command must be a drive letter. You cannot specify a path or filenames in the target. All files in the root of the target drive will be erased by the BACKUP command if the target drive is a diskette. If the target drive is not a diskette, then a backup subdirectory is created in the root directory. If the backup command cannot fit all the files onto a single diskette, it will prompt you to insert a new one.

The source-path argument of the BACKUP command can be any valid combination of a drive letter, path specification, and/or file specification. If the source-path does not include a file specification, then a file specification of * is assumed.

The following switches may be used with the BACKUP command. Common examples of the command follow this list.

- **/L:filename** — /L indicates that you want a log file created in the root directory of the source drive. The default name for the log file is BACKUP.LOG. If you wish to give the log file a different name, then you can augment that /L switch with a colon followed by the filename you wish to use.
- **/D:mm-dd-yy** — The /D switch is used to indicate that only files that were last modified on or after the date specified should be backed up.
- **/T:hh:mm:ss** — The /T switch is used to indicate that only files that were last modified on or after the time specified should be backed up.

Note

There is no way to combine the two switches to indicate, for example, that all files created after 09:00:00 on 7-4-96 should be backed up. The problem is that all files created between midnight and 9 a.m. on July 4, 1996, will be backed up because they meet the date criteria.

- /M — The /M switch is used to indicate that only those files whose archive attribute is turned on should be backed up. Another way of thinking about this is that only those files that have been created or changed will be backed up.
- /A — The /A switch is used to add this backup session to an already existing backup session and to therefore create one new, and larger, backup session that can then be restored using a single restore command.
- /F:size — The /F switch is used to instruct the backup command to format an unformatted diskette. This switch does not affect hard disks. You can modify the /F switch by specifying a colon followed by the diskette size. For example, /F:360, /F:720, /F:1200, F:1440, or /F:2880 are common parameters.
- /S — The /S switch is used to indicate that all subdirectories below the specified or implied source subdirectory should be included in the backup search.

Following are some examples of backup commands:

```
BACKUP D:\* A: /S
```

does a complete backup of the D: drive to diskettes in the A: drive.

```
BACKUP D:\WINWORD A: /M /S /A
```

backs up all files that have changed in the D:\WINWORD directory and adds them to the last backup.

```
BACKUP D: A: /F:1440
```

backs up all files in the current directory of the D: drive to diskettes in the A: drive. Additionally, all unformatted diskettes will be formatted for 1.44MB.

BLDLEVEL

Syntax: `BLDLEVEL executable-file`

Description: The BLDLEVEL command lets you display the internal revision level of executable files and dynamic link libraries. This can be useful when talking with an OS/2 support representative to ensure that you are at the latest service level of the operating system.

BOOT

Syntax: `BOOT /DOS`

Description: When you are in OS/2, BOOT /DOS will boot you to the DOS operating system, provided that you have it installed.

Syntax: `BOOT /OS2`

Description: When you are in DOS, BOOT /OS2 will boot you to the OS/2 operating system. Following a BOOT /OS2, CHKDSK will automatically be run when OS/2 is first being started. This is because if you deleted files while you were in DOS, their corresponding extended attribute information in the directory entries for those files is no longer valid. Those "dangling" extended attributes need to be removed from the database of extended attributes.

CACHE

Syntax: `CACHE`

Description: Used to display information about the current state of the High Performance File System (HPFS) cache buffer. The cache buffer that is controlled by the CACHE command is the area of memory dedicated to HPFS to store file system information before it is written out to disk. By writing information first to the cache, system performance is enhanced. The information in the cache is written to disk when

1. The system is less busy.
2. The cache becomes full.
3. The operating system is shutdown via the Desktop Shutdown utility.

Syntax: `CACHE [/LAZY:state] [/MAXAGE:time] [/DISKIDLE:time] [/ BUFFERIDLE:time]`

Description: The following switches may be used to modify the effect of the HPFS cache from the command line:

- ✗ /LAZY:ON — This is the default. This indicates that all writes to HPFS should go through the cache. This increases system performance at the expense of safety. Should the system crash before the cache is written out to disk, then you may suffer anything from a few lost files to a corrupted system.

- **/LAZY:OFF** — If you wish to ensure safety at the expense of system performance, then specify the following command in your STARTUP.CMD file:

  ```
  C:\OS2\CACHE /LAZY:OFF
  ```

Use the following switches to modify the cache statement that appears in CONFIG.SYS:

- **/MAXAGE:time** — This switch is used to specify the maximum age of the data in the cache before the data must be written out to disk. You specify the time option as a decimal number that represents milliseconds. A millisecond is $^1/_{1000}$ second, so 1,000 milliseconds is 1 second.

  The default value is 5,000 milliseconds, which is 5 seconds.

- **/DISKIDLE:time** — This switch, also a numeric value representing milliseconds, is the number of milliseconds that the disk must have been idle before it can accept data from the cache. This number must be greater than /BUFFERIDLE.

  The default value for this parameter is 1,000 milliseconds, or 1 second.

- **/BUFFERIDLE:time** — This switch indicates the amount of time that the cache must be idle before it starts writing its data out to the disk. The default value is 500 milliseconds or $^1/_2$ second.

CHKDSK

Syntax: `CHKDSK path [/F][/V] [/C][/F:n]`

Description: The CHKDSK is normally used to check for errors on an entire disk. It may also be used to check for errors in an individual file or directory. When used with an individual file or directory, it also informs you if that file or directory is contained in adjacent storage units on the disk.

To check the state of a complete drive, do not specify a path or drive argument. Add a file or directory argument to check the state of an individual file or directory. The file argument may contain wildcard characters. In all cases, the CHKDSK command displays general information about the disk as a whole.

The following switches may be used with the CHKDSK command:

- ✘ /F — Tells CHKDSK to fix any errors that it may encounter.

  If the error is that two or more files are cross-linked, then you must solve the problem by copying each of the files that are cross-linked to a file with a different name. Then after all of the files have been copied, delete the original files that CHKDSK reported as cross-linked. Then rename the copies back to the original names. Some of the files may have lost data.

  In order for the /F parameter to be used, there must not be any open files on the drive to be checked. This rules out the drive that OS/2 boots from. To run CHKDSK on that drive, you need to do the following:

  1. Open the Create Utility Diskettes object in the System Setup folder.
  2. Boot the system, using diskette 1 of the utility diskette.
  3. Run CHKDSK off of the utility diskettes.

- ✘ /C — Specifies that files will only be recovered if the system was started in an inconsistent state, which happens when the computer is turned off before all files are closed. This happens if you just turn the power off.

  This switch is unique to HPFS.

- ✘ /F:n — For an HPFS drive, it specifies the fix level where:

  /F:0 Causes the file system to be examined without any fixes being performed.

  /F:1 Causes the file system to resolve inconsistent file system structures.

  /F:2 Same as /F:1, but searches disk space that is marked as used but is not pointed to by any file system directories. If anything that resembles a file or directory is detected, then it is recovered. This is the default if just /F is specified.

  /F:3 Same as /F:2, but the entire disk partition is scanned.

CLIPOS2

Syntax: `CLIPOS2`

Description: Starts the OS/2 PM clipboard viewer application. This is equivalent to opening the Clipboard Viewer object in the Productivity folder, as shown in Figure 15-2.

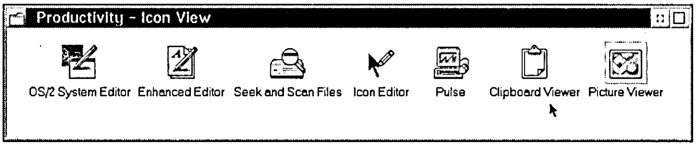


Figure 15-2. Clipboard View in Productivity Folder

CMD

Syntax: `CMD [comspec-path][/Q][/K or /C "string"]`

Description: CMD is the program name for the OS/2 command processor. This is the program that provides you with a command line. You enter commands, and CMD.EXE interprets the commands and any parameters and then executes them.

If CMD does all this for you, you may be wondering why you need to know how to start it and pass it commands. The reason is because you implicitly start CMD every time you use one of the OS/2 Command Prompt objects in the Command Prompts folder or when you use the START command. So by using the following syntax to specify parameters to CMD, you can modify or create new Command Prompt objects with a parameters field that has a syntax as described here.

- **comspec-path** — This parameter, if specified, is a drive and directory specification to a directory where a copy of CMD.EXE exists. The COMSPEC environment variable is then set to point to that directory. The value of the COMSPEC environment variable is used internally by CMD.EXE whenever it needs to start a new version of itself. It does this for the START and DETACH commands.

  This is why if you take the % character out of the Command Prompt objects, you get an error message about the COMSPEC variable not being set properly. That is because the argument being passed is being interpreted as this parameter.

- /Q — This is a quiet switch. It is used when you are redirecting input to CMD.EXE. When you specify the /Q switch, the commands that are being executed are not displayed.

- /K "**string**" — The /K switch instructs CMD.EXE to execute the command as specified by string and then to offer the user a normal prompt session afterward. To state it a different way, CMD.EXE behaves as you would expect, except that it first executes the command as specified by **string.**

This is extremely powerful and flexible because **string** can be any command that you might have entered at the command line. It can be a program, command file, or an internal command. It can be several commands connected via the command operators as described in the Command symbols section earlier in this chapter.

Whether or not you have to enclose **string** in quotes depends on where you are giving this command. If you are typing into the parameters field of a program object, then you do not need quotes. If on the other hand you are typing at the command line itself, as would be the case for a START or a DETACH command, and you are using command symbols that you want to be passed to the new copy of CMD.EXE and not to be interpreted by the command line, then you need to enclose the whole command string in quotes.

You will need to use double quotes if the program name you are passing has an embedded blank in it. For example

```
CMD /K ""my program""
CMD /K ""my program" arguments"
```

- ✘ /C — The /C switch is identical to the /K switch except that CMD.EXE ends immediately after executing the commands specified by the string parameter.

COMP

Syntax: `COMP [source-path] [target-path]`

Description: The COMP command is used to compare files to see if they match exactly. The source path can contain any combination of drive letter, path, or file specification. If a drive letter is specified, it must be in combination with either a file specification or a path name. If no file specification is given, then a filename of * is assumed.

The target-path can be any combination of drive, path, or file specification. Again, if no file specification is given, a filename of * is assumed. Unlike the source-path argument, the target-path argument may contain just a drive letter.

Like the COPY, RENAME, and MOVE commands, the COMP command accepts wildcard characters in either the source or the target. Wildcard characters in the source control which files are found. Wildcard characters in the target specify how to construct the target filename given the source file that was found. In this manner you can, for example, compare all files with the name *.CMD to files with the name *.BAK.

The source- and target-path parameters are optional. If either one or both are not specified, then you will be prompted for them.

DISKCOMP

Syntax: `DISKCOMP [source-drive] [target-drive]`

Description: The DISKCOMP command is used to compare compatible-sized diskettes to see if the information on them is identical. This command does not support nonremovable media such as hard disks. The diskettes being compared must be of the same size. This means, for example, that you can only compare a 1.44MB diskette to another 1.44MB diskette.

The source-drive is not required to be specified if the current drive is already a diskette drive and the drive represents both the source and the target. In this case you can just type the DISKCOMP command with no arguments and you will be prompted to insert the source and target diskettes in the current drive at the appropriate time.

The target-drive is not required to be specified if the current drive is already a diskette drive and it represents the target-drive. In this instance you would need to specify the source-drive if it is not the same as the current drive. If the current drive is not a diskette or the current drive is not the target-drive, then you would need to specify both the source, and the target-drive.

The source- and the target-drive may be the same drive, such as:

```
DISKCOMP a: a:
```

in which case the DISKCOMP command will prompt you to switch between the source and target diskettes in the same drive.

It is useful to run this command following a DISKCOPY command to ensure that the diskette was duplicated successfully.

DISKCOPY

Syntax: `DISKCOPY [source-drive] [target-drive]`

Description: The DISKCOPY command is used to copy compatible-sized diskettes. This command does not support nonremovable media such as hard disks. The diskettes being copied must be of the same size. This means, for example, that you can only copy a 1.44MB diskette to another 1.44MB diskette.

The source-drive is not required to be specified if the current drive is already a diskette drive and the drive represents both the source and the target. In this case you can just type the DISKCOPY command with no arguments and you will be prompted to insert the source and target diskettes in the current drive at the appropriate time.

The target-drive is not required to be specified if the current drive is already a diskette drive and it represents the target-drive. In this instance you would need to specify the source-drive if it is not the same as the current drive. If the current drive is not a diskette or the current drive is not the target-drive, then you would need to specify both the source- and the target-drive.

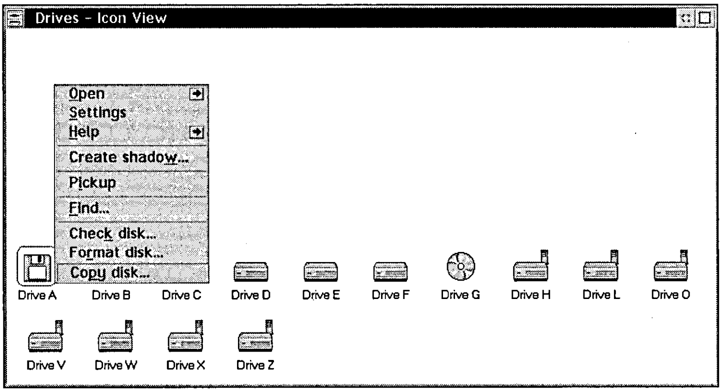
The source- and the target-drive may be the same drive, such as

```
DISKCOPY a: a:
```

in which case the DISKCOPY command will prompt you to switch between the source and target diskettes in the same drive.

The same function can be accomplished in the Workplace Shell by using the **Copy disk...** menu choice on a drive object in the Drives folder, as illustrated in Figure 15-3.

Figure 15-3. Copying a Disk from the Workplace Shell



E

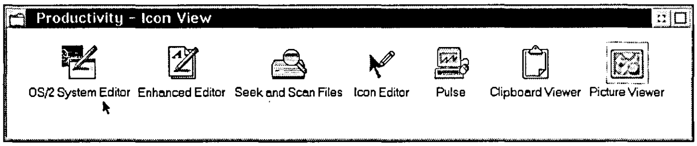
Syntax: `E [filename]`

Description: The E command is used to start the System Editor. You can choose to start it without a filename, in which case you will be editing a new empty file. Or you can start it by specifying a filename parameter, in which case you begin editing that file.

The filename parameter, if specified, must not contain any wildcard characters. The file indicated by the filename parameter should be an ASCII (otherwise known as a text) file. If you attempt to pass it a binary file, you will receive a cryptic message about embedded null characters being encountered.

This command can also be accessed from the Workplace Shell by opening the OS/2 System Editor object, which is located in the Productivity folder, as illustrated in Figure 15-4.

Figure 15-4. System Editor Object in Productivity Folder



EAUTIL

Syntax: `EAUTIL data-file [holdfile] /S [/R] [/P]`

Description: This form of the EAUTIL command is used split off all the extended attributes from a data file and place them in a holding file. This can be useful for either using a binary editor to examine the contents of the extended attributes or for sending the two files across a mailing system that does not support files with extended attributes.

In this case, the /S switch is required to indicate that you are performing the split operation. The other parameters and switches are

- **holdfile** — This file, if specified, is the name of the file that is created to house all the removed extended attributes. If this parameter is not specified, a holdfile that has the same name as the datafile is created in a subdirectory named EAS. The EAS subdirectory is in the same directory as the datafile. If the EAS subdirectory does not exist, it will be created.
- **/R** — The /R switch is used to indicate that the holdfile should be replaced if it already exists.
- **/P** — The /P switch is used to indicate that the extended attributes on the datafile should not be removed from the datafile. This function then performs a copy of the extended attribute information from the datafile to the holdfile.

Syntax: `EAUTIL data-file [holdfile] /J [/O or /M] [/P]`

Description: This is the join version of the EAUTIL command. It attaches the extended attribute information in the holdfile onto the datafile. The /J switch is required to indicate that this is the join version of the EAUTIL command. The additional parameters and switches are

- ✗ **holdfile** — If specified, this is the specific file to use to gather the extended attribute information from to attach to the datafile. If not specified, the holdfile is a file with the same name as the datafile, which must exist in a EAS subdirectory, which itself must exist in the same directory as the datafile.
- ✗ /O — Use the /O switch to indicate that the extended attributes contained in the holdfile should completely replace the extended attributes that may currently exist on the datafile. The implication of this is that all extended attributes on the datafile are first deleted before the extended attribute information in the holdfile is attached.
- ✗ /M — Use the /M switch to cause the extended attribute information in the holdfile to be merged with the extended attributes already on the datafile. The implications of this are

  1 All extended attribute information that exists in the holdfile that does not have a similarly named extended attribute on the datafile is simply added to the datafile.

  2 All extended attribute information that exists in the holdfile that does have a similarly named extended attribute on the datafile, replaces the one in the datafile with the one in the holdfile.

  3 All extended attribute information that exists on the datafile that does not exist in the holdfile is simply left alone.
- ✗ /P — The /P switch is used to indicate that the holdfile should not be deleted after the completion of the EAUTIL command.

The following example can be useful for adding one or more extended attributes to many files:

```
EAUTIL datafile holdfile /J /M /P
```

Now execute that command against each datafile to which you wish to add the extended attributes contained in holdfile.

The EAUTIL command itself does not support the use of wildcard characters. So you can use the FOR command to get the EAUTIL command to operate on several files at one time. For example,

```
FOR %I in ( * ) do EAUTIL %I /S
```

splits off the extended attributes from all files in the current directory.

```
FOR %I in ( config.sys autoexec.bat ) do EAUTIL %I /S
```

splits off the extended attributes only from the named files.

FDISK

Syntax: `FDISK [/QUERY] [/CREATE] [/DELETE] [/SETNAME:name] [/SETACCESS]`
`[/STARTABLE] [/FILE:filename] [/NAME:name] [/DISK:n] [/FSTYPE:x]`
`[/START:m] [/SIZE:m] [/VTYPE:n] [/BOOTABLE:s] [/BOOTMGR]`

Description: Although the FDISK command takes a large number of switches, most users will only run the FDISK command in a full-screen mode by not specifying any of the switches. This is the preferred way to run the FDISK command. The FDISK command is used to partition and manage the space on your hard disks. You use it to create primary and logical partitions and to assign them drive letters.

When trying to decide whether to create primary or logical partitions, remember that if you add a physical drive such as a CD-ROM, all the logical partitions will have their drive letters resequenced while the primary partition's drive letters are preserved. However, you can only have four partitions per hard drive at one time.

Although you will probably interact with the FDISK command without entering any switches, thus using its full-screen interface, you can use the following parameters with FDISK to automate some of your configuration tasks. If you enter any of the switches with FDISK, the full-screen interface will not be started.

- ✘ /QUERY — Used to display information about all your hard disk partitions.
- ✘ /CREATE — Used to create a new drive either in a primary or extended partition. The switch must be used in conjunction with the /VTYPE and the /DISK switches.

- **/DELETE** — This is used to delete a logical drive or primary partition. Use this switch in combination with the /NAME switch to determine which drive or partition should be deleted. You can also specify /DELETE:ALL in combination with the /DISK command to indicate that all drives and partitions on the specified disk should be deleted.
- **/SETNAME:name** — Use this switch to set the name of a primary partition or logical drive as seen in the Boot Manager window. It also marks the drive as startable.
- **/SETACCESS** — This switch is used to mark which primary partition on the disk will be accessible. Only a single primary partition on the same disk can be accessible at one time. By accessible, I mean that it has a drive letter associated with it. Use this in combination with other switches to set which partition is being made accessible. Otherwise it can be used to toggle between partitions.
- **/STARTABLE** — Use this switch in combination with other switches to mark a particular drive letter as startable. A startable partition is one that an operating system such as DOS or OS/2 can be started from.
- **/FILE:filename** — Specifies the name of a file that should be read to in switches to the FDISK command. If you put multiple switches on the same line, then they must be separated by commas.
- **/NAME:name** — The /NAME parameter is used to indicate the name of a partition in combination with other FDISK switches to perform actions on that named partition. The name option can be up to eight characters, and it is case sensitive.
- **/DISK:n** — The /DISK switch is used to augment additional switches to indicate which disk they should apply to.
- **/FSTYPE:*x*** — The /FSTYPE switch is used to augment additional switches to indicate what the file system type of the drive should be used in performing the FDISK action. This switch cannot be used in conjunction with the /SETACCESS command. The *x* option can be DOS, FAT, IFS, Free, or other.

- /START:*m* — Use the /START switch to indicate the starting location of the partition that you are attempting to modify with other switches. The *m* option can be either the letter *t*, which indicates that the partition should start at the top of the free space on the physical disk; or the letter *b*, which indicates that the partition should start at the beginning of the free space.
- /SIZE:*m* — Use the /SIZE switch, in combination with other switches, to indicate the size of the partition. The size, as represented by the *m* option, represents a numeric value in megabytes. For example, /SIZE:5 means a 5MB partition.
- /VTYPE:*n* — This switch indicates the type of partition that is being acted on by one of the other action switches. The *n* option has one of the following values:
    - 0, indicating that the space is not useable
    - 1, indicating that the space is a primary partition
    - 2, indicating that the space is a logical drive within an extended partition
    - 3, indicating that the space is free and can be used to create a primary or extended partition.
- /BOOTABLE:*s* — Used in conjunction with the /QUERY switch, the /BOOTABLE switch lets you look for those partitions that either are or are not bootable (or startable). The values of *s* can be 0 for not bootable or 1 for bootable.
- /BOOTMGR — The /BOOTMGR switch is used to indicate that the partition to be created or acted upon is the special Boot Manager partition. The Boot Manager partition is a 1MB partition, which when booted, allows you to select which operating system you want to boot.

FDISKPM

Syntax: `FDISKPM`

Description: The FDISKPM command performs the identical function as the FDISK command, except it interacts with the PM interface. Unlike the FDISK command, the FDISKPM command does not accept any switches.

The FDISKPM command can be accessed from the Workplace Shell through the **Create partition...** menu choice on the Drive folder.

FIND

Syntax: `FIND "string" [/V] [/C] [/I] [/N] [filename]`

Description: The FIND command is used to search for the string argument within the file specified by the file argument. The file argument must contain a filename with no wildcard characters. You may use a drive and or path specification before the filename. Since wildcard characters are not allowed, you may use the FOR command in combination with the FIND command to search multiple files. The following example shows how to search multiple files in this manner.

```
FOR %I in ( *.cmd ) do FIND "echo" /i %I
```

The FIND command can also be used as a filter. That is, it can take its input from standard input and write its output to either standard output or standard error. Another way to think of it is that you can use the redirection and piping characters before or after a filter command.

Since the string parameter must be enclosed in quotes, you use a backslash to indicate that an embedded quotation character is part of the search. If the backslash is part of the search, you need to use two backslashes together.

The following switches can augment the FIND command.

- ✘ /V — This is used to reverse the meaning of the find operation so that all lines that do *not* contain the search string are found.
- ✘ /C — This parameter will display a count of the number of lines that contain a match instead of the number of found lines.
- ✘ /I — Causes the search to be done without regard to the case of the string. For instance, if you were looking for the word "PATH" inside the file CONFIG.SYS and the lowercase form "path" occurred, it would not be found unless you specified the /I switch.
- ✘ /N — Puts the line number of the line in which the match occurred at the beginning of the output line.

When FIND gets its input from the filename parameter (see the following example), it will display a few blank lines followed by the filename before the actual found lines are displayed.

Example: The following is an example of the FIND command.

```
FIND /I "path" config.sys
```

If you just want to have the found lines displayed, then either one of the following examples will work:

```
FIND /I "path" < config.sys
TYPE config.sys | FIND /I "path"
```

FORMAT

Syntax: `FORMAT [/ONCE] [/4] [/T:tracks] [/N:sectors] [/F:size] [/FS:name] [/L] [/Q] [/V:label]`

Description: The FORMAT command is used to prepare a disk or diskette for use before any data can be written to it for the first time. The FORMAT command can also be used as an erase utility to erase the entire contents of the disk or diskette.

If you have used the FDISK or FDISKPM commands to create a new partition or logical drive, the partition or logical drive must be formatted before it can be used for the first time. The same is true for unformatted diskettes. I strongly suggest purchasing the preformatted diskettes.

The following switches are used to modify the behavior of the FORMAT command:

- /ONCE — Use this switch if you know for sure you only want to format one disk or diskette. Otherwise, the FORMAT command will prompt you to format another disk or diskette after it finishes formatting.
- /4 — Use this switch to format a 360KB diskette in a 1.2MB diskette drive. The catch is that the diskette can only then be used reliably in a 1.2MB diskette drive. You may have trouble reading or writing to it if you attempt to use it in a 360KB drive.
- /T — Used in combination with the /N switch, the /T switch specifies the number of tracks that the disk(ette) should contain. If /T is not specified and /N is, then the implicit value is 80 tracks.

- ✗ /N — This switch, used in combination with the /T switch, specifies the number of sectors on the disk(ette).

  Normally the FORMAT command can determine /N and /T values based on the disk(ette) size. You should only have to specify the /N or /T parameters if you attempt to format a diskette of one size in a drive of another — for instance, if you want to format a 720KB diskette in a 1.44MB drive.

  The following list should be used for determining the proper values of the /N and /T switch combinations:

5.25-Inch Diskettes:		
360KB	/T:40	/N:9
1.2MB	/T:80	/N:15

3.5-Inch Diskettes:		
720KB	/T:80	/N:9
1.44MB	/T:80	/N:18
2.88MB	/T:80	/N:36

- ✗ **/F:size** — The /F switch is a faster means of specifying the disk size to be formatted. It can be used instead of the /N and /T switch combination. Size must be one of

  5.25-Inch Diskettes:

  360, 360K, or 360KB

  1200,1200K, 1200KB, 1.2, 1.2M, or 1.2MB

  3.5-Inch Diskettes:

  720, 720K, or 720KB

  1440, 1440K, 1440KB, 1.44, 1.44M, or 1.44MB

  2880, 2880K, 2880KB, 2.88, 2.88M, or 2.88MB

- **/FS:name** — Used to specify which file system should be used on the disk. For diskettes, only the FAT file system is supported. For hard disks, you can specify either FAT or HPFS. If the disk was previously formatted, the default is the prior file system on the disk. For diskettes, the default is always FAT.
- **/L** — Specifies that you want the long formatting performed on hard disks and Read/Write optical diskettes.
- **/Q** — Used to quickly reformat a diskette. This can only be used on a diskette that has been formatted once without this switch. Since using this switch causes the FORMAT command to not look for bad areas on the diskette, you should be confident that the diskette is in good condition before using this switch.
- **/V:label** — Used to specify the new volume label for the disk(ette) if you do not want to be prompted for one.

If you want to format a 720KB diskette in a 1.44MB drive, assuming the diskette is in drive A:, you would enter one of these FORMAT commands:

```
FORMAT A: /T:80 /N:9
FORMAT A: /F:720
```

If you want to format your hard disk using HPFS, then you would enter:

```
FORMAT D: /FS:HPFS
```

Developer's Desktop

I use the following command to quickly format a diskette rather than trying to erase all the files on the diskette. It is much faster this way:

```
FORMAT A: /Q /ONCE
```

HELP

Syntax: `HELP [ON] [OFF] [[BOOK] subject] [message number]`

Description: Use the HELP command to

- Control the HELP bar at the top of the window.
- Get help on a subject in the on-line documentation.
- Get help on a message number received at the command line.

The following explains the parameters in more detail:

- ON — Turns on the HELP bar at the top of the window or full-screen session.

- **OFF** — Turns off the HELP bar at the top of the window or the full-screen session.
- **[book] subject** — Displays help on the subject. Book can be the name of any file that has an .INF extension. Specify the book name if the subject will not be found in the Command Reference (CMDREF.INF).
- **message number** — Displays help on the message number received at the command line if the message was prefixed by a three-character message filename.

The HELP controls the ON/OFF setting by simply setting the value of the PROMPT environment variable.

Warning

If you have built up your own custom PROMPT environment variable, then using the HELP command with either the ON or OFF parameters will destroy your custom PROMPT environment variable.

Tip

You can use a SET command similar to the following to save and later restore your custom PROMPT.

```
SET MYPROMPT=%PROMPT%
SET PROMPT=%MYPROMPT%
```

To search for help on the COPY command, use the following form of the HELP command:

```
HELP COPY
```

Tip

But if you want to get help on the DO command in REXX, then you have to specify the book name as follows:

```
HELP REXX DO
```

If you want to get help on a system error message you received, such as SYS0002, then you would use help in one of the following ways:

```
HELP 2
HELP SYS0002
```

ICONEDIT

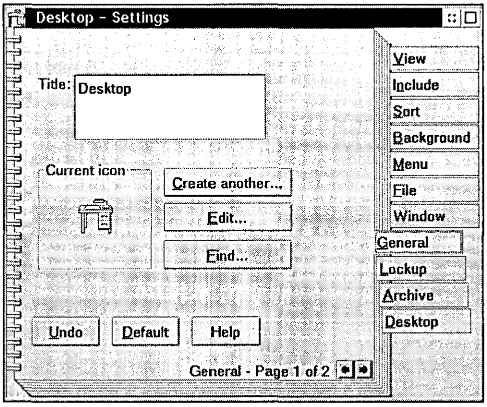
Syntax: `ICONEDIT [filename]`

Description: The ICONEDIT command is used to edit or create the following types of files:

- Icons

- ✗ Pointers
- ✗ Bitmaps

You can use icons to customize the appearance of your Workplace Shell objects. To edit the icon of an existing Workplace Shell object, you can launch the ICONEDIT utility by pressing the **Edit...** button on the General page of the object's Settings Notebook, as illustrated in Figure 15-5.


Desktop - Settings

Title: Desktop

Figure 15-5. Edit Button on General Page to Access Icon Editor

You can use pointers to customize the mouse pointer that is displayed when the mouse is over various parts of your window. After creating or editing your pointer with the ICONEDIT utility, you would add the pointer and select it using the Pointer page on the mouse object in the System folder, as illustrated in Figure 15-6.

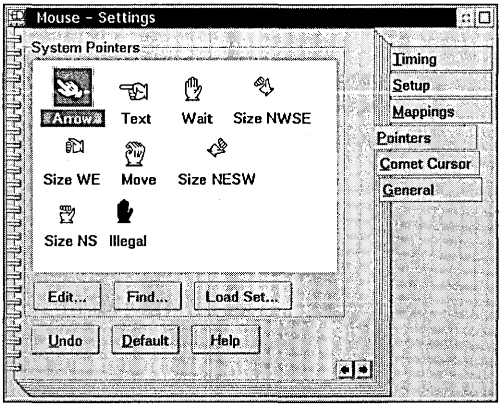

Mouse - Settings

Figure 15-6. Pointer Page on Mouse Object

You can use bitmaps for decorating the background of your Desktop or other Workplace Shell folders.

You can also start the ICONEDIT utility from the Icon Editor object in the Productivity folder.

The ICONEDIT command will only accept a single filename as a parameter. It cannot contain wildcards. If you want to create a new icon, pointer, or bitmap, run the ICONEDIT utility with no parameters and select **New** from the File menu.

KEYB

Syntax: `KEYB [country] [region]`

Description: When used without any parameters, the KEYB (think of it as the keyboard) command displays the country and region information that is used to interpret your keyboard.

The country parameter is used to specify what country information your keyboard is labeled with. Table 15-6 shows the available country settings.

Table 15-6. Country Parameters for the KEYB Command

AR	Arabic
BE	Belgium
BR	Brazil
CF	Canada (French)
CS	Czechoslovakia
DK	Denmark
SU	Finland
FR	France
GR	Germany
HE	Hebrew
HU	Hungary
IS	Iceland

IT	Italy
LA	Latin America
NL	Netherlands
NO	Norway
PL	Poland
PO	Portugal
SP	Spain
SV	Sweden
SF	Switzerland (French)
SG	Switzerland (German)
TR	Turkey
TW	Republic of China
UK	United Kingdom
US	United States
YU	Yugoslavia

In some countries, there is not a value that is applicable for the entire country. In that case you need to code the region parameter in addition to the country parameter. If the country parameter is specified by itself, then the first region value is used as the implicit specification for the region. Table 15-7 lists the available region parameters.

Table 15-7. Region Parameters for Use with the KEYB Command

243, 245	Czechoslovakia
189, 120	France
141, 142	Italy
166, 168	United Kingdom

You would need to use the KEYB command when your keyboard does not match the country-specific version of the operating system.

Developer's Desktop

I use the KEYB command when I am testing my code with a non-U.S. version of OS/2. I load that version of OS/2 on my computer and use the appropriate KEYB command to let the operating system know that I am using a U.S. keyboard. If I fail to do that, the key I press on the keyboard will not be the same character that I will see on my screen.

LABEL

Syntax: `LABEL [drive:][label]`

Description: Use the LABEL command to specify the volume label for the drive. If the drive is not specified, then the label is applied to the current drive. If the label is not specified, then you will be prompted to enter it.

Warning

If you specify both the drive and the label, then do not separate them with any spaces. The label should immediately follow the ":" of the drive letter. The label can be up to 11 characters long. If it contains blanks or other special characters, then enclose the label in quotes.

Example: In the following example, the LABEL command is used to set the volume label on a diskette in drive A: to the label "my label."

```
LABEL A:"my label"
```

MAKEINI

Syntax: `MAKEINI ini-file resource-file`

Description: The MAKEINI command is used to either create or modify the contents of an INI file from the specified resource file. The following examples demonstrate typical uses of the MAKEINI command.

```
MAKEINI OS2.INI INI.RC
```

The above command has two possible uses. It can be used to create an entirely new OS2.INI from the INI.RC file. The INI.RC file is created by the OS/2 install program. It can also be used to add back, or merge, the information in the INI.RC file to the OS2.INI file. You might use this command if you would like to have an entirely new desktop built for you.

```
MAKEINI OS2SYS.INI INISYS.RC
```

This command works the same as the example above, except it is either creating or merging the contents of the INISYS.RC file into the OS2SYS.INI file.

```
MAKEINI OS2.INI LOCK.RC
```

Use the example above if you forgot your password and you wish to reset the password back to no password. What it is really doing is removing the password information from your OS2.INI file.

MODE

Syntax: `MODE COMn[:] [,baud[,parity[,databits[,stopbits[,P]]]]] [,TO=state] [,XON=state] [,IDSR=state] [,ODSR=state] [,OCTS=state] [,DTR=state] [,RTS=state] [,BUFFER=state]`

Description: This syntax of the MODE command is used to set the asynchronous communications modes for a communcations device that may be connected to your communications ports such as COM1 or COM2.

The following parameters can be used with this form of the mode command:

- ✘ **COMn** — The communications port that the MODE command should be applied to, such as COM1, COM2, etc. The colon following the port name is optional.
- ✘ **baud** — Indicates the transmission rate of data through the port. The values can be one of 110, 150, 300, 600, 1200, 1800, 2400, 3600, 4800, 7200, 9600, 14400, 19200, 38400, or 57600. The default value is 1200.
- ✘ **parity** — Indicates the type of parity checking that is to be done on data through this port. The values can be one of N for none, O for odd, E for even, M for Mark, or S for space. The default for this parameter is even parity.
- ✘ **databits** — Indicates the number of bits that are used to represent a single piece of data, such as an individual character. The values can be one of 5, 6, 7, or 8 databits. The default value for this parameter is 7.
- ✘ **stopbits** — Indicates the number of bits that are used to indicate the end of a single piece of data. The values can be one of 1, 1.5, or 2. If databits is 5, then stopbits can be either 1 or 1.5. If stopbits is 1.5, then databits must be set to 5. If the baud parameter is set to 110, then the default for stopbits is 2. For all other values of the baud parameter, the default value for stopbits is 1.
- ✘ **P** — In a DOS session, the use of the P parameter indicates that the operating system should time-out in 30 seconds when the application is attemting to write directly to the hardware.

- ✘ TO — Set the TO parameter to ON to indicate the the system should wait indefinitely for a response from the communications port. A value of OFF, which is the default, indicates that the system will stop waiting after 60 seconds.
- ✘ XON — Set the XON parameter to ON when the communications protocol calls for the use of control flow characters. Otherwise set the value to OFF, which is the default.
- ✘ IDSR — Set the IDSR parameter to ON when the communications protocol calls for the use of the DSR bit for controlling the data flow into the communications port. Otherwise the value of OFF, which is the default, should be used.
- ✘ ODSR — Set the ODSR parameter to ON when the communications protocol calls for the use of the DSR bit for controlling the data flow out of the communications port. Otherwise the value of OFF, which is the default, should be used.
- ✘ DTR — The DTR parameter allows you to set the value of the Data Terminal Ready state to one of ON, OFF, or HS. ON is the default state.
- ✘ RTS — The RTS parameter allows you to set the value of the Request To Send state to one of ON, OFF, HS, or TOG. The default value is ON.
- ✘ BUFFER — The BUFFER parameter allows you to set the state of hardware buffering during communications. The value can be one of ON, OFF, or AUTO. To permit maximum performance, you should set the value to AUTO.

Syntax: `MODE [CONx] display[,rows]`

Description: This syntax for the MODE command is used to set the mode of your display device by allowing you to

- ✘ Switch display adapters.
- ✘ Set the display mode of your color graphics adapter.
- ✘ Set the number of character lines on the screen.

The following parameters are used with this syntax of the MODE command:

- ✘ CONx — On systems with more then one display adapter, the CONx parameter is used to indicate which adapter the MODE command is being issued for. x is the display number beginning with 1.

  On systems with only one display adapter, this parameter can be ignored.

- **display** — The display parameter is used to specify the display mode. It can be one of the following values: 40, 80, 132, BW40, BW80, BW132, CO40, CO80, CO132, or MONO.

  The number indicates the number of columns of characters. BW stands for black-and-white mode. CO stands for color mode.

- **rows** — The rows parameter, which must be used in combination with the display parameter, sets the number of rows to be displayed. Some common values are 25, 43, and 50. In an OS/2 Windowed Command Prompt, use of the rows parameter sets the window size and can be almost any value.

Syntax: `MODE LPTn or PRN [chars[,lines[,P]]]`

Description: This syntax of the MODE command is used to set the mode of a printer connected through a parallel port.

The following parameters can be used with this MODE command syntax:

- **LPTn** — The LPTn parameter is used to indicate which parallel port the mode command should affect. For example, it could be LPT1 or LPT2.
- **PRN** — You can use the parameter PRN instead of LPT1.
- **chars** — The chars parameter is used to specify the number of characters per line. Its value can be either 80 or 132, with 80 being the default.
- **lines** — The lines parameter is used to specify the lines per inch of vertical spacing. The value can be either 6 or 8, with a default of 6.
- **P** — The P parameter sets the state of the system infinitely retry to send the data to the parallel port when a time out is received.

Syntax: `MODE DSKT [VER=state]`

Description: This syntax of the MODE command displays or sets the diskette verification state of the system. Without the VER parameter, issuing the command MODE DSKT causes the diskette verification state of the system to be displayed. Use the VER parameter with a value of ON or OFF to set the diskette verification state of the system accordingly.

MORE

Syntax: `MORE`

Description: MORE is a pure filter command. It takes no arguments. It receives its input from standard input and writes to standard output. What it does is prompt you to continue whenever the number of lines it has written fills the current screen. By using the MORE filter, you can be sure that you see all the information that is being displayed.

Here are some examples of filtering the information through the MORE filter:

```
MORE < C:\CONFIG.SYS
TYPE C:\CONFIG.SYS | MORE
```

PATCH

Syntax: `PATCH filename`

Description: The PATCH command is a binary editor that can be used to modify the contents of a file. Enter the PATCH command followed by the filename to interactively modify the file, entering hexadecimal values.

Syntax: `PATCH filename /A`

Description: Use this form of the PATCH command when you receive a software patch from IBM. The filename parameter in this case is the file received from IBM. It contains the information about what file is to be patched and what the patching instructions are. All the updating is done without any interaction on your part.

PMCHKDSK

Syntax: `PMCHKDSK [drive:]`

Description: The PMCHKDSK command provides a PM front end to the CHKDSK command. See the CHKDSK command for more information, or run the PMCHKDSK command and click on the HELP button.

If you do not specify the drive parameter, then PMCHKDSK operates on the current drive.

The PMCHKDSK command can also be run from the Workplace Shell by choosing the **Check disk...** menu choice on a drive object in the Drives folder, as illustrated in Figure 15-7.

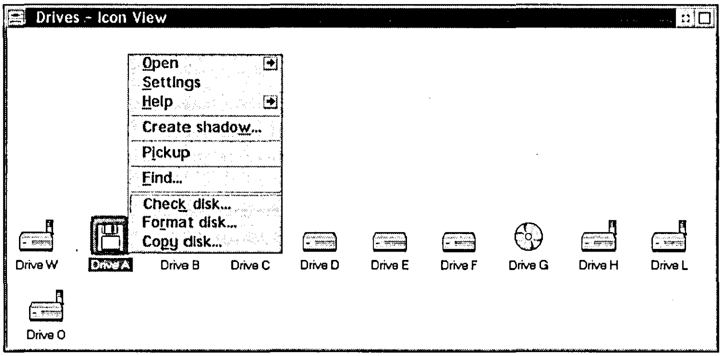


Figure 15-7. Check Disk Menu Choice on Disk Object

PMFORMAT

Syntax: `PMFORMAT [drive:]`

Description: The PMFORMAT command provides a PM front end to the FORMAT command. See the FORMAT command for more information, or run the PMFORMAT command and click on the HELP button.

If you do not specify the drive parameter, then PMFORMAT operates on the current drive.

The PMFORMAT command can also be run from the Workplace Shell by choosing the **Format disk...** menu choice on a drive object in the Drives folder, as illustrated in Figure 15-8.

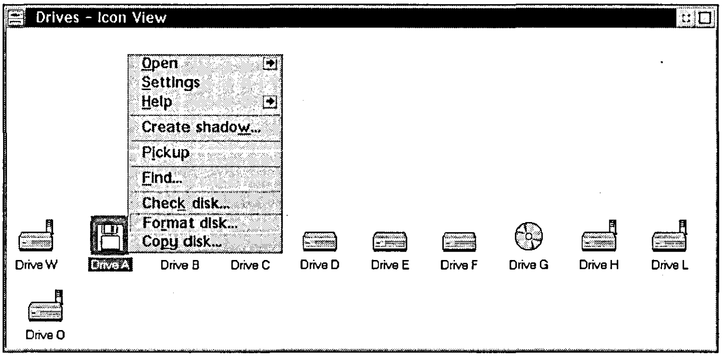


Figure 15-8. Format Disk Menu Choice on Disk Object

PMREXX

Syntax: `PMREXX rexx-command-file arguments`

Description: Use the PMREXX command to run REXX command files in a PM window. Certain REXX command files can only be run using PMREXX. These command files use extensions to REXX that allow them to display dialog boxes or other PM windows.

Tip

Many people have asked for an OS/2 Window that has an infinite history. They are looking for the ability to always scroll back and see the information that went off the top of the screen. Furthermore, they would like the ability to copy the information from the window back to the command line. By creating the following REXX command file and then running it using PMREXX, you can achieve that result:

```
/* */
"cmd"
```

Now assuming that you named the command file PMCMD.CMD, then execute it as follows:

```
PMREXX PMCMD.CMD
```

The REXX command file parameter must be a filename that contains a valid REXX command file. You are not required to include the extension in the filename. If you do not specify it, a .CMD extension is assumed.

PRINT

Syntax: `PRINT filename [/D:device] [/B]`

Description: The command is used to PRINT the file specified by the filename parameter to the printer. By default, the printer is the LPT1 device. If spooling is enabled, then the PRINT command sends the file to the spool queue for that printer device. The spooler then prints the job in the background. You can specify multiple filename parameters on one print command and each filename parameter may contain wildcard characters.

The following switches can be used to augment the default behavior of the PRINT command:

- **/D:device** — This switch specifies the printer device, such as LPT1, LPT2, COM1, and so on.

- /B — This switch indicates that the file should be printed in binary mode. As described in the COPY command, binary mode means that the command does not look for embedded CTRL-Z characters in the file. If this switch is not used, then printing stops at the first CTRL-Z character in the file.

Syntax: `PRINT [/C or /T] [/D:device]`

Description: This form of the PRINT command is used to control the printer queue for the device specified by the /D switch. If the /D switch is not specified then the device defaults to LPT1.

Use **PRINT /C** to cancel the current job that is printing.

Use **PRINT /T** to cancel the current job and all other jobs that are waiting to be printed.

Note

In this context, cancel means that the spooler will stop sending information about the print job to your printer. However, your printer may have a large amount of memory, in which case it will continue to print the information that has already been sent and is sitting in its memory.

Developer's Desktop

I usually solve that problem on my laser printer by pulling out the paper drawer enough so that it will not feed and then waiting until it gives me an "out of paper" error. I then power the printer off and on to clear its memory. Yes, I do put the drawer back in before turning the printer on. One of these days I will look up the operator panel combination on the printer for instructions on clearing its memory.

PSTAT

Syntax: `PSTAT [/C][/S][/L][/M][/P:pid]`

Description: Use the PSTAT command to gain information about the various processes that are running in the system. When used without any of the switches, the PSTAT command displays the following information about each process:

- The current processes
- The threads used by each process
- A list of system semaphores in use by each process
- Shared memory for each process
- The dynamic-link libraries (files which end in .DLL) being used by each process

Use the following switches to modify the behavior of the PSTAT command:

- /C — Only displays information about processes and threads.
- /S — Displays a listing of the system semaphores in use by each thread.
- /L — Displays a listing of the Dynamic Link Libraries in use by each process.
- /M — Displays information about shared resource per process. This would include things such as file handles and shared memory.
- **/P:pid** — Limits the display of the information to the process with the Process Identification Designation number represented by the pid option to the /P switch.

RECOVER

Warning

The only thing that should be said about the RECOVER command is that you should delete it off your hard disk. It is too dangerous a command.

Tip

You should go out and buy a good set of disk utilities.

REPLACE

Syntax: `REPLACE source-path [target-path] [/A] [/S] [/P] [/R] [/W] [/U] [/F]`

Description: The REPLACE command is one of my favorite external commands. It is also one of the least utilized of all the commands. Think of it as a smart copy utility. It has two methods of operation:

1. It is used to replace all occurrences of the specified source files on the target-path. What this means is that in order for the file to be copied from the source to the target, the file must already exist on the source and on the target.
2. It is used to add files that exist on the source-path but do not exist on the target-path to the target-path. When using this mode, files that already exist on the target are not disturbed. This is used to add files from the source to the target.

The following parameters and switches are used with the REPLACE command:

- **source-path** — The source-path parameter must include a filename. It can also additionally include a drive or a path specification. The filename may contain wildcard characters.

- **target-path** — The target-path may NOT contain a filename. It can be a drive and or path specification.
- /A — The /A switch indicates that you want to use the REPLACE command in the second mode. That is, you want to add files from the source that do not exist on the target.
- /S — The /S switch, which cannot be use in combination with the /A switch, says that all subdirectories below the specified target directory should be searched for files that match the source specification.
- /P — When you use the /P switch, you will be prompted before any files are copied.
- /R — Use the /R switch when the target file is marked read-only. The file will be copied over and then it will get marked read-only after the copy.
- /W — Prompts the user to press **Enter** before the replace operation begins. This is useful when REPLACE is used as part of an automated process to replace files from diskette.
- /U — The /U switch indicates that files should only be replaced on the target if the source file is newer. This is the update switch.
- /F — The /F switch is used to force the REPLACE command to fail if the extended attributes could not be copied to the target. Normally, the file is copied, and a working message is issued that the extended attributes where not copied.

Here is an example of how I used the REPLACE command every day while I was working on this book. I have two files associated with this chapter, CHAPT15.DOC and BOOK.DOT. After I first created these files, I used the COPY command to copy them to a diskette. Whenever I finished working on the chapter for the day, I used the following REPLACE command to update the copies on my diskette:

```
REPLACE * A: /U
```

The alternative is to use the following two copy commands:

```
COPY CHAPT15.DOC A:
COPY BOOK.DOT A:
```

Sometimes I worked on these files on a different computer, so when I wanted to transfer them back to the other computer, I just used the following REPLACE command:

```
REPLACE A:* C:\BOOK /U
```

The subtle beauty is that I can have lots of other files on the diskette in my A: drive, and I usually do, but the REPLACE command knows to only copy those files from the A: drive that also exist in the C:\BOOK subdirectory. Again, it saves me from having to type out the two copy commands. So as you can see, it is useful for keeping a set of files synchronized.

RESTORE

Syntax: `RESTORE [source-drive] [target-path] [/P] [/M] [/B:mm-dd-yy] [/A:mm-dd-yy] [/E:hh:mm:ss] [/L:hh:mm:ss] [/S] [/N] [/F] [/D]`

Description: The RESTORE command is the converse of the BACKUP command. It allows you to get files from your backup disk(ette)s to the target drive you indicate.

Tip

Now the thing to remember about the RESTORE command is that you cannot restore files from the backup disk(ette)s to an arbitrary path on the target. Files must be restored into the same directory they were backed up from. That means that if you want to restore specific files, you must specify the target directory to be the same as the directory the file existed in when it was backed up. Otherwise you will get a file not found message.

For example, assume that I backed up the file D:\BOOK\CHAPT15.DOC to diskette. Now I want to restore just that file, but let's say to the E: drive.

Example: The following example shows the command I need to restore the file to the E: drive.

```
RESTORE A: E:\BOOK\CHAPT15.DOC
```

Only after the file is restored can I then move it to another directory if I so choose.

Let's examine the switches that can be used in conjunction with this command:

- /P — The /P switch is used to force the RESTORE command to prompt you before restoring the file on to the target drive.
- /M — When the /M switch is used, only those files on the target that have been modified can be restored. This is useful if you are trying to restore only those files that currently exist on the target and have changed.

- **/B:mm-dd-yy** — The /B switch only allows files that exist on the target drive with a date on or before the date used as the option to the /B switch to be restored. So it is almost like the REPLACE command. The file must already exist on the target.
- **/A:mm-dd-yy** — Similar to the /B switch, the /A switch says that the target file must have a date on or after the date used with the /A switch.
- **/E:hh:mm:ss** — The /E switch only allows files that exist on the target drive with a time on or before the time used as the option to the /E switch to be restored.
- **/L:hh:mm:ss** — Similar to the /E switch, the /L switch says that the target file must have a time on or after the time used with the /E switch.
- **/S** — The /S switch is used to cause subdirectories on the backup disks to be searched for the files specified.
- **/N** — The /N switch causes only those files from the backup that do not already exist at the target to be restored. This is very similar to the add mode of the REPLACE command.
- **/F** — The RESTORE command will fail if you specify the /F switch and the target drive does not support extended attributes and the source files contain extended attributes.
- **/D** — The /D switch causes a list of files to be displayed of which files would have been restored without the files actually being restored. Use the /D switch when you want to be assured of which files will be restored before attempting the restore operation.

So now that we know about the /S switch, we can use it to modify the previous example to make our lives a little easier:

```
RESTORE A: E:\CHAPT15.DOC /S
```

Now to restore the complete backup we did, BACKUP D:\* A: /S in the discussion of the BACKUP command, we will use the following command:

```
RESTORE A: C:\* /S
```

REXXTRY

Syntax: `REXXTRY [rexx command]`

Description: Use the REXXTRY command without any arguments to experiment with REXX commands. Alternatively, you can enter the name of a REXX command as a parameter. If you run the REXXTRY command with a REXX command argument, it will execute that command and then exit.

SETBOOT

Syntax: `SETBOOT [/T:x] [/T:NO] [/M:m] [/Q] [/B] [/IBA:n] [/IBD:d] [/X:x] [/N:name]`

Description: The SETBOOT command is used to configure your Boot Manager partition from the command line without going into the FDISK command. You must specify one or more of the following switches with the SETBOOT command:

- **/T:n** — The /T switch is used to set the time-out value for the Boot Manager. The value for n specifies the number of seconds the Boot Manager should wait before starting the default operating system. A value of zero indicates that the default operating system should be started immediately without showing the Boot Manager screen. Use the value of NO to indicate that the Boot Manager screen should wait indefinitely until you make a selection.
- **/M:m** — The /M switch is used to set the mode of the Boot Manager screen to either normal or advanced. Use a value of a for advanced and n for normal mode.
- **/Q** — Use the /Q switch to query the values of your Boot Manager partition.
- **/B** — A shutdown of the operating system and a restart of the system occurs when you use SETBOOT with the /B switch.
- **/IBA:n** — Performs a shutdown of the operating system and then restarts the operating system located in the partition named by the n option. The name that you use for the n option is a name that appears on the Boot Manager screen.
- **/IBD:d** — Performs a shutdown of the operating system and the restarts the operating system located on the logical drive specified by the d option.
- **/X:x** — The /X switch is used to start the Boot Manager in either attended or unattended mode. Use a value of 0 for x for attended mode and a value of 3 for unattended mode.

SORT

Syntax: `SORT [/R] [/+n] [/Tn]`

Description: SORT is a filter command. It takes its input from standard input and sends its output to standard output. You typically use the redirection and piping symbols in conjunction with the SORT command.

The following switches can be used with the SORT command:

- ✗ **/R** — The /R switch is used to reverse the sort order. Normally SORT will sort in ascending order unless you specify the /R switch, in which will case it then sorts in descending order.
- ✗ **/+n** — The /+ switch is used to indicate which column the sorting should begin on. Use the n option to specify the column number.
- ✗ **/Tn** — The /T switch is used to specify how many columns should be skipped after a tab character is seen. The default value is zero.

SPOOL

Syntax: `SPOOL [/D:in-device] [/O:out-device] [/Q]`

Description: The SPOOL command can either be used to query the state of the spooler or to redirect the output of one device to another. For instance, if you want to redirect all output that an application sends to LPT1 to LPT2, then you can use the following SPOOL command:

```
SPOOL /D:LPT1 /O:LPT2
```

If you just want to query the state of the spooler, then you can use the following command:

```
SPOOL /Q
```

SYSLEVEL

Syntax: `SYSLEVEL`

Description: Use the SYSLEVEL command to view the revision level of all the major installed components of the operating system or of applications that have provided SYSLEVEL information.

SYSLOG

Syntax: `SYSLOG [/S] [/R] [/P:name] [/W:x]`

Description: You use the SYSLOG command to interact with the system error logging facility. Running the SYSLOG command with no arguments starts the error logging facility. Use the following switches to access the other SYSLOG functions:

- ✗ **/S** — Use the /S switch to suspend error logging.
- ✗ **/R** — Use the /R switch to resume error logging after it has been suspended.
- ✗ **/P:name** — The /P switch directs the SYSLOG command to use the filenamed by the name option as the file to store system error log information in.
- ✗ **/W:x** — The /W switch is used to set the size of the error log file in kilobytes. The default value is 64KB. The minimum value you can use is 4KB. You can use the /W switch to increase or shrink the current error log file.

TEDIT

Syntax: `TEDIT [filename]`

Description: Use the TEDIT command to edit files in an OS/2 window or an OS/2 Fullscreen.

Uncensored

The TEDIT command was added to the system to provide a means of editing your system configuration files when Presentation Manager is not available, such as starting OS/2 from diskette.

You can optionally start the TEDIT command with one or more filenames.

TREE

Syntax: `TREE [drive:] [/F]`

Description: The TREE command is used to display a listing of all the directories on either the default drive or the drive that you specify. Use the /F switch if you want to have a listing of all the files in each directory displayed also.

TUTORIAL

Syntax: `TUTORIAL`

Description: Use the TUTORIAL command to run the OS/2 Warp tutorial. This command can also be started by opening the Tutorial object in the Information folder, as illustrated in Figure 15-9.

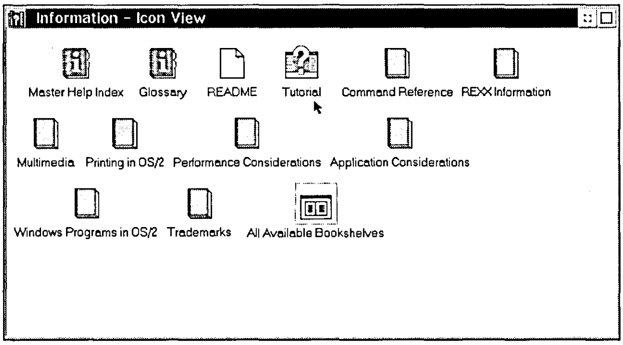


Figure 15-9. Tutorial Object in Information Folder

UNDELETE

Syntax: `UNDELETE [path] [/L] [/S] [/A] [/F]`

Description: OS/2 Warp has the ability to undelete files that have been erased, provided that you have enabled this option by setting the value of the DELDIR environment variable. The DELDIR environment variable is added to CONFIG.SYS by the OS/2 Warp installation utility but is commented out by use of the REM command. To activate this feature, remove the REM command from the beginning of the line and restart your system.

The path and filename argument represent the path and name of the file as it existed when it was deleted. It is recovered to the current drive and directory.

The following additional switches can be used with the UNDELETE command:

- ✗ /L — Use the /L switch to get a listing of the files that are available to be undeleted without actually undeleting them.
- ✗ /S — Use the /S switch to search subdirectories for the filename specified.
- ✗ /A — Use the /A switch to undelete all the requested files without prompting you for them if they are there to be undeleted.
- ✗ /F — Use the /F switch to force delete the files so that they can no longer be restored using the UNDELETE command.

UNPACK

Syntax: `UNPACK file-name [target-path] [/V] [/F] [/N:filename] [/SHOW]`

Description: Use the UNPACK command to decompress and copy a file from a compressed file or a compressed bundle file. You can specify the target-path for the file to be decompressed into, but you cannot specify the filename itself. If you do not specify the target-path on the UNPACK command, then the directory where the file is stored in the bundle file must exist in the default drive in order for the file to be unpacked. The filename is stored in the compressed file. The following switches can be used with the UNPACK command:

- **/V** — The /V switch causes the copied data to be verified to ensure that it was copied correctly.
- **/F** — The /F switch causes the copy operation to fail if the target device does not support extended attributes and the source file contains extended attributes.
- **/N:filename** — When the source file is the name of a bundle file, then use the /N switch to specify which file from the bundle file you want to extract and decompress.
- **/SHOW** — Use the /SHOW switch to show information about the contents of a bundle file.

A compressed file that can be decompressed with the UNPACK command contains at least one @ character in its file extension. For example,

```
UNPACK A:\BUNDLE /SHOW
```

displays the contents of the packed BUNDLE file on the installation diskette in drive A:.

```
UNPACK A:\BUNDLE /N:VIEW.EXE
```

unpacks the file VIEW.EXE from the BUNDLE file on the installation disk in the A: drive to the \OS2 directory on the default drive. To determine that it will go into the \OS2 directory, I used the /SHOW switch with the UNPACK command on the same BUNDLE file to see in what directory the file is stored in the BUNDLE file.

```
UNPACK A:\BUNDLE D:\TEMP /N:VIEW.EXE
```

unpacks the file VIEW.EXE from the BUNDLE file on the installation disk in the A: drive to the \TEMP directory on the D: drive.

VIEW

Syntax: `VIEW file-name [topic]`

Description: Use the VIEW command to display an on-line reference book with a file extension of .INF. If you want to look for a specific topic in the book, then specify the topic name also.

```
VIEW CMDREF COPY
```

The above example searches for the COPY command within the CMDREF.INF on-line reference.

Syntax: `VIEW /topic`

Description: Use this variation of the VIEW command to look for the specified topic in all on-line reference files that are listed in your BOOKSHELF environment variable.

```
VIEW /while
```

The command given above searches for help on the WHILE command. It was found in the on-line REXX reference.

WPDSINIT

Syntax: `WPDSINIT [/K] [/S] [/Q]`

Description: The WPDSINIT command is used to allow programs to communicate with the Workplace Shell using DSOM. DSOM stands for IBM's Distributed System Object Model. Under normal circumstances, you should not have to run this command unless your program documentation instructs you to. When executed without any switches, the WPDSINIT command displays a simple help message describing itself.

The following switches can be used with the WPDSINIT command:

- /K — The /K switch ends the WPDSINIT server process.
- /S — The /S switch starts the WPDSINIT server process.
- /Q — The /Q switch querys the state of the WPDSINIT server process.

XCOPY

Syntax: `XCOPY source-path [target-path] [/D:mm-dd-yy] [/S] [/E] [/P] [/V] [/A] [/M] [/H] [/T] [/R] [/O] [/F]`

Description: The XCOPY command is a variation on the COPY command that can be used to copy a subdirectory structure and have that subdirectory structure created at the target. Of course you can use it to copy files from a single source directory to an existing target directory, or it can create the directory for you.

The source-path and target-path parameters are any combination of drive, path, or filenames. If a filename is not specified, then a filename of * is assumed.

The directory structure at the target does not have to be at the same level as the directory structure at the source.

The following switches can be used with the XCOPY command:

- **/D:mm-dd-yy** — Copies only files which have been modified on or after the date specified.
- /S — Copies files that match the file portion of the source-path parameter from all sub-directories below the specified or implied source directory in addition to any occurrences of the file that exist in the source directory. This causes a matching subdirectory structure to be created below the target directory if any files are found.
- /E — When used with the /S switch, the /E switch causes empty subdirectories to be created at the target if the subdirectory was empty at the source.
- /P — The /P switch causes you to be prompted for confirmation before the file is copied.
- /V — The /V switch causes the data to be verified after the file is copied.
- /A — The /A switch causes only those files that match the source file specification and that have their Archive attribute enabled to be copied.
- /M — The /M switch works the same as the /A switch, except that it turns off the Archive attribute on the file at the source after the file has been successfully copied.

OS/2 WARP Developer's Desktop

I use this switch when I am trying to XCOPY files to a diskette. If all the files do not fit, I just keep running the same command over and over, using as many diskettes as it takes, until all the files have been copied.

- ✗ /H — The /H switch copies hidden files from the source to the target.
- ✗ /T — The /T switch copies system files from the source to the target.
- ✗ /R — The /R switch copies read-only files from the source to the target and has the file retain its read-only attribute at the target. If you do not use this switch, then read-only files will be copied but they will lose their read-only attribute at the target.
- ✗ /O — The /O switch allows read-only, system, or hidden files to be overlaid at the target.
- ✗ /F — The /F switch causes the XCOPY command to fail if the target drive does not support extended attributes and the source file contains them.

Here are some examples of the XCOPY command:

```
XCOPY C:\DATA1 D:\BACKUP\DATA2
```

If you assume that DATA1 and DATA2 are both filenames, then this command will copy the file C:\DATA1 to D:\BACKUP\DATA2. It will create the BACKUP directory if it does not exist. The XCOPY command will prompt you to state whether or not DATA2 represents the name of the target file or the name of the directory that should be created at the target to house the target file. If you answer that DATA2 is a file, then C:\DATA1 will be copied to D:\BACKUP\DATA2. If you answer that DATA2 is a directory, then it will copy the file C:\DATA1 to the file D:\BACKUP\DATA2\DATA1.

If you assume that DATA1 and DATA2 are directories, then it will copy all files from the DATA1 subdirectory to the DATA2 subdirectory. The XCOPY command will prompt you to state whether or not DATA2 represents the name of the target file or the name of the directory that should be created at the target to house the target files. If you answer that DATA2 is a file, then every file from the C:\DATA1 directory will be copied to a file called D:\BACKUP\DATA2. The end result of that is that D:\BACKUP\DATA2 is just a copy of the last file from C:\DATA1 that was copied. If you answer that DATA2 is a directory, then every file from C:\DATA1 is copied to the directory D:\BACKUP\DATA2. The file will have the same name in the DATA2 directory that it had in the DATA1 directory.

```
XCOPY C:\DATA1 D:\BACKUP\DATA2\*
```

This command is the same as the one prior to it, but using this syntax instead will prevent the XCOPY command from prompting you to determine if DATA2 is a directory or a file.

```
XCOPY C:\DATA1 D:\BACKUP\DATA2\* /s
```

Same as the example before it, except that all files that exist in the subdirectories below DATA1 are copied over under DATA2, and the subdirectories will be given the same names they had at the source.

```
XCOPY C:\DATA\FILE.TXT D:\DIR1\DIR2\NEW.EXT
```

This is used to copy a single file from the source to the target. Both dir1 and dir2 will be created if they do not already exist.

XDFCOPY

Syntax: `XDFCOPY [source-drive or image file] [target drive or image file]`

Description: Use the XDFCOPY command to copy XDF-formatted diskettes. The source and targets of the copy can be either a diskette drive or an image file. An image file is a file that contains the diskette all packed together as a single file. So you can create an image file from diskette or you can create a diskette from an image file. You can also create a diskette from another diskette. For example,

```
XDFCOPY A: D:IMAGE
```

makes an image file on drive D: of the contents of the XDF-formatted diskette in drive A:.

```
XDFCOPY D:IMAGE A:
```

makes an XDF-formatted diskette in drive A: from the contents of the image file stored on drive D:.

Summary

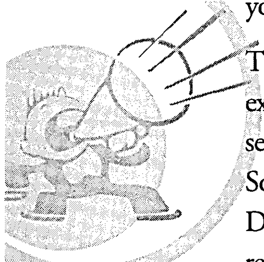
This chapter does not attempt to replace the on-line command reference that comes with OS/2 Warp. I have attempted to give you additional insight into the OS/2 commands as I know them. I also find it more convenient to be able to see a listing of all the commands and their parameters in one place than using all those hypertext links in the on-line command reference when I am attempting to get a broad overview of a subject. I have not explored several of the unique DOS mode commands, nor did I touch on any of the specific Windows commands in this chapter. I strongly suggest exploring the on-line command reference once you have found a command that you are interested in for other details, such as related commands.

Peter G. Magid is an Advisory Programmer on the Workplace Shell Development Team. As the User Interface Design Lead for the OS/2 Workplace Shell, Peter has had chief design responsibilities for what users see and how they interact with the Workplace Shell. Peter also plays an active role in reviewing and consulting on the content and direction of the Workplace Shell programming model, and he developed the code for the LaunchPad. Prior to working on the Workplace Shell, Peter was a team lead in the OS/2 commands and utilities group.

Chapter 16

DOS and WIN-OS/2 Settings

Pssst! Hey, Bub, Need a Setting?



This chapter describes the Disk Operating System (DOS) and WIN-OS/2 settings. You will not need to refer to this chapter unless you are having problems running DOS or Windows application programs.

DOS and WIN-OS/2 settings are used to modify environment settings in the DOS session and its associated Virtual Device Drivers (VDD). Some applications expect more advanced settings than what is defined as the default for the session. This chapter describes the DOS and WIN-OS/2 settings from both external and internal usage. Understanding the internal usage of the DOS and WIN-OS/2 settings should help you to feel more in control of the system.

The WIN-OS/2 settings enable you to customize or fine tune the session for the execution of DOS or Windows applications. The following settings are the standard settings for OS/2 Warp, but an installed vendor VDD can add extra DOS settings. Some DOS and WIN-OS/2 settings, like DOS_FILES, can be modified once the DOS session is running. Others, such as DOS_AUTOEXEC, cannot be modified for reasons that will be discussed later in this chapter.

Note

At present there is no way to modify a DOS setting across all Virtual DOS Machines (VDM) running or that will be run in the system. The CONFIG.SYS does provide limited initial setting of a few DOS setting values, such as DOS=UMB. After the CONFIG.SYS has been read when OS/2 starts, there is no way to modify a central VDM environment area to change DOS settings for all VDMs.

Overview Of DOS and Windows Support

The OS/2 operating system uses the concept of Multiple Virtual DOS Machines (MVDM), which runs in Virtual 8086 mode (V86). This processor mode of the Intel 386/486/586 architecture allows programs to run in protect mode but act as if they are running in real mode.

In OS/2 1.x, the processor was dynamically switched or rebooted at specified intervals between real and protect mode to allow real and protect mode programs to run at the same time. Running the processor in two different modes severely hampered any communication between protect and real mode subfunctions in the operating system. This switching also limited the overall performance of the user's system. Some DOS extenders do the same thing, so your system won't perform as well as it would if it were running in the same mode all the time.

V86 mode running in protect mode allows for easier communication to other protect mode programs such as the OS2KRNL. MVDM manages Virtual DOS Machines (VDM). The MVDM is invoked every time you bring up a DOS full-screen, DOS window, WIN-OS/2 full-screen, or WIN-OS/2 window. If no other real mode applications have been started, the MVDM and a single VDM session are brought up the first time a real mode application is started. After this initial startup point, only a new VDM session is started because the MVDM remains loaded.

Each VDM is thought to be a separate DOS session. These DOS sessions each have their own DOSKRNL to manage DOS functions, such as file system calls. This means that a separate DOS machine is booted every time you bring up a VDM session. What happens in one DOS machine will not affect any other DOS machine. This concept is different from running multiple DOS programs in a real DOS machine. In real DOS, you have to be concerned that multiple programs loaded in the 640K real mode memory will corrupt or conflict with RAM resources. With separate DOS machines, this corruption or conflict with RAM resources does not exist.

The DOSKRNL does not have its own file system as in real DOS but instead communicates to the OS2KRNL and uses its FAT or IFS file system, such as HPFS. This communication occurs when the DOSKRNL makes calls to MVDM, which manages access to the OS2KRNL functions running in protect mode. Limitations may be seen because of the lack of ownership of a file system in the VDM. Many DOS programs assume that the file system's internal tables may be modified; that is, the file system will automatically react because of the changes in these internal tables.

Uncensored

In OS/2, the file system's internal tables are actually found in protect mode. Any change made to the file system tables in real mode is not reflected in protect mode. This protection of the internal tables is for everyone's protection, not just to cripple DOS programs. The OS/2 file system only allows access to the file system via the documented Application Program Interfaces (APIs). These interfaces ensure that little damage is inflicted to the file system by an application.

DOS Protected Mode Interface (DPMI) is used to make calls into the OS2KRNL. WIN-OS/2, which runs Windows 3.1, 3.11, and Windows for Workgroups 3.1, 3.11 applications, uses DPMI exclusively to perform its functions.

Virtual Device Drivers (VDD) perform the necessary coordination of work that allows multiple DOS sessions to think they all have exclusive use of physical hardware as in real DOS. The VDDs talk to Physical Device Drivers (PDD) and MVDM to assist in this coordination. Default VDDs are included in every instance of a VDM.

Tip

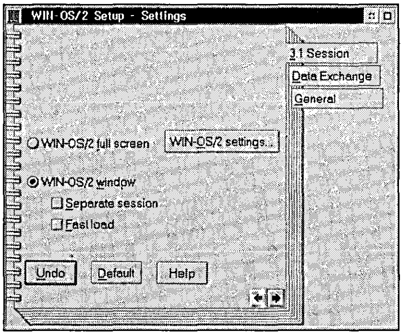
The default VDDs have to be installed in the \OS2\MDOS directory or else the VDM will not start. This is important to know if you decide to prune your hard disk for more space. It is best to leave the \OS2\MDOS directory intact.

The default VDDs do not appear in the CONFIG.SYS file. The VDDs that appear in the CONFIG.SYS file are installable VDDs. The installable VDDs are necessary for extra or nondefault operations in a VDM. For example, if you do not need COM support, then at install time select that you do not want COM support. This will cause install to not copy into CONFIG.SYS the necessary COM PDD and VDD support. Without this support you will not have COM support available to OS/2. If the COM.SYS PDD is copied to the CONFIG.SYS but not to the VCOM.SYS VDD you only have COM support in protect mode.

DOS or WIN-OS/2 Session Settings

You can find the default set of WIN-OS/2 settings in the Settings Notebook for the WIN-OS/2 Setup object in the System Setup folder (see Figure 16-1). To view or change the default settings, click on the **WIN-OS/2 settings...** push button on the 3.1 Session page of the Settings Notebook for the WIN-OS/2 Setup object.

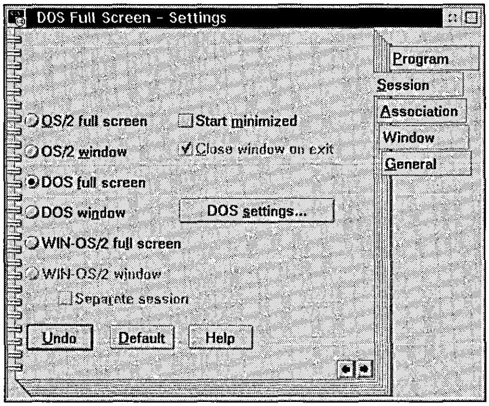
Figure 16-1.
Win-OS/2 Setup - 3.1 Session Page



○ WIN-OS/2 full screen
● WIN-OS/2 window
□ Separate session
□ Fast load

The Settings Notebook for a DOS Program Reference object contains the DOS settings (see Figure 16-2).

Figure 16-2.
DOS Full Screen - Session Page



○ OS/2 full screen
○ OS/2 window
● DOS full screen
○ DOS window
○ WIN-OS/2 full screen
○ WIN-OS/2 window
□ Separate session
□ Start minimized
☑ Close window on exit

The Settings Notebook for a Windows Program Reference object contains both the DOS settings and the WIN-OS/2 settings (see Figure 16-3).

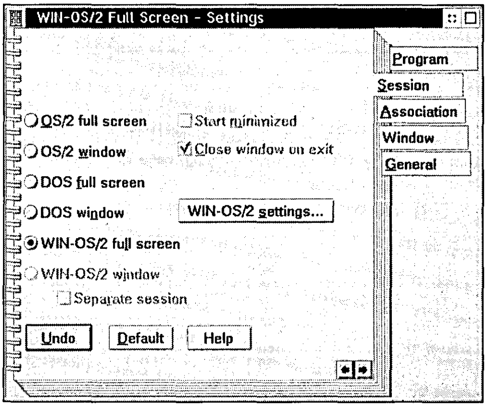


Figure 16-3.
Win-OS/2
Full Screen -
Session Page

WIN-OS/2 Full Screen - Settings

○ OS/2 full screen
□ Start minimized
○ OS/2 window
☑ Close window on exit
○ DOS full screen
○ DOS window
● WIN-OS/2 full screen
○ WIN-OS/2 window
□ Separate session

To view or change the DOS or WIN-OS/2 settings, open the Settings Notebook for the Program Reference object and click on the **DOS settings...** or WIN-OS/2 settings... push button on the Session page.

DOS or WIN-OS/2 Settings Categories

The DOS or WIN-OS/2 settings are broken down into categories to make it easier for you to find the settings you want to change. The dialog box shown in Figure 16-4 is a new, user-friendlier dialog box for OS/2 Warp. The titles for the categories were abstracted from the contextual help available for the DOS settings. Each category is a list of settings that will be displayed for you to change.

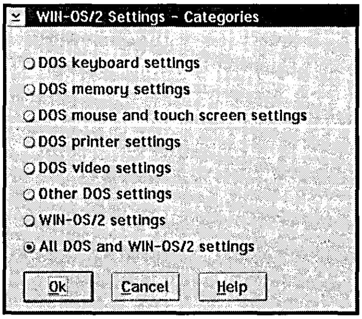


Figure 16-4.
WIN-OS/2
Settings –
Categories

WIN-OS/2 Settings - Categories

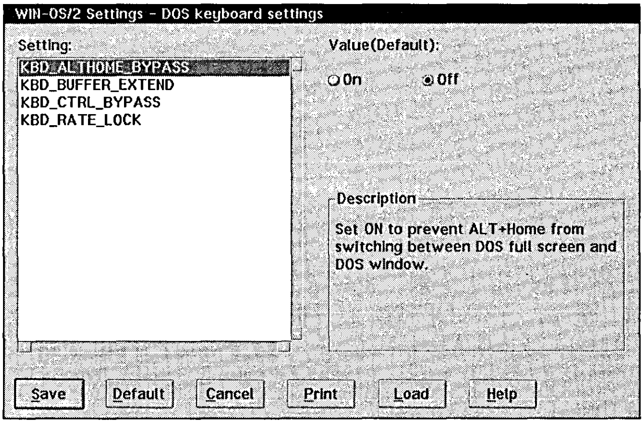
○ DOS keyboard settings
○ DOS memory settings
○ DOS mouse and touch screen settings
○ DOS printer settings
○ DOS video settings
○ Other DOS settings
○ WIN-OS/2 settings
● All DOS and WIN-OS/2 settings

This dialog box enables you to select which DOS or WIN-OS/2 settings are to be modified for this DOS or WIN-OS/2 session. Contextual help is available for each category by clicking on the **Help** push button in the dialog box.

DOS keyboard settings

The dialog box shown in Figure 16-5 enables you to modify the keyboard settings for a DOS or WIN-OS/2 session. New for OS/2 Warp is the capability of obtaining contextual help for each setting by clicking on the **Help** push button in the dialog box.

Figure 16-5. WIN-OS/2 - DOS Keyboard Settings Page



WIN-OS/2 Settings - DOS keyboard settings

Setting:
KBD_ALTHOME_BYPASS
KBD_BUFFER_EXTEND
KBD_CTRL_BYPASS
KBD_RATE_LOCK

Value(Default):
○ On ● Off

Description
Set ON to prevent ALT+Home from switching between DOS full screen and DOS window.

KBD_ALTHOME_BYPASS

This setting is registered by VDD VKBD.SYS as a default VDD. Use this setting to stop Alt-Home from switching between DOS full screen and a DOS window.

This setting is important, for example, when you are using terminal emulators. Many terminal emulators expect to have full control of the keyboard and any unique key sequences, except for Ctrl-Break or Ctrl-C. Since the OS/2 operating system uses Alt-Home to switch between the DOS full screen and the DOS window, interference can occur with the DOS application's use of this sequence. By disabling the operating system from *hooking* this sequence, the DOS application is able to process the sequence as if it were running on DOS.

Note

The concept of hooking is like getting a ticket at the deli case of a supermarket. By getting a sequentially numbered ticket, you reserve your place in line for access to the people working at the counter. Applications running in DOS can do the same thing by hooking the interrupt. In hooking the interrupt, the application asks the operating system for access to the location where the data comes into a particular device. By hooking the Alt-Home command, for example, the application will access the necessary data, do whatever it is supposed to in response to the command, and then will discard the data so another application or the operating system cannot act on it. This "access, process, and then discard" sequence is common when dealing with keyboard data but not always with other data from other devices.

KBD_BUFFER_EXTEND

This setting is registered by VDD VKBD.SYS as a default VDD. Use this setting to increase the size of the keyboard type-ahead buffer for a session.

Increasing the keyboard type-ahead buffer enables more keystrokes to be entered into the DOS application before the overflow beeps are sounded from the speaker.

KBD_CTRL_BYPASS

This setting is registered by VDD VKBD.SYS as a default VDD. Use this setting to have the operating system ignore one of the control-key sequences so that the DOS program can use it. For example, you can specify that Ctrl-Esc work as defined in the DOS program.

Using this setting is very important, for example, when using terminal emulators. Many terminal emulators expect to have full control of the keyboard and any unique key sequences, except for Ctrl-Break or Ctrl-C. Since the OS/2 operating system uses Ctrl-Esc to display the task list, interference can occur with the DOS application's use of this sequence. By disabling the operating system from hooking this sequence, the DOS application is able to process the sequence as if it were running on DOS.

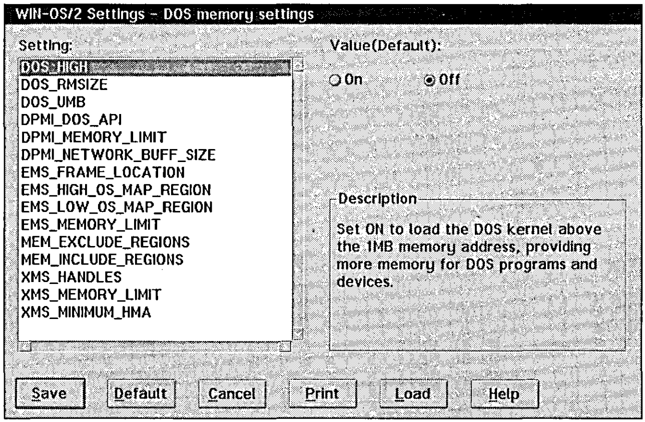
KBD_RATE_LOCK

This setting is registered by VDD VKBD.SYS as a default VDD. Use this setting to prevent a session from changing the system keyboard repeat rate.

Using this setting can help to keep keyboard repeat-rate continuity among all sessions of the OS/2 operating system. It can be frustrating if the DOS application dynamically modifies the keyboard repeat rate, and you are expecting the same rate among all sessions. Some DOS applications change the rate to enable more processor time to be scheduled for other functions. In the OS/2 operating system, the scheduling of processor time per function is a part of the kernel and not an application-changeable feature. Since the OS/2 operating system base device drivers and operating system primitives provide more aggregate functions than DOS, there is no need to slow down one function of the system just to assist another, less-capable part of the system.

DOS memory settings

The dialog box shown in Figure 16-6 enables you to modify the memory settings for a DOS or WIN-OS/2 session.



WIN-OS/2 Settings - DOS memory settings

Setting:

DOS_HIGH
DOS_RMSIZE
DOS_UMB
DPMI_DOS_API
DPMI_MEMORY_LIMIT
DPMI_NETWORK_BUFF_SIZE
EMS_FRAME_LOCATION
EMS_HIGH_OS_MAP_REGION
EMS_LOW_OS_MAP_REGION
EMS_MEMORY_LIMIT
MEM_EXCLUDE_REGIONS
MEM_INCLUDE_REGIONS
XMS_HANDLES
XMS_MEMORY_LIMIT
XMS_MINIMUM_HMA

Value(Default):

○ On ● Off

Description

Set ON to load the DOS kernel above the 1MB memory address, providing more memory for DOS programs and devices.

Figure 16-6. Win-OS/2 Settings - DOS Memory Settings

DOS_HIGH

This setting is registered by VDD VBIOS.SYS as a default VDD. Use this setting if you want the DOS kernel loaded above the High Memory Area (HMA) area above 1MB, allowing more memory for DOS programs.

Sometimes DOS applications assume that DOS will never be loaded high in memory and depend on modifying fixed DOS kernel locations on the fly. Making DOS run high invalidates the assumptions of these DOS applications and can also cause unpredictable results if the area being modified is used by another DOS application.

DOS_RMSIZE

This setting is registered by VDD VBIOS.SYS as a default VDD. Use this setting to decrease the amount of available memory to less than 640KB. Some DOS applications cannot handle having more than 640KB free because then their calculations for determining segment loading addresses overflow.

For example, there aren't very many applications that have trouble with having 640KB free, but the application PC/APL is one of them. It must have less than or equal to 512K or its internal segment load calculations will overflow.

DOS_UMB

This setting is registered by VDD VBIOS.SYS as a default VDD. Use this setting if you use a Terminate and Stay Resident program (TSR) or device driver that must own and manage Upper Memory Blocks (UMB). The DEVICEHIGH and LOADHIGH commands in CONFIG.SYS work only when DOS owns UMBs.

Tip

UMB area can become reduced if Expanded Memory Specification (EMS) and Extended Memory Specification (XMS) are set too high.

DPMI_DOS_API

This setting is registered by VDD VDPX.SYS as an installable VDD. Use this setting to control whether DOS API translation is enabled for DOS Protected Mode Interface (DPMI) applications in this session.

Tip

This is an important setting for running WIN-OS/2, which enables Windows 3.1 applications to run. WIN-OS/2 requires the DPMI to be enabled so that the enhanced-mode Windows applications can execute the protect-mode APIs they are expecting.

DPMI_MEMORY_LIMIT

This setting is registered by VDD VDPMI.SYS as an installable VDD. Use this setting to specify the amount of DPMI memory, in megabytes, available for a DOS session. Set the limit to 0 to disable DPMI.

Tip

If your Windows application is running out of resources, it could be because the DPMI_MEMORY_LIMIT is set too low. It is best to set the DPMI_MEMORY_LIMIT higher than necessary. Only the amount of DPMI memory needed will actually be allocated, so this value cannot hinder your system's available resources.

DPMI_NETWORK_BUFF_SIZE

This setting is registered by VDD VDPX.SYS as an installable VDD. Use this setting to control the size, in kilobytes, of the network translation buffer for DPMI applications in this session. This value should be provided with the network software you are running.

EMS_FRAME_LOCATION

This setting is registered by VDD VEMM.SYS as an installable VDD. Use this setting to change the location of the 64KB Expanded Memory Specification (EMS) frame. You need to change the EMS location if it conflicts with device drivers.

Device drivers and hardware tend to consume EMS memory area, so before setting this value it is best to run a hardware reference diskette to know the amount of available EMS memory. (See EMS in the Glossary of Terms section of this chapter for a further explanation of what this entails.)

EMS_HIGH_OS_MAP_REGION

This setting is registered by VDD VEMM.SYS as an installable VDD. Use this setting to adjust the size of an additional Expanded Memory Specification (EMS) region.

This DOS setting allows you to define "another" region of memory to be used as the pseudobanks of memory traditionally found on the special hardware memory card. (See EMS in the *Glossary of Terms* section of this chapter for further explanation of what this entails.)

EMS_LOW OS_MAP_REGION

This setting is registered by VDD VEMM.SYS as an installable VDD. Use this setting if your program can use remappable, conventional memory, and you want to set the size of the remappable, conventional memory available to a DOS session.

This DOS setting allows you to define the size of the real EMS memory available to the processor. (See EMS in the *Glossary of Terms* section of this chapter for further explanation of what this entails.)

EMS_MEMORY_LIMIT

This setting is registered by VDD VEMM.SYS as an installable VDD. Use this setting to adjust the amount of Expanded Memory Specification (EMS) memory, in kilobytes, available to a DOS session.

This DOS setting allows you to define the "normal" region of memory to be used as the pseudobanks of memory traditionally found on the special hardware memory card. (See EMS in the *Glossary of Terms* section of this chapter for further explanation of what this entails.)

MEM_EXCLUDE_REGIONS

This setting is registered by VDD VBIOS.SYS as a default VDD. Use this setting to specify regions that Expanded Memory Specification (EMS) and Extended Memory Specification (XMS) cannot use because they are needed for device drivers or other uses.

Some DOS applications and adapters specify what areas in the RMSIZE to 1MB region they are reserving for their own use. Specifying regions reduces the chances of dynamic memory contamination and device driver failure.

MEM_INCLUDE_REGIONS

This setting is registered by VDD VBIOS.SYS as a default VDD. Use this setting to specify regions that Expanded Memory Specification (EMS) and Extended Memory Specification (XMS) can use between RMSIZE and 1MB.

Specifying regions for EMS and XMS helps ensure availability of adequate UMB space.

XMS_HANDLES

This setting is registered by VDD VXMS.SYS as an installable VDD. Use this setting to adjust the number of handles needed to identify blocks of Extended Memory Specification (XMS).

Reducing the number of preallocated handles frees memory space for other uses. Check to see whether your DOS application uses XMS. If not, then set the number of handles to 0.

XMS_MEMORY_LIMIT

This setting is registered by VDD VXMS.SYS as an installable VDD. Use this setting to adjust the amount of Extended Memory Specification (XMS) memory, in kilobytes, available to a DOS session.

Check to see whether your DOS application uses XMS. If not, then set the number of kilobytes of memory to 0. Decreasing the value to 0 frees room needed for Windows applications running in WIN-OS/2 in this VDM.

XMS_MINIMUM_HMA

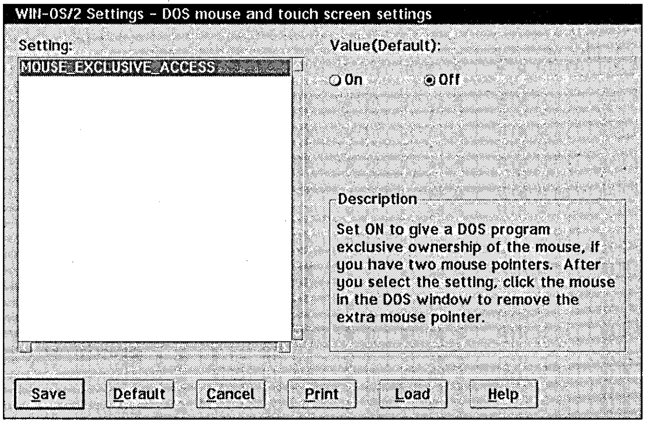
This setting is registered by VDD VXMS.SYS as an installable VDD. Use this setting to adjust the minimum allocation size of the High Memory Area (HMA).

Reducing the minimum allocation size can help you take advantage of what might have been dead memory chunks in the HMA.

DOS mouse and touch screen settings

The dialog box shown in Figure 16-7 enables you to modify the mouse and touch screen settings for a DOS or WIN-OS/2 session.

Figure 16-7.
WIN-OS/2 Settings — DOS Mouse and Touch Screen Settings


WIN-OS/2 Settings - DOS mouse and touch screen settings

Setting:
MOUSE_EXCLUSIVE_ACCESS

Value(Default):
○ On ● Off

Description
Set ON to give a DOS program exclusive ownership of the mouse, if you have two mouse pointers. After you select the setting, click the mouse in the DOS window to remove the extra mouse pointer.

MOUSE_EXCLUSIVE_ACCESS

This setting is registered by VDD VMOUSE.SYS as an installable VDD. If you are executing a DOS application and notice two mouse pointers on the screen, that DOS application requires exclusive ownership of the mouse. After you select the setting, click the mouse in the DOS window to remove the extra mouse pointer.

Tip

Some DOS applications provide their own mouse supports, which will conflict with the mouse support automatically provided by the OS/2 operating system. Enabling the DOS application to manage the mouse enables only one visible mouse pointer for use.

TOUCH_EXCLUSIVE_ACCESS

This setting is registered by VDD VTOUCH.SYS as an installable VDD. This setting gives the windowed DOS program exclusive ownership of the touch-screen display. When this setting is selected and active, only the DOS application receives touch display data, not Presentation Manager (PM). Mouse emulation in PM is deactivated.

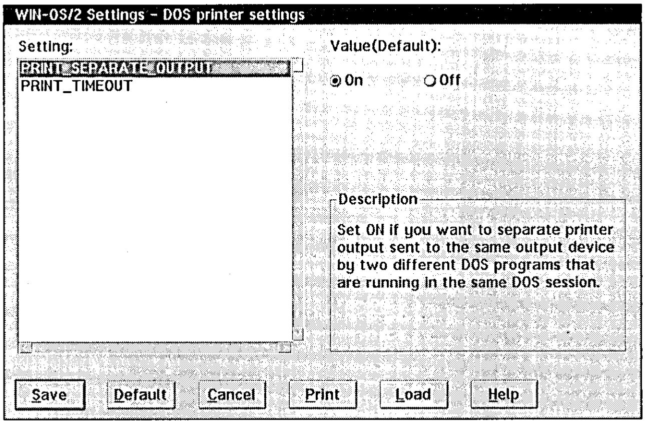
Tip

This setting is very specific to the use of touch displays and should be changed from the system default only if you are using a touch display. The windowed DOS session uses PM to provide a window on the desktop. The window and mouse usage for this windowed DOS session is managed by PM. The mouse activity and video services are filtered back and forth between PM and the DOSKRNL in a windowed DOS session.

The touch display replaces the mouse functions. The touch display actions need to be filtered and sent through PM to the DOSKRNL as if you had a mouse attached. This DOS setting set to ON eliminates any confusion PM may have about which session should be passed the touch-screen information.

DOS printer settings

The dialog box shown in Figure 16-8 enables you to modify the printer settings for a DOS or WIN-OS/2 session.



PRINT_SEPARATE_OUTPUT
PRINT_TIMEOUT

◉ On ○ Off

Set ON if you want to separate printer output sent to the same output device by two different DOS programs that are running in the same DOS session.

Figure 16-8. WIN-OS/2 Settings — DOS Printer Settings

PRINT_SEPARATE_OUTPUT

This setting is registered by VDD VLPT.SYS as a default VDD.

Set ON if you want to separate printer output sent to the same output device by two different DOS programs that are running in the same DOS session. Using this setting in the OFF state will combine output from two or more DOS programs running in the same DOS VDM session. The ON state separates the data when it reaches the OS/2 spooler. Most of the time you will want to have the output separated. Under most conditions you think about the output as being separate reports. As an option this has been added for OS/2 Warp. The ON state was hardcoded in the VLPT VDD in previous releases of OS/2 2.x.

PRINT_TIMEOUT

This setting is registered by VDD VLPT.SYS as a default VDD. Use this setting to adjust the amount of time, in seconds, that the operating system waits before forcing a print job to the printer.

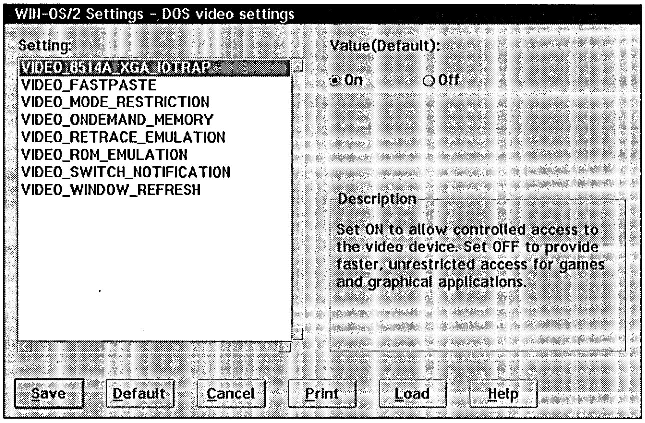
Tip

Using this setting can help DOS applications that assume DOS spooling is in use. Most DOS applications expect when writing to the print device that printing will start when the print buffer is full. The OS/2 operating system looks for an end-of-file character as the indicator to close the spool file and make it available to be printed. Since most DOS applications do not close the printer, which causes an end-of-file character to be generated in the print data stream, their spool file could remain open indefinitely. This PRINT_TIMEOUT value ensures that, if no end-of-file character is seen in the print data stream before the time-out value, then the print data will automatically be sent to the printer upon expiration of the time-out.

DOS video settings

The dialog box shown in Figure 16-9 enables you to modify the video settings for a DOS or WIN-OS/2 session.

Figure 16-9. WIN-OS/2 Settings — DOS Video Settings



WIN-OS/2 Settings - DOS video settings

Setting:

VIDEO_8514A_XGA_IOTRAP
VIDEO_FASTPASTE
VIDEO_MODE_RESTRICTION
VIDEO_ONDEMAND_MEMORY
VIDEO_RETRACE_EMULATION
VIDEO_ROM_EMULATION
VIDEO_SWITCH_NOTIFICATION
VIDEO_WINDOW_REFRESH

Value(Default):

● On ○ Off

Description

Set ON to allow controlled access to the video device. Set OFF to provide faster, unrestricted access for games and graphical applications.

VIDEO_FASTPASTE

This setting is registered by VDD VVIDEO.SYS as a default VDD. Use this setting to speed up input from sources other than the keyboard, such as text pasted from the clipboard.

Tip

Using this setting is helpful to ensure that your clipboard operations get a higher boost of processor time to transfer the data quickly. Applications using clipboard functions without enabling this DOS setting may experience delayed transfer of data between Presentation Manager (PM) to WIN-OS/2 or WIN-OS/2 to WIN-OS/2.

VIDEO_MODE_RESTRICTION

This setting is registered by VDD VVIDEO.SYS as a default VDD. Use this setting to extend DOS conventional memory beyond 640KB by limiting the video mode support to text or CGA graphics.

Tip

This DOS setting is useful if you always plan to use this DOS session for some application that does not need any better graphics than those supported with CGA. An example would be a text editor such as PE2.EXE.

VIDEO_ONDEMAND_MEMORY

This setting is registered by VDD VVIDEO.SYS as a default VDD. Use this setting to delay allocation of a video-save buffer, which can free memory swap space for use by a full-screen DOS session.

Tip

This setting may slow initial swapping of video memory, but it provides the fine tuning needed to give more space to your full-screen DOS session.

VIDEO_RETRACE_EMULATION

This setting is registered by VDD VVIDEO.SYS as a default VDD. Use this setting to disable simulated video retrace, forcing retrace to occur only in the intervals requested by the DOS program.

Tip

This DOS setting can free needed processor time that is being consumed to perform the simulated video retrace when your DOS program does not really need it.

VIDEO_ROM_EMULATION

This setting is registered by VDD VVIDEO.SYS as a default VDD. Use this setting to disable emulation of commonly used video ROM functions, such as WriteChar, WriteTTY, and full-screen scroll, if your video ROM adds enhancements to these functions.

Tip

This DOS setting may help prevent conflicts that can occur when off-planar VGA graphics adapters are used together with base video ROM functions typically provided with planar-based graphics systems.

VIDEO_SWITCH_NOTIFICATION

This setting is registered by VDD VVIDEO.SYS as a default VDD. Use this setting to notify the DOS program when the session switches to or from full screen.

Tip

Some DOS applications with certain video adapters suspend writing to the screen when the program goes into the background. Notifying the DOS application of this change can help prevent blackout areas from appearing on the screen.

VIDEO_WINDOW_REFRESH

This setting is registered by VDD VVIDEO.SYS as a default VDD. Use this setting to adjust the window update frequency, in tenths of seconds, for a specific DOS session.

Tip

Setting this value greater than the default allows for increased system performance by not tying up the video subsystem in refreshing this screen as often. A batch program that does not update the screen frequently can benefit from setting this value higher than the default.

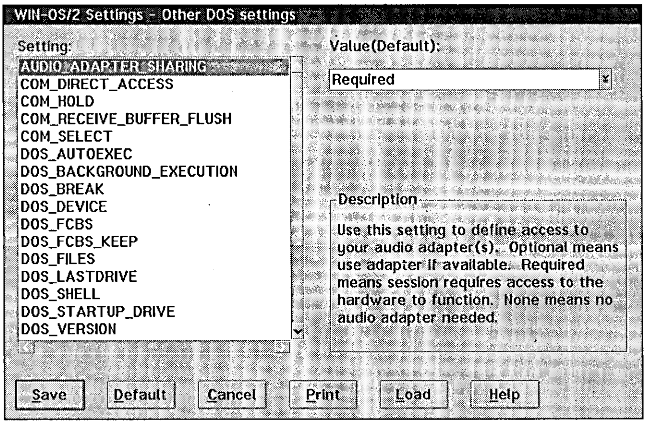
VIDEO_8514A_XGA_IOTRAP

This setting is registered by VDD VXGA.SYS as an installable VDD. Use this setting to provide faster, unrestricted access to 8514/A hardware.

If you have an 8514 display and are having trouble using Dynamic Data Exchange (DDE) with WIN-OS/2, set this DOS setting to OFF. Setting this DOS setting to ON slows down the 8514 video-painting process. If there is any other I/O-intensive process running, such as DDE, there will be contention. The result is that you will see very slow painting to your screen.

Other DOS settings

The dialog box shown in Figure 16-10 enables you to modify the other settings that do not fit a unique category for a DOS or WIN-OS/2 session.



WIN-OS/2 Settings - Other DOS settings

Setting:

AUDIO_ADAPTER_SHARING
COM_DIRECT_ACCESS
COM_HOLD
COM_RECEIVE_BUFFER_FLUSH
COM_SELECT
DOS_AUTOEXEC
DOS_BACKGROUND_EXECUTION
DOS_BREAK
DOS_DEVICE
DOS_FCBS
DOS_FCBS_KEEP
DOS_FILES
DOS_LASTDRIVE
DOS_SHELL
DOS_STARTUP_DRIVE
DOS_VERSION

Value(Default): Required

Description

Use this setting to define access to your audio adapter(s). Optional means use adapter if available. Required means session requires access to the hardware to function. None means no audio adapter needed.

Figure 16-10. WN-OS/2 Settings — Other DOS Settings

AUDIO_ADAPTER_SHARING

This setting is registered by VDD AUDIO.SYS as an installable VDD. Use this setting to define access to your audio adapter(s):

- ✗ **Optional** means use adapter if available.
- ✗ **Required** means the session requires access to the hardware to function.
- ✗ **None** means no audio adapter is needed.

Use this setting to give the program that is running the ability to access the sound card if needed. This will effectively give ownership to the DOS or WIN-OS/2 program the first time it touches the sound card. Once it gives ownership to the VDM, then the MMPM/2 device drivers will cooperate. MMPM/2 will not attempt to gain access to the sound card until the DOS or WIN-OS/2 VDM terminates.

COM_DIRECT_ACCESS

This setting is registered by VDD VCOM.SYS as an installable VDD. Use this setting to give the program running in this session direct access to COM ports.

Using this setting will help programs that think they are running a version of DOS to feel right at home. Because the OS/2 operating system provides a solid base level of device driver support, there is good reason to use this support instead of a vendor-written device driver for each application. For those programs that require control of the COM port hardware to execute, this DOS setting allows them to have it without the OS/2 operating system device support playing the final arbitrator.

COM_HOLD

This setting is registered by VDD VCOM.SYS as an installable VDD. Use this setting to keep open a communications resource (for example, COM1) until the DOS session ends.

Since the OS/2 operating system provides a solid base-level of device driver support, there are built-in, "time-out if no activity" checks that will cause the COM port to shut down cleanly. For those applications that assume ownership of the COM port and assume they are the only one watching the COM port, this is the setting to enable. Since there are standard ANSI asynchronous, control-line protocols that define what is going on at all times, these controls have been built into the base COM port device driver support.

Tip

This DOS setting tells the base COM port device driver to ignore any error/shutdown conditions. By using the COM HOLD setting, you are allowing the application to make all of those decisions. This setting should keep applications from prematurely crashing when the base COM device driver shuts down.

For example, when you creating a secondary shell or child DOS session, the terminal emulator application Procomm Plus (PCPLUS.EXE) can lose its hold on the COM port. This DOS setting will keep the COM port open for the parent DOS session and the child session.

COM_RECEIVE_BUFFER_FLUSH

This setting is registered by VDD VCOM.SYS as an installable VDD. Use this setting to control the content of the received data buffers when the DOS session is switched to the foreground or when the DOS program enables the received data interrupt. This setting also controls the flushing of data for the VCOM device driver:

✗ If you select the value of **Switch to Foreground**, data will be discarded when the session is switched to the foreground.

✗ If you select the value of **Receive Data Interrupt Enable,** data will be discarded every time the received data interrupt is raised.

✗ If you select the value of **All**, then the receive buffers will be flushed both in the foreground and when the VCOM device driver raises the receive data interrupt.

✗ The default value of **None** will allow all received data to be saved in the buffers.

COM_SELECT

This setting is registered by VDD VCOM.SYS as an installable VDD. Use this setting to limit access to the selected COM port (for example, COM1) only until the DOS session ends.

Tip

Using this setting will guarantee that any other VDM or protect-mode application cannot dynamically get this COM port during your session's execution. This is a good setting if you are not sure what other communication applications are running in the system. During initialization, sometimes applications (terminal emulators) will acquire access to devices they think they might need despite your installed setup. If this happens while you are expecting the COM port to be owned by your current VDM, your application may terminate. This DOS setting will protect you from losing the COM port dynamically.

For example, when you attempt to run more than one version of terminal emulator application Procomm Plus (PCPLUS.EXE), the COM ports may be dynamically disabled by the other copy of PCPLUS that is attempting to initialize. COM_SELECT will avoid this problem.

DOS_AUTOEXEC

This setting is registered by DOSKRNL. Use this setting to allow an alternate batch file to execute during startup of a DOS session. The default is \AUTOEXEC.BAT on the boot drive.

DOS_AUTOEXEC allows you to set up a unique batch file to start up each DOS session. It also permits you to pass parameters to your AUTOEXEC.BAT file. Leaving the name of the file blank will cause no batch file to be executed, including AUTOEXEC.BAT. The following example shows a sample batch file:

```
C:\session1.BAT CONFIG.SYS

session1.BAT

    @PE2.EXE %1

    @EXIT
```

DOS_BACKGROUND_EXECUTION

This setting is registered by VDD VTIMER.SYS as a default VDD. Use this setting to allow DOS programs to run in the background. When set to OFF, the program will stop when it is not in the foreground. This setting should be set to ON for communication programs and WIN-OS/2 Dynamic Data Exchange (DDE).

Since the OS/2 operating system uses preemptive multitasking and multithreading simultaneously, the system must handle passing out time slices to all tasks in the system. For good reasons, all tasks run at the same priority. The VDMs, in which DOS emulation runs, are considered by tasking to be just another ring 3 application. All user applications run in ring 3.

Warning

It is possible to starve a ring 3 application if the system becomes tied up too often in servicing another process in the system running at a higher priority. If the VDM becomes starved, it will appear to be hung. If the VDM has a DOS application doing communication using the base device driver support, time-outs will occur. When time-outs occur, the device driver shuts down the communication port. If this happens, your communication program may terminate unexpectedly. A setting of ON increases the chance your application will not lose its needed time slices when switching sessions. Switching away from the VDM as your foreground task will relinquish priority over other tasks running at the same level. Voluntarily lowering your priority can be detrimental to an application that expects to be in total control of the system, such as an emulator application.

For example, when you are running the terminal emulator application Procomm Plus (PCPLUS.EXE) in the background, the COM port can become starved for time slices. You can be logged off the host because of lack of activity just from the background execution scenario. The DOS setting DOS_BACKGROUND_EXECUTION = ON will provide additional time slices to the application running as a terminal emulator.

DOS_BREAK

This setting is registered by VDD VBIOS.SYS as a default VDD. Use this setting to activate Ctrl-C (or Ctrl-Break) as a break character in your DOS session.

DOS_BREAK is the same as the real DOS command BREAK=ON|OFF in CONFIG.SYS. When DOS_BREAK=ON, the Ctrl-C or Ctrl-Break are looked at during every INT21 call to the file system. This setting can help if you have a program that consistently seems to ignore the keyboard when you want to stop it from running. When DOS_BREAK=OFF, the Ctrl-C or Ctrl-Break are looked at only during I/O INT21 calls to the file system.

DOS_DEVICE

This setting is registered by VDD VBIOS.SYS as a default VDD. Use this setting to load a DOS device driver only into the selected DOS session. Alternatively, specify DEVICE= in CONFIG.SYS if you want the driver loaded into all DOS sessions.

Only character device drivers, and not block device drivers, are supported in Multiple Virtual DOS Machines (MVDM). An example of a block device driver is one for controlling a fixed disk. The DOS_DEVICE DOS setting can be used to enable only the device drivers needed for any single DOS session.

Following are examples of when you might use this setting:

- ✘ You are using a network in one DOS session and not in another.
- ✘ The network device drivers are needed just for that single session.

✗ The network device drivers are not in the CONFIG.SYS file.

✗ The network device drivers are only included in this DOS_DEVICE list.

Including the network device drivers in more than one VDM DOS_DEVICE list or having them in the CONFIG.SYS and another DOS_DEVICE list will cause a conflict. When a VDM startup device driver conflict occurs, the VDM may be killed by MVDM before the DOS session is started.

It is best to include in the DOS_DEVICE list only the device drivers that are needed for that DOS session.

For example, the command DXMA0MOD.SYS could be given in an MLE text editor dialog box.

DOS_FCBS

This setting is registered by VDD VBIOS.SYS as a default VDD. Use this setting to specify the maximum number of File Control Blocks (FCB) that can be open concurrently in one DOS session. This setting affects only file-sharing modules.

Most users will not need to concern themselves with this DOS setting since most applications today do not use FCBs. FCBs relate primarily to applications written to run on a DOS version before DOS 2.0. Most applications today were written to run on DOS 2.0 or higher and use SFT-file-handle-based I/O.

This setting makes MVDM allocate a System File Table entry (SFT) per FCB during MVDM initialization. These SFT/FCBs are set up to be managed by DOSKRNL and the file system. DOSKRNL also sets up the necessary information for each SFT/FCB in the DOSKRNL read-only, lists-of-lists structure that many applications count on for finding the current file system status.

DOS_FCBS_KEEP

This setting is registered by VDD VBIOS.SYS as a default VDD. Using this setting reserves a number of File Control Blocks (FCBs) that the operating system cannot close to make room for opening another file.

Most users will not need to concern themselves with this DOS setting since most applications today do not use FCBs. The use of FCBs primarily relates to applications written to run on a DOS version before DOS 2.0. Most applications today were written to run on DOS 2.0 or higher and use SFT-file-handle-based I/O.

This setting is checked against the DOS_FCBS DOS setting since the number of FCBs to reserve should not be greater than the total number allocated in the DOS_FCBS DOS setting. DOS_FCBS_KEEP is saved by an MVDM as a control variable for the DOSKRNL in managing FCBs and deciding how many FCB/SFTs to close.

DOS_FILES

This setting is registered by VDD VBIOS.SYS as a default VDD. Use this setting to specify the maximum number of files that can be opened in a DOS session.

Although this value sets up the maximum number of file handles, it is still up to the program to grow the file-handle table. This setting overrides the FILES= in CONFIG.SYS. You are given a minimum of 20 open file handles despite the value in the DOS_FILES setting and FILES= in CONFIG.SYS.

Note

Changing this DOS_FILES value does not guarantee that your DOS application will take advantage of growing the file-handle table beyond the minimum. An example is dBASE IV. dBASE IV looks at the DOS_VERSION value and decides if it understands the version. If the version is less than 10.0, then dBASE attempts to grow the file-handle table. If the version number is greater than or equal to 10.0, then dBASE assumes this version DOS does not support growing the file-handle table. It then executes attempting to make do with the minimum number of 20 default handles available.

DOS_LASTDRIVE

This setting is registered by VDD VBIOS.SYS as a default VDD. Use this setting to identify the highest disk-drive letter available to the DOS session.

Tip

Some DOS applications, such as LANs, expect to reserve for private use disk drive letters following the highest disk drive letter. Always having the highest letter of Z may cause these types of applications to fail.

DOS_SHELL

This setting is registered by VDD VBIOS.SYS as a default VDD. Use this setting to specify a path and file name for the DOS command processor you want to use in the selected session.

In each DOS session, it is possible to have a unique command processor other than the default COMMAND.COM that is standard with OS/2.

DOS_STARTUP_DRIVE

This setting is registered by VDD VBIOS.SYS as a default VDD. Use this setting to specify the location of the DOS kernel to boot.

The location of your VMBOOT disk may not be drive A:. This DOS setting allows you to select drive A:, drive C:, or a path to a VMDISK-created boot-drive image. Using this setting allows you to have multiple hard disk boot-drive images for operating systems that conform to the Intel architecture and use the INT21 file system calling conventions. DOS 5.0 is an example of this kind of operating system.

DOS_VERSION

This setting is registered by DOSKRNL. Use this setting to select program versions compatible with the DOS version used in this DOS session.

The particular DOS version selected will cause programs to take different paths of execution. Since each DOS evolved and was a superset of the previous version, new methods for accessing the system can be used. The DOS lists-of-lists changed the meaning of its fields and structures; therefore, the programs had to understand and change as needed to run on each new version of DOS. The OS/2 operating system as a default tries to be as compatible with DOS 5.0 as possible. Changing the DOS_VERSION field does not make the OS/2 operating system compatible with that version of DOS. This field only informs the program being executed of the selected version of DOS it is running on. An example of an application that changes its execution path because of the version is dBASE IV.

For example, you could set DOS_VERSION = dBASE.exe,5,00,255, where the first parameter is the executable name, the second parameter is the 5 of DOS 5.00, the third parameter is the 00 of DOS 5.00, and the fourth parameter is the number of times this version will be replied when this application asks what DOS version is running. The number 255 means to answer indefinitely the previously noted DOS version number.

HW_NOSOUND

This setting is registered by VDD VTIMER.SYS as a default VDD. Use this setting to prevent the session from generating program-defined sounds.

Shhh! Tip

This setting can be used to deactivate the program sounds when this run-time option is not available. For example, you might use this setting to deactivate the squeaking keystroke feedback sound when using Procomm Plus.

HW_ROM_TO_RAM

This setting is registered by VDD VBIOS.SYS as a default VDD. Use this setting to copy ROM BIOS code to RAM, for example, if you want to set breakpoints while debugging a ROM program, or if you want to speed up processing.

Shhh! Tip

This setting is useful if the program has to dynamically modify ROM BIOS. This can also be helpful if you are debugging a ROM and want to set breakpoints in the debugger in this area of memory. Most debuggers typically replace any debug point in RAM with some predefined hardware interrupt call, for example INT3. To debug a ROM with breakpoints, you must copy the ROM to RAM so that the memory can be modified.

HW_TIMER

This setting is registered by VDD VTIMER.SYS as a default VDD. Use this setting to give the program running in this session direct access to hardware timer ports.

Shhh! Tip

This DOS setting allows the running DOS application to implement its own timer services instead of using the base operating system primitives to control the timer.

IDLE_SECONDS

This setting is registered by VDD VKBD.SYS as a default VDD. Use this setting to select a period of allowable idle time before the operating system reduces the idle program's portion of processor time.

Shhh! Tip

This DOS setting enables a VDM not to yield its time slice to the scheduler. This may be important if you have a DOS application that is waiting on a slow device to do something that should not be interrupted.

IDLE_SENSITIVITY

This setting is registered by VDD VKBD.SYS as a default VDD. Use this setting to set a threshold for polling time before the operating system reduces the polling program's portion of processor time. Setting this value to 100 will disable the idle detection.

IDLE_SENSITIVITY provides a way to make your DOS applications run faster. If you plan to run multiple VDM sessions, it is best to decide from experimentation your best mix of IDLE_SENSITIVITY settings among the VDMs. All VDM sessions cannot actually run at the maximum value so it is best to decide which application would take best advantage of increased IDLE_SENSITIVITY and which could operate with less IDLE_SENSITIVITY. Any application that is polling the keyboard will need a higher IDLE_SENSITIVITY than ones that mostly crunch numbers in the background.

Another scenario is a hung DOS application that begins to speed up when you press any key redundant. Most likely, this DOS application is hooking the IDLE_SENSITIVITY Interrupt Request Queue (IRQ). The IDLE_SENSITIVITY code in the VDD will become confused, and the application will appear to be hung. Pressing any key redundant returns the IDLE_SENSITIVITY code to synchronize, and the application resumes its normal processing speed. Increasing the IDLE_SENSITIVITY for the session will overcome this problem, and your application will run smoothly.

INT_DURING_IO

This setting is registered by VDD VPIC.SYS as a default VDD. Use this setting to enable interrupts during file reads and writes. When set to OFF, DOS programs will not receive interrupts until after read/write is done. This setting should be set to ON for multimedia applications.

Multimedia applications expect a high rate of interrupt processing when performing full-motion video or high-fidelity sound reproduction. By allowing for this overlapped interrupt I/O processing, a more contiguous, less jerky video and sound reproduction can be accomplished.

SESSION_PRIORITY

This setting is registered by VDD VDPMI.SYS as a default VDD. Use this setting to change the priority level from 1 (lowest priority) to 32 (highest priority) for a specific DOS or WIN-OS/2 session.

Using this setting is helpful to boost the priority of the session higher than other VDMs or protect mode programs in the system. Changing this value to greater than the default of 1 will affect the print spooling in the system.

Warning

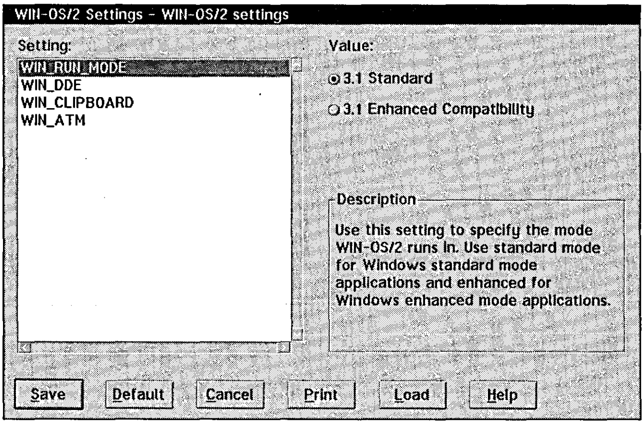
This setting should only be changed if there appears to be a problem using your sound card or other real-time process in this DOS or WIN-OS/2 session. The session priority is set at the beginning of the DOS or WIN-OS/2 session initialization. If there is some other process in the system that causes the priority to decrease, you will not always be running under the increased priority selected in this setting. If the session priority is set too high and you switch back to the desktop, there may be some difficulty getting focus to the desktop.

Tip

This lack of focus is just one of the side effects of this setting. It is advisable not to deviate the setting much from the default without testing the side effects it has on the rest of the system.

WIN-OS/2 settings

The dialog box shown in Figure 16-11 enables you to modify just the WIN-OS/2 settings for a WIN-OS/2 session.



Setting:

- WIN_RUN_MODE
- WIN_DDE
- WIN_CLIPBOARD
- WIN_ATM

Value:

● 3.1 Standard

○ 3.1 Enhanced Compatibility

Description

Use this setting to specify the mode WIN-OS/2 runs in. Use standard mode for Windows standard mode applications and enhanced for Windows enhanced mode applications.

Figure 16-11. The WIN-OS/2 Settings

WIN_RUN_MODE

This setting is registered by VDD VWIN.SYS as an installable VDD. Use this setting to specify the mode WIN-OS/2 runs in. Use standard mode for Windows standard mode applications and enhanced mode for Windows enhanced mode applications.

Tip

This relieves you from having to remember the correct syntax of the command line switches for starting WIN-OS/2 in the correct mode for your applications.

WIN_DDE

This setting is registered by VDD VWIN.SYS as an installable VDD. Use this setting to allow WIN-OS/2 to share DDE information between public WIN-OS/2 and OS/2 sessions.

This DOS setting should be in the OFF position when the DDE function is not required for your DOS or Windows application. When this setting is in the OFF position, portions of the VDM DDE resources are freed. The memory normally used for these resources becomes available for other functions in the VDM session.

WIN_CLIPBOARD

This setting is registered by VDD VWIN.SYS as an installable VDD. Use this setting to allow WIN-OS/2 to share clipboard information between public WIN-OS/2 and OS/2 sessions.

This DOS setting should be in the OFF position when the clipboard function is not required for your DOS or Windows application. When this setting is in the OFF position, portions of the VDM clipboard resources are freed. The memory normally used for these resources becomes available for other functions in this VDM session.

WIN_ATM

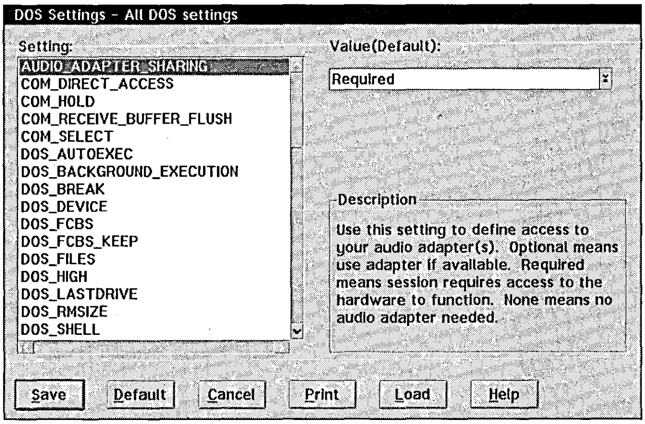
This setting is registered by VDD VWIN.SYS as an installable VDD. Use this setting to tell WIN-OS/2 to load and use Adobe Type Manager (ATM) fonts for WIN-OS/2 sessions.

This WIN-OS/2 setting should be in the OFF position when you do not need ATM fonts. These fonts take up considerable memory and resources inside of WIN-OS/2, and many Windows programs do not need these fonts to run. However, some programs such as Passport Designs Encore MIDI sequencing software require these fonts or else the program will not complete installation. The Passport Designs Encore program will also need a postscript printer setup using ATM SONATA fonts before it will run. When this setting is in the OFF position, the memory normally used for these resources becomes available for other functions in this VDM session.

All DOS settings

The dialog box shown in Figure 16-12 enables you to modify all of the DOS settings for a DOS session.

Figure 16-12. DOS Settings — All DOS Settings


DOS Settings - All DOS settings

Setting:

AUDIO_ADAPTER_SHARING
COM_DIRECT_ACCESS
COM_HOLD
COM_RECEIVE_BUFFER_FLUSH
COM_SELECT
DOS_AUTOEXEC
DOS_BACKGROUND_EXECUTION
DOS_BREAK
DOS_DEVICE
DOS_FCBS
DOS_FCBS_KEEP
DOS_FILES
DOS_HIGH
DOS_LASTDRIVE
DOS_RMSIZE
DOS_SHELL

Value(Default): Required

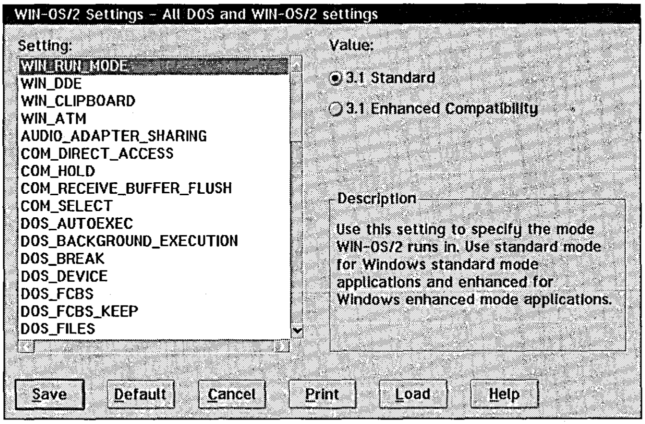
Description

Use this setting to define access to your audio adapter(s). Optional means use adapter if available. Required means session requires access to the hardware to function. None means no audio adapter needed.

All DOS And WIN-OS/2 settings

The dialog box shown in Figure 16-13 enables you to modify all of the DOS and WIN-OS/2 settings for a WIN-OS/2 session.

Figure 16-13. WIN-OS/2 Settings — All DOS and WIN-OS/2 Settings


WIN-OS/2 Settings - All DOS and WIN-OS/2 settings

Setting:

WIN_RUN_MODE
WIN_DDE
WIN_CLIPBOARD
WIN_ATM
AUDIO_ADAPTER_SHARING
COM_DIRECT_ACCESS
COM_HOLD
COM_RECEIVE_BUFFER_FLUSH
COM_SELECT
DOS_AUTOEXEC
DOS_BACKGROUND_EXECUTION
DOS_BREAK
DOS_DEVICE
DOS_FCBS
DOS_FCBS_KEEP
DOS_FILES

Value:

● 3.1 Standard
○ 3.1 Enhanced Compatibility

Description

Use this setting to specify the mode WIN-OS/2 runs in. Use standard mode for Windows standard mode applications and enhanced for Windows enhanced mode applications.

Glossary of Terms

- ✘ Multiple Virtual DOS Machines (MVDM) — MVDM manages VDMs and their arbitration service needed for virtualization of physical hardware and access to the protect-mode kernel and file system(s).
- ✘ Virtual DOS Machines (VDM) — Each VDM is thought to be a separate DOS session. These DOS sessions each have their own DOSKRNL to manage DOS functions such as file system calls.

- ✘ DOSKRNL — The kernel that allows DOS programs to run in the OS/2 Warp operating system version 3.0. The DOSKRNL uses the file system managed by OS2KRNL in protect-mode.
- ✘ OS2KRNL — The protect-mode kernel of the OS/2 operating system that allows applications to use the file system and communicate with physical devices.
- ✘ **DOS Protected Mode Interface (DPMI)** — The interface that allows DOS programs to use OS2KRNL services that run in protect-mode. DPMI is mainly used with WIN-OS/2 applications, but some DOS applications also use it.
- ✘ **Virtual Device Drivers (VDD)** — Device drivers that make the DOS programs think they are talking directly to the physical device drivers.
- ✘ **Physical Device Drivers (PDD)** — Device drivers that manage the physical hardware.
- ✘ **Lists-of-lists** — The area in the DOSKRNL that has the information related to the file system. Many DOS applications read and write to this information area. The OS/2 operating system DOSKRNL only supports reading from this area.
- ✘ **Expanded Memory Specification (EMS)** — EMS traditionally is a hardware and software combination for managing extra memory found on a special memory card. The memory on this card would be set up into banks of memory in a region. These banks could then be manipulated under calls to EMS software. The real EMS memory available to the processor was found below the 1MB line. The EMS software could be directed to swap those hardware banks of memory with the memory found below the 1MB line.
- ✘ **Extended Memory Specification (XMS)** — XMS is a memory specification for addressing memory above the 1MB line.
- ✘ **Upper Memory Blocks (UMB)** — This is memory between the 640KB and 1MB lines that DOS can use as conventional memory.
- ✘ **High Memory Area (HMA)** — This is the first 640KB of XMS memory above the 1MB line. Real mode programs can access this memory as if it were conventional memory.

Printing and Loading DOS or WIN-OS/2 Settings

OS/2 Warp V 3.0 has a new feature that lets you print out the DOS or WIN-OS/2 settings to a printer or to a file. If printed to a file, the settings can be loaded back in for any DOS or WIN-OS/2 object on your system. Since the printed file is in ASCII text, it can be sent to users in the field. The users in the field can quickly send a copy of their DOS or WIN-OS/2 settings into service when needed. We created this feature because many users in the field asked for it.

Reliability of user changes via a text editor

A DOS or WIN-OS/2 settings file can be reduced by any ASCII text editor to just the DOS settings necessary for the session. The settings, once loaded and saved, will be checked by the associated VDDs for accuracy. As a result, it is very unlikely that the setting dialog stating that the system has a problem with this setting will ever appear when attempting to run this session. Another way of propagating the settings per application would be to use the MIGRATE.EXE DATABASE.TXT. This database is an ASCII text file, but there is no validation of the data by the VDM prior to the starting of the session. As a result, the settings dialog stating the system has a problem with the settings often appears when attempting to run this object. By using the LOAD and SAVE of the DOS or WIN-OS/2 settings, this is very unlikely to occur. Still another way of propagating the settings per application would be to use a REXX command file. This too has the same chance for error as the MIGRATE.EXE DATABASE.TXT.

Why making changes via a text editor is important

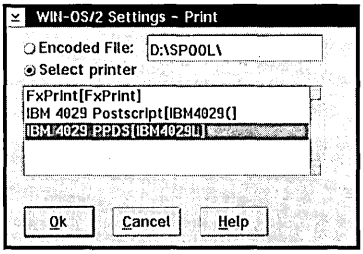
Being able to make user changes via a text editor is important to a power user because if you have found a way under your environment to run a particular DOS or WIN-OS/2 application, you will be able to easily propagate these settings to others.

Also, if you need technical support, you will be able to print out the settings. You can then send the printout to service, thus cutting down the debug time needed for service to analyze whether the problem is just a DOS or WIN-OS/2 settings problem.

Printing DOS or WIN-OS/2 settings

You can print the DOS and WIN-OS/2 settings by clicking on the **Print** push button to display (see paragraph following figure) one of the DOS or WIN-OS/2 settings dialog boxes (see Figure 16-14).

Figure 16-14. Print Settings Dialog



WIN-OS/2 Settings - Print

○ Encoded File: D:\SPOOL\

● Select printer

FxPrint[FxPrint]
IBM 4029 Postscript[IBM4029(]
IBM 4029 PPDS[IBM4029U]

The **Print** push button causes the Print dialog box to be displayed. You have two printing options:

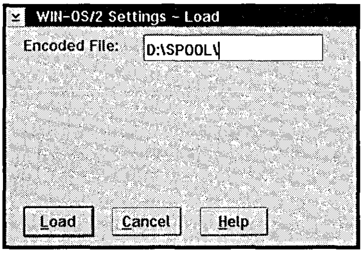
- You can print to file. If you select the **Encoded File:** radio button, then you must enter a file name in the entry field provided. If the full path is not provided, then the spooler path from the system is used to save the file.
- You can print to a printer queue. If you select the **Select printer** radio button, then you must choose a printer queue from the list of printers defined to the spooler.

In either case, you must click on the **Print** push button to print.

Loading DOS or WIN-OS/2 settings

You can reload a previously saved set of DOS or WIN-OS/2 settings by clicking on the **Load** push button on one of the DOS or WIN-OS/2 settings dialog boxes (see Figure 16-15).

Figure 16-15. WIN-OS/2 Settings — Load Settings Dialog Box



WIN-OS/2 Settings - Load

Encoded File: D:\SPOOL\

You must enter the name of the file containing the saved DOS or WIN-OS/2 settings in the entry field provided. If the full path is not provided, then the spooler path from the system is used as the path for the saved file.

You must click on the **Load** button to load the DOS settings file into the DOS settings notebook properties. When the file is being processed, each DOS setting name is matched for registered settings before any change to a setting occurs. For a multiple line editor (MLE) DOS setting, such as DOS_DEVICE, the processing of the loaded settings data is done by tricking the MLE into thinking the data is being manually typed into the MLE. This trick is done by simulating the cursor placement over and off the MLE editing area. This movement triggers the MLE to change its previous contents to the new contents.

- ✗ Printing of the DOS settings enables you to print a copy of the pertinent information contained in the DOS settings. The Print dialog box uses a list box to display the local and remote printer and queue names. You can select to print to any printer on the list. Postscript printing compatibility has been enabled. You can also choose to save an encoded file version of the settings. This is a flat ASCII text file readable by any text editor.
- ✗ The loading of the DOS settings file into an single VDM allows the user to customize the DOS settings that will make the application best for the intended environment. This enables Original Equipment Manufacturers (OEMs) to include a DOS setting file to be used with their applications or VDD-PDD pair that sets up the VDM with minimal user intervention required. IBM technical support and the OEMs have been sending out printouts of DOS settings for customers to use, and they also have been requesting customers to send in hand-written copies or screen image printouts of their DOS settings. Besides being cumbersome and time-consuming, this form of transfer is prone to user error. Creation of an encoded file in item 1 and loading in item 2 will increase the usability of the DOS settings and reduce the overhead in providing service to the VDM.

Description Of The Encoded File Format

Parameter	Description
s = DCF	This is the signature string that must be at the beginning of the first line of the file.
i =	This is the icon title string for the file so that you can distinguish the DOS settings file between application programs.
p =	Property registered by the VDD, DOSKRNL, or WIN-OS/2.
t =	Type of property
	0 = boolean, matches VDMP_BOOL define
	1 = integer, matches VDMP_INT define
	2 = enumerated string, matches VDMP_ENUM define
	3 = string, matches VDMP_STRING define
	4 = multiple line editor (MLE), matches VDMP_MLSTR define
	5 = WIN-OS/2
v =	Value
	For type 0 (boolean):
	1 = True – Displayed as **On**
	0 = False – Displayed as **Off**
	For type 1 (integer): 0 – unsigned long
	For type 2 (enumerated string): string buffer from enumerated list of strings
	For type 3 (string): string buffer
	For type 4 (MLE): text buffer strings free-form
	For type 5 (WIN-OS/2):
	For property = WIN_RUN_MODE:
	320 = 3.1 Standard Mode
	321 = 3.1 Enhanced Compatibility

continued

Description Of The Encoded File Format, continued

Parameter	Description
	For property = WIN_DDE:
	1 = True – Displayed as **On**
	0 = False – Displayed as **Off**
	For property = WIN_CLIPBOARD:
	1 = True – Displayed as **On**
	0 = False – Displayed as **Off**
d =	Default indicator if setting saved is the default.
	1 = True – Displayed as **On**
	0 = False – Displayed as **Off**

Changing a setting and printing before saving may still display the default because the default flag is not set until the setting is saved.

Each DOS setting entry must be followed by a blank line as a separator. Each DOS setting entry must meet these conventions or else the loading process will reject the setting. You can create an encoded file with a text editor as long as the file format is used.

Sample DOS Settings File

```
s=DCF
i=WIN-OS/2 Full Screen - Settings
p=WIN_RUN_MODE
t=5
v=321  3.1 Enhanced Compatibility
d=1  On

p=COM_DIRECT_ACCESS
t=0
v=0  Off
d=1  On

p=COM_RECEIVE_BUFFER_FLUSH
t=2
v=NONE
d=1  On

p=DOS_AUTOEXEC
t=3
v=C:\AUTOEXEC.BAT
d=1  On
```

```
p=DOS_VERSION
t=4
v=DCJSS02.EXE,3,40,255
v=DFIA0MOD.SYS,3,40,255
v=DXMA0MOD.SYS,3,40,255
v=EXCEL.EXE,10,10,4
v=IBMCACHE.COM,3,40,255
v=IBMCACHE.SYS,3,40,255
v=ISAM.EXE,3,40,255
v=ISAM2.EXE,3,40,255
v=ISQL.EXE,3,40,255
v=MSD.EXE,5,00,255
v=NET3.COM,3,40,255
v=NETX.COM,4,00,255
v=NETX.EXE,5,00,255
v=PSCPG.COM,3,40,255
v=SAF.EXE,3,40,255
v=WIN200.BIN,10,10,4
d=1  On

p=DOS_FCBS
t=1
v=16
d=1  On
```

REXX Setup of a DOS or WIN-OS/2 Object

A DOS or WIN-OS/2 object in a folder can be created from a REXX command file executed from a CMD.EXE command line. The following is an example of a REXX command file to create a WIN-OS/2 Window object:

```
/* REXX */

/*—load the rexx utilities—*/
call RxFuncAdd SysLoadFuncs, RexxUtil, SysLoadFuncs
call SysLoadFuncs

/*—prepare data for the create object call—*/
classname='WPProgram'
title='HOST'
location='<WP_NOWHERE>'
option='REPLACE'   /* choices are: Fail, Replace, Update */
icon    ='ICONFILE=C:\DESKTOP\TEMPLATE\ICON.ICO;'
    program ='EXENAME=C:\WIN31\CLOCK.EXE;'
    type    ='PROGTYPE=WINDOWEDVDM;'
    startup ='STARTUPDIR=C:\WIN31;'
    set1    ='SET COM_DIRECTO_ACCESS=0;'
    set2    ='SET DOS_FILES=65;'
    set3    ='SET DOS_AUTOEXEC=C:\AUTOEXEC.BAT;'
    set4    ='SET COM_RECEIVE_BUFFER_FLUSH=NONE;'
    set5    ='SET DOS_VERSION=DBASE.EXE,5,00,255;'
```

```
open    ='OPEN=DEFAULT;'
setup=program||type||startup||open||icon||set1||set2||set3||set4||set5

result = SysCreateObject(classname, title, location, setup, option)
/* result=1 means WinCreateObject was successful, =0 means failure */
EXIT result
```

Explanation of Setup Parameters in REXX Setup Example

Parameter	What the parameter does
icon	This setup parameter specifies the full path of the ICON file to be used in the settings setup.
program	This setup parameter specifies the full path of the program to be used in the settings setup. This is the actual program that will be run when the object is selected.
type	This setup parameter specifies the appropriate session type for the object in the settings setup. In this example, it will setup the type as a DOS window versus a DOS full screen.
startup	This setup parameter specifies the full path to the working directory for the program in the settings setup.
set1	This setup parameter specifies the first of *n* number DOS/WIN-OS/2 settings. This is a boolean-type setting. The setting is interpreting 0 as ON and 1 as OFF.
set2	This setup parameter specifies the second of *n* number DOS/WIN-OS/2 settings. This is an integer-type setting. The setting is interpreting the 65 as a number.
set3	This setup parameter specifies the third of *n* number DOS/WIN-OS/2 settings. This is a string-type setting. The setting is interpreting the C:\AUTOEXEC.BAT literally.
set4	This setup parameter specifies the fourth of *n* number DOS/WIN-OS/2 settings. This is an enumerated-type setting. The setting is interpreting the NONE literally. The string must match exactly what is available in the enumerated list of selections in the DOS/WIN-OS/2 settings.

continued

Parameter	What the parameter does
set5	This setup parameter specifies the fifth of *n* number DOS/WIN-OS/2 settings. This is a Multiple Line Editor(MLE)-type setting. The setting is interpreting the DBASE.EXE,5,00,255 literally. The string must match the syntax for this setting. The DBASE.EXE is the name of the executable to be told the unique DOS version that the DOS emulation session is running under. The 5 specifies the 5 of DOS version 5.00. The 00 specifies the 00 of DOS version 5.00. The 255 specifies the number of times you want to be able to reply to the executable this different DOS version, in this case, an unlimited number of times.
open	This setup parameter specifies the action the object should take upon being executed. DEFAULT specifies that the object will OPEN every time the icon is selected.

Tip

It is best to use all upper case letters for string, enumerated, and MLE settings data just to ensure that there is no problem when the object saves the parameters.

Warning

If you deviate from these specifications, the settings will be passed garbage. Attempting to run a session with garbage for settings will cause a settings dialog to be displayed stating that the system has a problem with the settings. The only recourse is to manually modify the settings before the session will run.

Note

More settings can be included by using the same double || characters as separators.

WIN-OS/2 Window — Separate Session

In the settings notebook for a WIN-OS/2 Window you can specify whether the window started should use its own separate session or be started as part of another existing session. This setting can help you get around device-sharing problems when you have two applications that both want access during initialization to the same piece of hardware.

For example, the MCS MusicRack and America Online icons will both want to touch COM ports and the sound card ports during initialization and will cause a long delay in startup. This interference can occur when using separate sessions. Starting with applications in the same session eliminates this delay.

WIN-OS/2 TIMER.DRV Changes for MIDI

In order to support the limitation of VDM in OS/2, the maximum timer interrupt rate per VDM needs to be varied between 55 and 125 interrupts per second. TIMER.DRV needs to change the TDD_MAX386RESOLUTION down from 1000 to this new value. This will speed up the drawing and animation action needed for multimedia applications.

WIN-OS/2 Media player running on ISA machines with a SoundBlaster Pro card needs the interrupts per second to be varied lower than the value of 125, which is hardcoded for the TIMER.DRV supplied with OS/2 2.1. The following TIMER.DRV section is automatically added to the end of your installed \WINDOWS\SYSTEM.INI file by 4029OW21.EXE when the printer install occurs:

```
[timer.drv]
TIMERMax386Res=10
TIMERMax286Res=10
```

In the example given above, the interrupts per second value is determined by the value of the divisor *n* in 1000/n. In OS/2 2.1, the TIMER.DRV divisor value is hardcoded to be 1000/8=125 interrupts per second. In OS/2 Warp, the TIMER.DRV divisor value from the \WINDOWS\SYSTEM.INI is set up as 1000/10=100 interrupts per second.

The divisor should not be above 18 since the resulting interrupts/second minimum is 55. The TIMERMax386Res=10 statement is used when the WIN-OS/2 setting WIN_RUN_MODE is set to ENHANCED. The TIMERMax286Res=10 statement is used when the WIN-OS/2 setting WIN_RUN_MODE is set to STANDARD.

Summary

You should feel empowered by the knowledge you've gained from this chapter on the DOS settings. In many cases you won't have to change the settings, but if you do, this chapter should eliminate some of the potential confusion.

It is best to think of DOS settings as a superset replacement for some of the CONFIG.SYS settings, which almost everyone changes at some point. Remember, you can mess up your CONFIG.SYS settings so that DOS will not come up. The same is true with the DOS settings. You just have to be conscious of what you are doing when you change a setting and be aware of the implications of the changes you make.

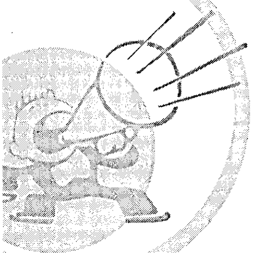
John M. Hare is a Senior Associate Programmer at IBM Boca Raton. He has been with the OS/2 Development Team since 1985, working on DOS emulation and WIN-OS/2 support. He has written code for many OS/2 Warp features, including DOS settings categories, printing and loading of the DOS Settings, displaying of the OEM DOS settings, and TIMER.DRV divisor value settings in SYSTEM.INI.

Appendix A

CD-ROM Contents

The CD-ROM included at the back of this book is divided into the following subdirectories:

- BITMAPS — Contains bitmaps that you can use for the background of your Desktop or for any other folder.
- ICONS — Contains icons that you can use to replace or supplement the standard OS/2 Warp icons on folders and other objects.
- OBJTOOL — Contains an Object Tool that can display and change the properties of any Workplace Shell object.
- POINTERS — Contains individual pointers and pointer sets that you can use to replace the standard mouse pointers.
- REXX — Contains sample REXX scripts.



- ✗ REXX\CREATE — Contains sample REXX scripts to recreate or fix all of the standard Workplace Shell objects. This does not include any of the optional OS/2 components, such as Multimedia or the BonusPak.

Each of these directories contains its own README file that lists all of the files in that directory and what they contain.

Bitmaps

The BITMAPS directory contains a set of high-quality images that can be used for the background image of your Desktop, any Workplace Shell folder, or Lockup. To use any of these images, copy them from the CD-ROM to a directory on your fixed disk and specify the image file on the appropriate Settings Notebook page.

The images on the CD-ROM are a mixture of small, tileable images (like the bitmaps supplied with OS/2 Warp) and large, full-screen images.

Table A-1 contains a summary of the images and their sizes. The README file in the BITMAPS directory contains a complete list.

Table A-1. BITMAPS directory contents

File Name	Description	Size	Mode
BLUBRICK.BMP	blue bricks	150x150	Tiled
BLUEDROP.BMP	blue water drops	150x150	Tiled
BOCAGRND.BMP	underwater scene	601x606	Normal
CLOUDS.BMP	clouds	256x256	Tiled
CONCRETE.BMP	concrete	150x150	Tiled
DARKBLU.BMP	textured, dark blue background	150x150	Tiled
DESERT.BMP	full-screen desert landscape	1024x768	Normal
DUNES.BMP	sand dunes	150x150	Tiled
FALCO.BMP	descending falcon	1024x768	Normal
FLAMES.BMP	makes your desktop look like it's on fire	150x150	Tiled

File Name	Description	Size	Mode
GRMARBLE.BMP	green marble	150x150	Tiled
JEANS1.BMP	alternative blue jeans image	150x150	Tiled
LEDGES.BMP	colorful full-screen underwater image	1024x768	Normal
MARS3.BMP	original full-screen rendering of a voyager fly-by of Mars	1024x768	Normal
MOONSKY.BMP	nighttime seascape	1024x768	Normal
NEBULA.BMP	enhancement of a NASA photo of a nebula	1024x768	Normal
PALMSKY.BMP	small image of a blue sky as viewed through the leaves of a palm tree	320x200	Normal or Scaled
PAPYRUS.BMP	ancient writing paper	150x150	Tiled
RIPPLES.BMP	rippling water	150x150	Tiled
ROPE.BMP	rope	150x150	Tiled
SAND.BMP	sand	150x150	Tiled
SHARK.BMP	a shark	320x200	Normal or Scaled
TYEDYE1.BMP	the look of tie-dye	300x300	Tiled
TYEDYE2.BMP	another tie-dye look	150x150	Tiled
UNCENSRD.BMP	full-screen image of the cover art for this book	1024x768	Normal
WARPING.BMP	full-screen image of the OS/2 Warp logo over a melting desktop	1024x768	Normal
WATER.BMP	water	150x150	Tiled
WOODPNL.BMP	wood paneling	150x150	Tiled
WTRLILLI.BMP	water lillies (not the Monet version)	150x150	Tiled
W_LOGO1.BMP	tiled, gray version of the OS/2 Warp logo	140x69	Tiled

Icons

The ICONS directory contains a set of icons that can be used for any Workplace Shell object. To use any of these icons, open to the General page (or General page 2 to replace the animation icon for a folder) of the object for which you want to change the icon, and use the **Find...** push button to locate the icon on the CD-ROM.

Alternatively, you can open the Disk object in the Drives folder corresponding to the CD-ROM and copy the icon to the General page of the object you want to customize. Since the Workplace Shell saves the customized icon in the object's data, you do not have to copy the icon files from the CD-ROM before using them.

Table A-2 contains a summary of the icons. The README file in the ICONS directory contains a complete list.

Table A-2. ICONS directory contents

File Name (Normal or Closed Folder)	File Name (Open Folder)	Description	Intended Use
BLACKHOL.ICO		black hole	an object that deletes other objects
BOOK.ICO		a book	any book
BOOKC.ICO		book with the letter *C*	C reference
BOOKCP.ICO		book with the letters *CP*	Control Program reference
BOOKDOS.ICO		book with the DOS logo	DOS reference
BOOKLAN.ICO		book with the Network logo	LAN reference
BOOKMMPM.ICO		book with musical notes	multimedia reference
BOOKOS2.ICO		book with the OS/2 logo	OS/2 reference
BOOKPM.ICO		book with the letters *PM*	PM reference
BOOKTOOL.ICO		book with a wrench image	Toolkit reference
BOOKWIN.ICO		book with the Windows logo	Windows reference
DISKETTE.ICO		3.5-inch diskette without a letter	3.5-inch diskette drive

File Name (Normal or Closed Folder)	File Name (Open Folder)	Description	Intended Use
DRIVE.ICO		drive without a letter	fixed disk
DRIVEA.ICO		3.5-inch drive with the letter *A*	A: drive — 3.5"
DRIVEB.ICO		3.5-inch drive with the letter *B*	B: drive — 3.5"
DRIVEA5.ICO		5.25-inch drive with the letter *A*	A: drive — 5.25"
DRIVEB5.ICO		5.25-inch drive with the letter *B*	B: drive — 5.25"
DRIVEC.ICO through DRIVEZ.ICO		hard drive with the letters *C* through *Z*	C: through Z: drives
FAMIPRO.ICO	FAMIPRO2.ICO	AmiPro folder	AmiPro folder
FAPPS.ICO	FAPPS2.ICO	PM apps folder	PM applications folder
FCAL.ICO	FCAL2.ICO	Time and Place/2 folder	TAP/2 folder
FCOREL.ICO	FCOREL2.ICO	Corel Draw folder	Corel Draw folder
FDESCR.ICO	FDESCR2.ICO	Describe folder	Describe folder
FDOS.ICO	FDOS2.ICO	DOS folder	DOS applications folder
FGAMESA.ICO	FGAMESA2.ICO	games folder	games folder
FGAMESB.ICO	FGAMESB2.ICO	another games folder	games folder
FGOPHER.ICO	FGOPHER2.ICO	gopher folder	Gopher folder
FIBMCPP.ICO	FIBMCPP2.ICO	IBM C-Set folder	IBM C-Set++ Folder
FLOT123.ICO	FLOT1232.ICO	Lotus 1-2-3 folder	Lotus 1-2-3 folder
FNET.ICO	FNET2.ICO	network folder	network folder
FOLDER.ICO	FOLDER2.ICO	yellow folder	any folder
FOLDERB.ICO	FOLDERB2.ICO	blue folder	any folder
FOLDERG.ICO	FOLDERG2.ICO	green folder	any folder
FOLDERP.ICO	FOLDERP2.ICO	pink folder	any folder
FOLDERY.ICO	FOLDERY2.ICO	gray folder	any folder
FOS2.ICO	FOS22.ICO	OS/2 folder	OS/2 System Folder

continued

Table A-2. ICONS directory contents, continued

File Name (Normal or Closed Folder)	File Name (Open Folder)	Description	Intended Use
FSOUNDS.ICO	FSOUNDS2.ICO	sounds folder	sound bites folder
FTOOLS.ICO	FTOOLS2.ICO	Toolbox folder	a folder containing your tools
FVISAGE.ICO	FVISAGE2.ICO	Visual Age folder	Visual Age folder
FWIN.ICO	FWIN2.ICO	Windows folder	Windows applications folder
FWP.ICO	FWP2.ICO	WordPerfect folder	WordPerfect folder
FXL.ICO	FXL2.ICO	Excel folder	Excel folder

Object Tool

The OBJTOOL directory contains an Object Tool that can be used to display and change the properties of any draggable Workplace Shell object.

To install the Object Tool, open an OS/2 command prompt window, make the OBJTOOL directory on the CD-ROM the current directory, and type **install.** This will install the Object Tool and create an Object Tool object on your Desktop.

To use the tool, drag any other Workplace Shell object to the Object Tool object.

You can read about the Object Tool in the help information that is provided as part of the Object Tool.

Pointers and Pointer Sets

The POINTERS directory contains pointers and pointer sets. Most of the pointers and some of the pointer sets are corrected versions of the pointers and pointer sets that are included with OS/2 Warp. Some of the pointers and pointer sets are new.

To use these pointers and pointer sets, copy them from the CD-ROM to the /OS2/POINTERS directory on your boot drive. These will replace the incorrect OS/2 Warp pointers and pointer sets with the corrected versions and add the new pointers and pointer sets. You can use these pointers and pointer sets in the Pointers page of the Mouse object.

Table A-3 contains a summary of the pointers and pointer sets. The README file in the POINTERS directory contains a complete list.

Table A-3. POINTERS directory contents

File/Directory Name	Description	New or Replacement
ARROW.PTR	hand (arrow pointer)	replacement
NEWBWAIT.PTR	black clock (wait pointer)	replacement
NEWWWAIT.PTR	white clock (wait pointer)	replacement
LH_ARROW.PTR	left-handed hand (arrow pointer)	new
BLACK	black pointer set	replacement
WHITE	white pointer set	replacement
BIG_BLAC	big, black pointer set	replacement
BIG_WHIT	big, white pointer set	replacement
HANDS	hand pointer set	new
LH_BLACK	left-handed black pointer set	new
LH_WHITE	left-handed white pointer set	new
LH_BIG_B	left-handed big black pointer set	new
LH_BIG_W	left-handed big white pointer set	new
LH_HANDS	left-handed hand pointer set	new

REXX Scripts

The REXX directory contains sample REXX scripts. Most of these scripts contain examples of techniques mentioned elsewhere in this book.

Table A-4 contains a summary of the REXX scripts. The README file in the REXX directory contains a complete list.

Table A-4. REXX directory contents

File Name	Description
PLAYSND.CMD	play a random startup sound
PLAYVOL.CMD	play a .WAV file at a specified volume
SCHEME.CMD	create OS/2 Warp Scheme Palette
SETSND .CMD	set a random shutdown sound
VIDBOUNC.CMD	play a video file at a random place and size

The REXX\CREATE directory contains a set of REXX scripts that can be used to recreate or fix the standard OS/2 objects.

For example, if you accidentally delete the Games folder, you can recreate it by invoking the GAMES.CMD script. If you change the icon for the System Setup folder and later want to get the original icon back, you can invoke the SYSSETUP.CMD script.

Table A-5 contains a summary of the REXX scripts included in the REXX\CREATE directory. The README file in the REXX\CREATEdirectory contains a complete list.

Table A-5. REXX\CREATE directory contents

File Name	What It Creates	Where It Is Created
ADDPGMS.CMD	Add Programs	System Setup folder
APPLICAT.CMD	Application Considerations Book	Information folder
CASTLE.CMD	CASTLE.MAH	Mahjongg Solitaire folder
CHESS.CMD	OS/2 Chess game	Games folder
CLIPBOAR.CMD	Clipboard Viewer	Productivity folder
CMDREF.CMD	Command Reference Book	Information folder
COUNTRY.CMD	Country	System Setup folder
CREATUTL.CMD	Create Utility Diskettes	System Setup folder
DDINSTAL.CMD	Device Driver Install	System Setup folder
DEFAULT.CMD	DEFAULT.MAH	Mahjongg Solitaire folder

File Name	What It Creates	Where It Is Created
DESKTOP.CMD	Desktop folder and its contents	Root directory of boot drive
DOSFROMA.CMD	DOS From Drive A:	Command Prompts folder
DOSFS.CMD	DOS Full Screen	Command Prompts folder
DOSWIN.CMD	DOS Window	Command Prompts folder
DRIVES.CMD	Drives folder	OS/2 System folder
DUALBOOT.CMD	Dual Boot	Command Prompts folder
ENHANCED.CMD	Enhanced Editor	Productivity folder
FONTPAL.CMD	Font Palette	System Setup folder
GAMES.CMD	Games folder	OS/2 System folder
GIZEH.CMD	GIZEH.MAH	Mahjongg Solitaire folder
GLOSSARY.CMD	Glossary	Information folder
HELPINDX.CMD	Master Help Index	Information folder
ICONEDIT.CMD	Icon Editor	Productivity folder
INFORMAT.CMD	Information folder and its contents	Desktop folder
KEYBOARD.CMD	Keyboard	System Setup folder
LAUNCHPA.CMD	LaunchPad	Desktop folder
MAHFLDR.CMD	Mahjongg Solitaire folder and its contents	Games folder
MAHJONGG.CMD	Mahjongg Solitaire game	Mahjongg Solitaire folder
MINIMIZE.CMD	Minimized Windows Viewer	OS/2 System folder
MIXCOLOR.CMD	Mixed Color Palette	System Setup folder
MOUSE.CMD	Mouse	System Setup folder
MULTIMED.CMD	Multimedia Book	Information folder
NETWORK.CMD	Network folder	Desktop folder
NOWHERE.CMD	Nowhere folder	Root directory of boot drive
OS2FS.CMD	OS/2 Full Screen	Command Prompts folder

continued

Table A-5. REXX\CREATE directory contents, continued

File Name	What It Creates	Where It Is Created
OS2SYS.CMD	OS/2 System folder	Desktop folder
OS2WIN.CMD	OS/2 Window	Command Prompts folder
PERFORMA.CMD	Performance Considerations Book	Information folder
PICTURE.CMD	Picture Viewer	Productivity folder
PLUGPLAY.CMD	Plug and Play	System Setup folder
POWER.CMD	Power	System Setup folder
PRINTER.CMD	Printer	Desktop folder
PRINTING.CMD	Printing in OS/2 Book	Information folder
PRODUCTI.CMD	Productivity folder	OS/2 System folder
PROMPTS.CMD	Command Prompts folder and its contents	Productivity folder
PULSE.CMD	Pulse	Productivity folder
README.CMD	README File	Information folder
REXXINFO.CMD	REXX Information Book	Information folder
SCHEMPAL.CMD	Scheme Palette	System Setup folder
SEEKSCAN.CMD	Seek and Scan files	Productivity folder
SELINSTL.CMD	Selective Install	System Setup folder
SELUNINS.CMD	Selective Uninstall	System Setup folder
SHREDDER.CMD	Shredder	OS/2 System folder
SOLCOLOR.CMD	Solid Color Palette	System Setup folder
SOLITAIR.CMD	Solitaire — Klondike Game	Games folder
SOUND.CMD	Sound	System Setup folder
SPOOLER.CMD	Spooler	System Setup folder
STARTUP.CMD	Startup folder	OS/2 System folder
SYSCLOCK.CMD	System Clock	System Setup folder

File Name	What It Creates	Where It Is Created
SYSEDIT.CMD	OS/2 System Editor	Productivity folder
SYSSETUP.CMD	System Setup folder and its contents	OS/2 System folder
SYSTEM.CMD	System	System Setup folder
TAJMAHAL.CMD	TAJMAHAL.MAH	Mahjongg Solitaire folder
TEMPLATE.CMD	Templates folder	Desktop folder
TILEDEF.CMD	TILEDEF.DLL	Mahjongg Solitaire folder
TILEHK.CMD	TILEHK.DLL	Mahjongg Solitaire folder
TOUCH.CMD	Touch	System Setup folder
TOWERBRI.CMD	TOWERBRI.MAH	Mahjongg Solitaire folder
TRADEMAR.CMD	Trademarks Book	Information folder
TUTORIAL.CMD	Tutorial	Information folder
VIEW.CMD	View	Nowhere folder
WINCFG.CMD	WIN-OS/2 Setup	System Setup folder
WINDOWSP.CMD	Windows Programs in OS/2 Book	Information folder
WINOS2FS.CMD	WIN-OS/2 Full Screen	Command Prompts folder
WINOS2WI.CMD	WIN-OS/2 Window	Command Prompts folder

Appendix B

Developer's Desktop

Throughout this book, we have shown you glimpses of our own system setups, both at home and at the office. In this appendix, I will take you on a complete guided tour of my home Desktop.

I don't expect you to run right out and set up your system just like mine. In fact, I would be surprised to find out *anyone* had a system like mine. However, I want to show you how I set up my system to give you some ideas. There are some rather strange techniques I used in setting up my system that might be useful to you. Even if you don't find anything I did particularly interesting, perhaps seeing how I twisted the system around might inspire you to make your system more friendly to you.

Desktop Basics

Before I show you my system, let me tell you a little about the environment I work in at home.

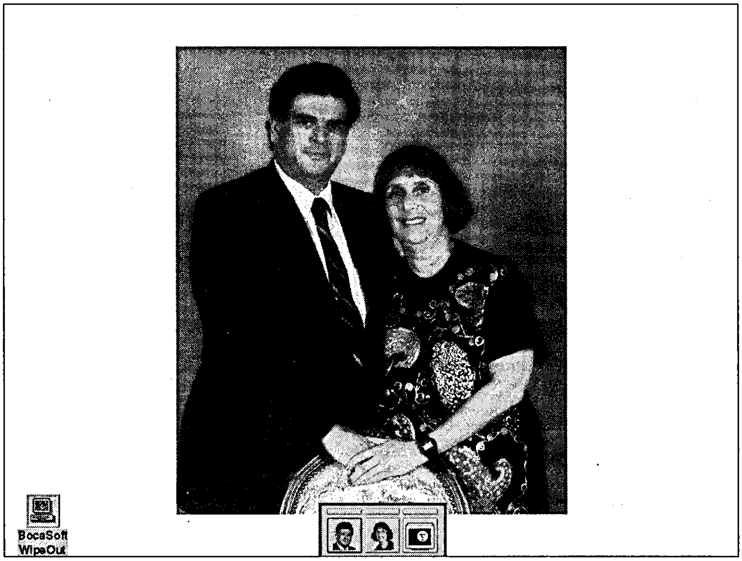
There are two computer users in my house — my wife, Lorraine, and me. As you can probably guess, I am an "expert" user, but Lorraine is a novice. I wanted to tailor the system so Lorraine would have enough functions available to her without having the capability of accidentally messing up my view of the system.

The first thing I had to do was find a way to let Lorraine see a different view of the system than I see. I chose Work Area folders as the means for accomplishing this. Since each of us has our own Work Area folder, there was no need for any objects to remain on the Desktop, except for a mechanism for specifying who was using the system. Figure B-1 shows my Desktop. It is completely empty (unlike my real desktop at work or my computer workstation at home). The only thing you see is the LaunchPad, which has three icons on it: one to open my Work Area folder, one to open Lorraine's Work Area folder, and one to shut down the system.

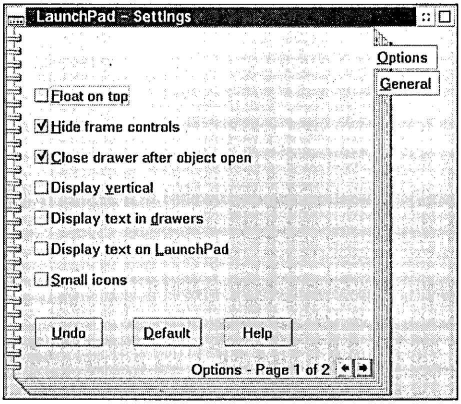
Note

I can't show you the background bit map I normally use, since it is a picture of the cast of "Star Trek: The Next Generation," which is copyrighted by Paramount Pictures. Instead, you get to see a photograph of Lorraine and me.

Figure B-1.
Desktop Folder



The third icon on the LaunchPad isn't the Shutdown action push button that comes with the LaunchPad. If you have experimented with the LaunchPad you will find that you can't have only one of the action buttons — you either have all four or you have none. I selected none, installed a Shutdown object (that doesn't come with OS/2 Warp), and put an icon on the LaunchPad for Shutdown. The icon itself is the same one used in OS/2 Warp. Figure B-2 and Figure B-3 show how my Desktop LaunchPad is customized.



LaunchPad - Settings

□ Float on top

☑ Hide frame controls

☑ Close drawer after object open

□ Display vertical

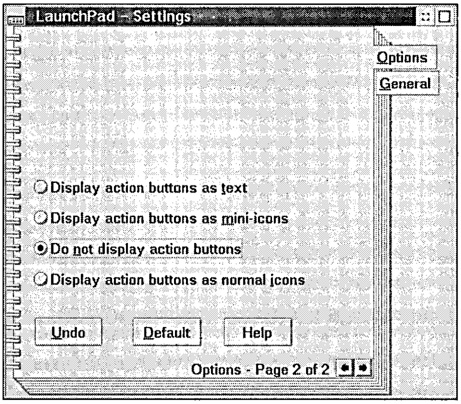
□ Display text in drawers

□ Display text on LaunchPad

□ Small icons

Options - Page 1 of 2

Figure B-2. Desktop LaunchPad - Options Page 1



LaunchPad - Settings

○ Display action buttons as text

○ Display action buttons as mini-icons

● Do not display action buttons

○ Display action buttons as normal icons

Options - Page 2 of 2

Figure B-3. Desktop LaunchPad - Options Page 2

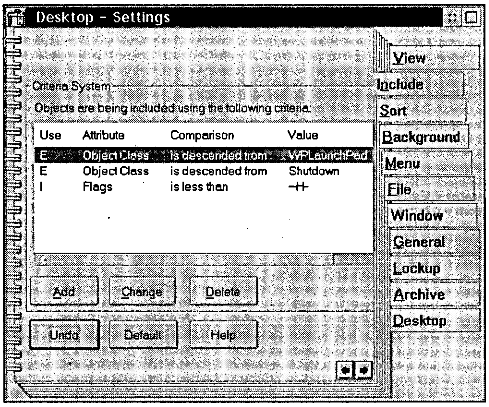
The two Work Area folders themselves don't show up on the Desktop because their directories are not based on \DESKTOP. They are in the root directory of the boot drive. My \DESKTOP directory really is empty.

Note

When I originally set up this system under OS/2 2.0, I couldn't think of any other way of structuring it, so the folders weren't visible. However, with OS/2 Warp, I can have my Work Area folders be children of the Desktop folder and suppress their icons using the Include page of the Desktop's Settings Notebook.

Figure B-4 shows the Include page of my Desktop's Settings Notebook. I added two exclusion criteria to hide the LaunchPad and the Shutdown object. That is why the Desktop appears completely clean.

Figure B-4. Desktop Settings Notebook - Include Page



Criteria System

Objects are being included using the following criteria:

Use	Attribute	Comparison	Value
E	Object Class	is descended from	WPLaunchPad
E	Object Class	is descended from	Shutdown
I	Flags	is less than	-H-

The buttons on the LaunchPad could have directly opened the Work Area folders, but what fun would that be? Instead, they open program reference objects that run a REXX script (see Figure B-5). This REXX script lets me do some processing before opening the folder. In this script, I do the following

1 Determine which folder to open based on the argument passed to the script

2 Call the internal function IraPreOpen or LorrainePreOpen

- IraPreOpen — Performs no function

- LorrainePreOpen — Opens Lorraine's nighttime Work Area folder if it is after 6 p.m.

3 Open the appropriate folder

4 Put the name of the person opening the folder into C:\FLAGS\WHO.FLG

5 Call the internal function IraInit or LorraineInit

- IraInit — Invokes IBMAV2 once a week; invokes BACKUP.CMD once a day
- LorraineInit — Performs no function

The C:\FLAGS\WHO.FLG file is read by several other REXX scripts to determine who is using the system. This is a very primitive "logon" facility.

Figure B-5. REXX Script To Open Work Area Folders

```
/* Ira or Lorraine is opening their desktop folder */
/* (1) /
arg DesktopName
if Pos('IRA',DesktopName)>0 then
  Who = 'IRA'
else
  Who = 'LORRAINE'
'@echo off'

/* Load the REXX utilities */
Call RxFuncAdd 'SysLoadFuncs','REXXUTIL','SysLoadFuncs'
Call SysLoadFuncs

/* Do the pre-open testing */
/* (2) */
interpret 'call' who || 'PreOpen'

/* Open the specified desktop folder */
/* (3) */
Call SysSetObjectData DesktopName,'open=default'

/* Indicate which folder is open */
/* (4) */
FlagPath = Value('flagpath',,'OS2ENVIRONMENT')
FlagPath = 'c:\flags'
Flag = FlagPath || '\who.flg'
call lineout Flag,Who,1
call lineout Flag

/* Do the appropriate initialization for the folder and exit */
/* (5) */
interpret 'call' who || 'Init'
'exit'

/* Ira is opening his desktop folder:
     No processing required */
/* (2) */
IraPreOpen:
return
```

```
/* Lorraine is opening her desktop folder:
     Only let her open her folder before 6:00 PM */
/* (3) */
LorrainePreOpen:
if Time('H')>=18 then
  do
    DesktopName="D:\LORRAIN1"
  end
return

/* Ira opened his desktop folder:
     Invoke IBMAV2 once a week */
/* (5) */
IraInit:
'call nthday c:\cmd\nthday.dat'
LastAccessFile=FlagPath || '\last.acc'
Today=Date()
LastAccess=Linein(LastAccessFile)
if Today<>LastAccess then
  do
    RC=Lineout(LastAccessFile,Today,1)
    RC=Lineout(LastAccessFile)
    /* Stuff to be done once a day goes here:
         Perform daily backup */
    address cmd 'start backup.cmd'
  end
return

/* Lorraine opened her desktop folder:
     No processing required */
/* (5) */
LorraineInit:
return
```

As you can see, this REXX script enables the system to perform some basic functions when Lorraine or I start using it. In Lorraine's case, the script chooses which view of the system she sees — her normal Work Area or her nighttime Work Area. In my case, it performs some special processing that I would normally have done at startup, but that I don't want bothering Lorraine if she's started up the system.

System Startup

Speaking of startup, Figure B-6 shows my STARTUP.CMD file. It performs the following functions:

1. Plays a random system startup sound (see Figure B-7). You saw most of the PLAYSND.CMD command file at the end of Chapter 8.
2. Sets a random system shutdown sound (see Figure B-8). You also saw most of the SETSND.CMD command file at the end of Chapter 8.
3. Starts the Comet Cursor.
4. Initializes the printer to PostScript mode. For reasons that escape me, I have my IBM 4019 LaserPrinter start up in IBM Personal Printer Data Stream (PPDS) mode, even though I use the printer in PostScript mode most of the time.
5. Clears the flags used by some of my REXX scripts. I use the presence of files in the C:\FLAGS directory like semaphores.
6. Starts the screen saver program, WipeOut.

Figure B-6.
STARTUP.CMD File

```
/* This is a Rexx procedure */
'@echo off'
/* (1) */
'start /min playsnd.cmd'
/* (2) */
'call setsnd.cmd'
/* (3) */
'start cometrun.exe'
/* (4) */
'copy c:\dos\vdisk\printer\4019ps lpt1 >nul'
/* (5) */
FlagPath = Value('flagpath',,'OS2ENVIRONMENT')
if stream(FlagPath || '\*.flg','c','query exists')<>''
then
  'del %flagpath%\*.flg >nul'
/* (6) */
'c:'
'cd \wipeout'
'start /min wipeout.exe'
'exit'
```

Figure B-7.
PLAYSND.CMD File

```
/* Play Random Startup Sound Effect */
call RxFuncAdd 'SysLoadFuncs','RexxUtil','SysLoadFuncs'
call SysLoadFuncs
'@echo off'

/* Select random Startup Sound Effect */
call SysFileTree 'c:\mmos2\sounds\startup\*.wav',startup,,
                 'o'
ix=Random(1,startup.0)

/* Play Startup Sound Effect */
'call playvol file='startup.ix' volume=60'
'exit'
```

Note

PLAYVOL.CMD is a modified version of the PLAY.CMD file that allows me to specify the volume for the sound effect.

Figure B-8.
SETSND.CMD File

```
/* Select Startup/Shutdown Sound Effects */
call RxFuncAdd 'SysLoadFuncs','RexxUtil','SysLoadFuncs'
call SysLoadFuncs
'@echo off'

/* Turn off Startup Sound Effect */
call Sysini 'd:\mmos2\mmpm.ini','MMPM2_AlarmSounds','7',,
            '#System startup#60'X2C('00')

/* Select random Shutdown Sound Effect */
call SysFileTree
'c:\mmos2\sounds\shutdown\*.wav',shutdown,,
                 'o'
ix=Random(1,shutdown.0)
call Sysini 'd:\mmos2\mmpm.ini','MMPM2_AlarmSounds','8',,
            shutdown.ix'#System shutdown#60'X2C('00')
```

My Startup folder is not too interesting, as you can see in Figure B-9. It only contains the System Clock.

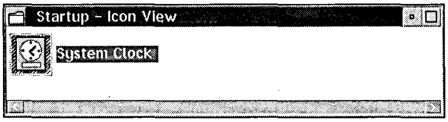


Figure B-9.
Ira and Lorraine's Startup Folder

Note

You might wonder why I started WipeOut from the STARTUP.CMD file rather than from the Startup folder. I had originally put a shadow of the WipeOut object in the Startup folder. Occasionally I ran into a problem where the system would announce that the WipeOut program couldn't be started and that my settings must be incorrect. I was never able to track down the cause of this problem, but moving the startup of WipeOut into the STARTUP.CMD file bypassed the problem. I start many programs from the Startup folder at work, including WipeOut, so I don't know what my system was complaining about.

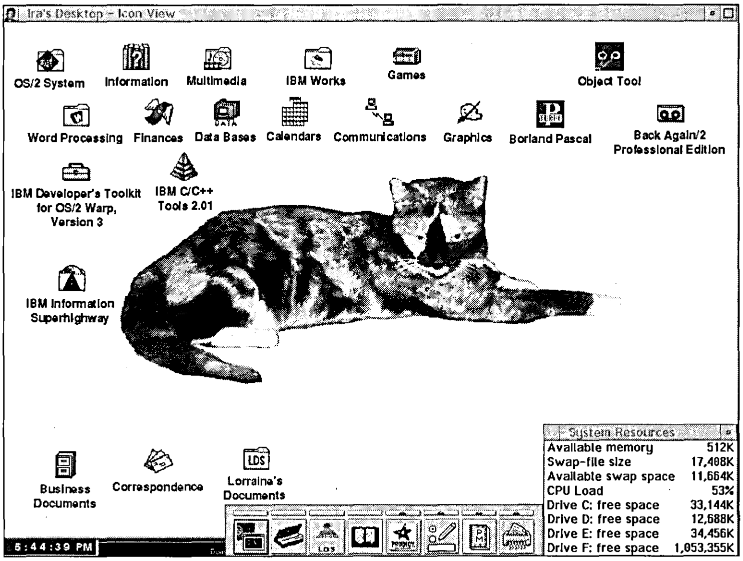
Work Area Folders

OK, back to my Desktop setup. Figure B-10 shows my Work Area folder.

Note

Once again, I can't show you the bit map I normally use for a background, so you will have to settle for a picture of Natasha, one of our cats.

Figure B-10.
Ira's Work Area Folder



System Resources

```
Available memory              512K
Swap-file size             17,408K
Available swap space       11,664K
CPU Load                       53%
Drive C: free space        33,144K
Drive D: free space        12,688K
Drive E: free space        34,456K
Drive F: free space     1,053,355K
```

This looks a little more familiar. I put the standard OS/2 Warp folders (OS/2 System, Information, and Multimedia) on the top row. I moved other standard objects around (such as the Shredder and the Templates folder) and put most of them into the OS/2 System folder.

The second row of objects are folders that contain programs grouped into various categories. I drew some of these icons, but I acquired most of them from public domain sources. For example, there are many packages of Windows icons available on public bulletin boards and FTP sites. These icons are directly usable by OS/2 Warp. Other icons, like the Word Processing icon, I constructed by overlaying part of one icon (the Word for Windows icon) on another (the standard Folder icon). I also modified many of the icons to add an open folder icon to the folders.

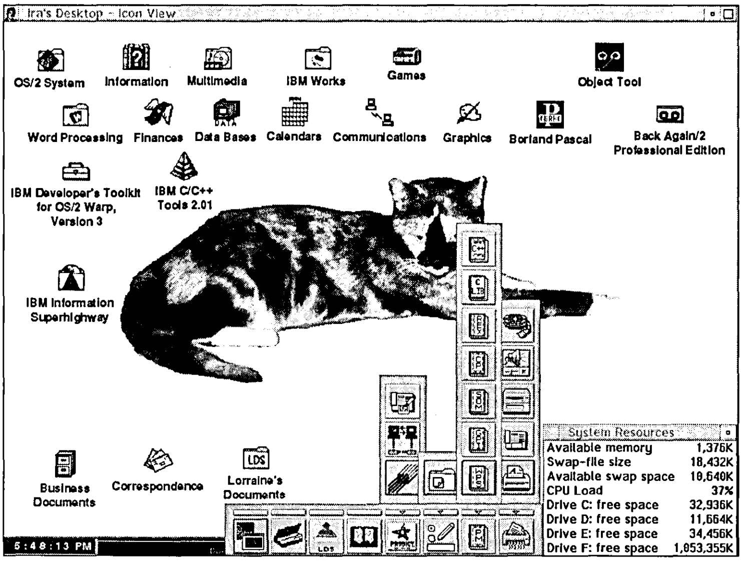
The third row of objects are mostly folders containing various tools that I use. Even though I am a programmer working for IBM, I also enjoy developing programs at home, so you can see the OS/2 Developer's Toolkit and the IBM C++ folders. I also use the IBM Internet Connection to connect to the Internet and to connect to our internal network at IBM.

The row of objects near the bottom of the folder are shadows of directories containing word processing files. Lorraine runs a typing service business at home, and the Business Documents icon represents the folder containing all of her client folders. Figure B-12 shows the Desktop with this folder open. (Actually, the Business Documents folder is a Work Area folder that automatically opens the WINWORD folder, which contains our business documents in Word for Windows format.) If you look carefully, you will see the icon for the Business Documents folder now shows a filing cabinet with an open drawer. It is relatively easy to construct an open folder icon using the Icon Editor and attach it to the folder from the second General page of the folder's Settings Notebook.

In Figure B-10 you can also see two objects that open automatically when I open the Work Area folder: the System Resources object, at the lower right corner of the screen, and a LaunchPad, at the bottom of the screen. This LaunchPad contains buttons for my most commonly used programs: an OS/2 windowed command prompt, my Checkbook program, the Dome Accounting program, an Edit Log Book program, Prodigy, the Dialog Editor, the Presentation Manager reference manual, and the Shredder. Several of these buttons have drawers of related programs associated with them. Figure B-11 shows the contents of all of these drawers.

Look at the drawer that is second from the right. You will notice a bunch of customized book icons. Again, I found that it doesn't take too much time to draw a couple of letters on an icon so I can identify the name of the book represented by the icon.

Figure B-11. Ira's LaunchPad With All Drawers Open



```
System Resources
Available memory            1,376K
Swap-file size             18,432K
Available swap space       10,640K
CPU Load                       37%
Drive C: free space        32,936K
Drive D: free space        11,664K
Drive E: free space        34,456K
Drive F: free space     1,053,355K
```

Word for Windows Setup

Let's look at my Business Documents folder in Figure B-12. This folder contains other folders, each of which is the name of a client. Each client folder contains the documents, or folders of documents for that client, depending on how much work we have done for that particular client.

Note

The Business Documents folder (and the client folders) are kept on a FAT partition so I can freely use long file names for these folders without upsetting Word for Windows. The long file names are shown by the Workplace Shell, while Word for Windows sees the normal FAT file names. The actual documents must have regular FAT file names, since Word for Windows would destroy the extended attributes containing the long file names when we edit the documents.

The New Document object is a dummy data file that lets me start Word for Windows without specifying a real data file. The New Folder object is a template of a folder that contains a New Document object. It lets me easily build a new client folder.

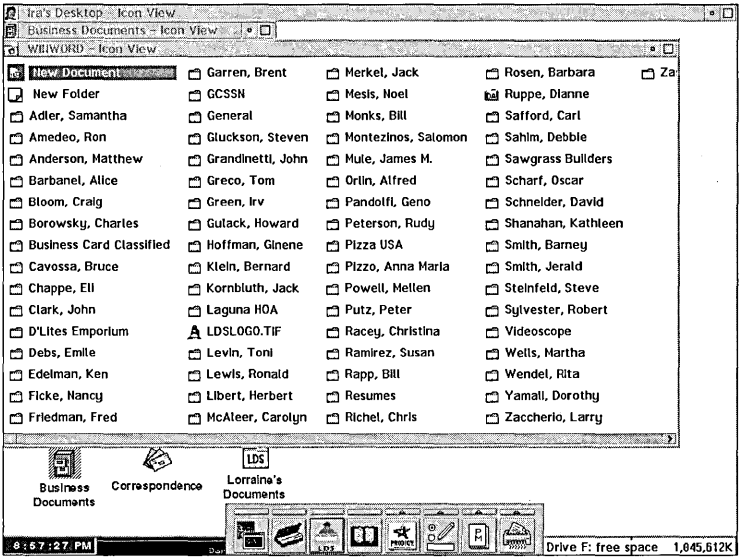


Figure B-12.
Business
Documents Folder

Word for Windows (Winword) allows you to specify a file name when you start the program. However, it doesn't automatically change its directory to the directory that contains the input file. This is rather annoying. Also, Lorraine and I wanted slightly different customizations for Word for Windows. Here is how I fixed both problems.

The Word for Windows object, which is in my Word Processing folder, invokes a REXX script instead of directly invoking Winword. Figure B-13 shows my WINWORD.CMD REXX script. This REXX script, in conjunction with the Autoexec macro I defined for Word for Windows (see Figure B-14), sets up the proper environment for Winword, invokes Winword with the specified file, and causes Winword to change to the directory in which the file resides.

The WINWORD.CMD script performs the following functions:

1. Loads the required utility functions. REXXUTIL contains the standard REXX Utility Functions; PMSW2 contains a set of REXX utilities someone developed here at IBM.
2. Writes the name of the current directory into the C:\FLAGS\WHERE.FLG file. This file is read by my Word for Windows Autoexec macro (see Figure B-14).
3. Copies the proper set of Word for Windows configuration files to the WINWORD directory.

4 Sets up to pass the file name to Word for Windows.

5 Opens the Word for Windows object. This was the only way I found to be able to open Word for Windows in a windowed WIN-OS/2 session.

6 Waits for Word for Windows to exit.

7 Updates the appropriate set of Word for Windows configuration files in case they changed.

8 Erases the C:\FLAGS\WHERE.FLG file.

Figure B-13.
WINWORD.CMD

```
/* Load REXX Utility functions */
/* (1) */
if 0 < RxFuncQuery('SysLoadFuncs') then
  do
    call RxFuncAdd 'SysLoadFuncs','RexxUtil','SysLoadFuncs'
    call SysLoadFuncs
  end
if 0 < RxFuncQuery('PMSW2') then
  do
    call RxFuncAdd 'PMSW2','PMSW2','PMSW2'
  end
/* Set up the proper path and invoke Word for Windows */
/* (2) */
env = 'os2environment'
/* Write the drive letter and path name to the
   WHERE.FLG file */
Arg FileSpecification
'@echo off'
FlagPath = Value('flagpath',,env)
FlagFile = FlagPath || "\where.flg"
if Stream(FlagFile,'c','query exists')<>"" then
  "erase" FlagFile
if FileSpecification <> "" then
  do
    Drive = FileSpec("drive",FileSpecification)
    Path = FileSpec("path",FileSpecification)
    Path = Substr(Path,1,Length(Path)-1)
    FileName = FileSpec("name",FileSpecification)
    if Substr(FileName,1,8) = "DESKTOP" then
      FileSpecification = ""
    if Substr(FileName,1,4) = "OS!2" then
      FileSpecification = ""
    if FileSpecification <> "" then
      do
        RC = Lineout(FlagFile,Strip(Drive||Path),1)
        RC = Lineout(FlagFile)
      end
  end
```

```
/* Set up the environment for Word for Windows */
/* (3) */
WinwordPath = 'c:\winword'
WhoFile = FlagPath || '\who.flg'
if Stream(WhoFile,'c','query exists')<>'' then
  do
    Who=Linein(WhoFile,1,1)
    call stream WhoFile,'c','close'
    'copy' WinwordPath || '\' || Who WinwordPath '>nul'
    'copy' WinwordPath || '\normal.dot' WinwordPath,
           || '\template >nul'
  end
else
  Who=''
/* Invoke Word for Windows */
/* (4) */
Parameters = "%"
if FileSpecification <> "" then
  if Substr(FileName,1,8) <> "_NEW_DOC" then
    Parameters = FileName
/* (5) */
call SysSetObjectData "<WINWORD>",,
     "PARAMETERS="Parameters";OPEN=DEFAULT"
/* Wait for Winword to start */
/* (6) */
WinwordName = '*Microsoft Word*'
do forever
   rc = PMSW2(WinwordName,'/r')
   if rc='READY:' then leave
   Call SysSleep 10
end
/* Wait for Winword to finish */
do forever
   rc = PMSW2(WinwordName,'/r')
   if rc='ERROR:' then leave
   Call SysSleep 5
end;
call SysSleep 5
/* Save the environment if it changed */
/* (7) */
'copy' WinwordPath,
       || '\template\normal.dot' WinwordPath '>nul'
if Who<>'' then
  'replace' WinwordPath || '\*' WinwordPath || '\',
            || Who '/u >nul'
/* Erase the WHERE.FLG file */
/* (8) */
if Stream(FlagFile,'c','query exists')<>"" then
  'erase' FlagFile '>nul'
```

Figure B-13 is an extraction from my Autoexec macro. I removed the parts of this macro that deal with the situation when Word for Windows is invoked without opening one of the Work Area folders (which can't happen) or if it is invoked directly without going through my WINWORD.CMD file.

Note

You might be interested in the macro only if you are using Word for Windows. However, the technique I used in building this macro might be useful for other programs.

Figure B-14. Word for Windows Autoexec Macro

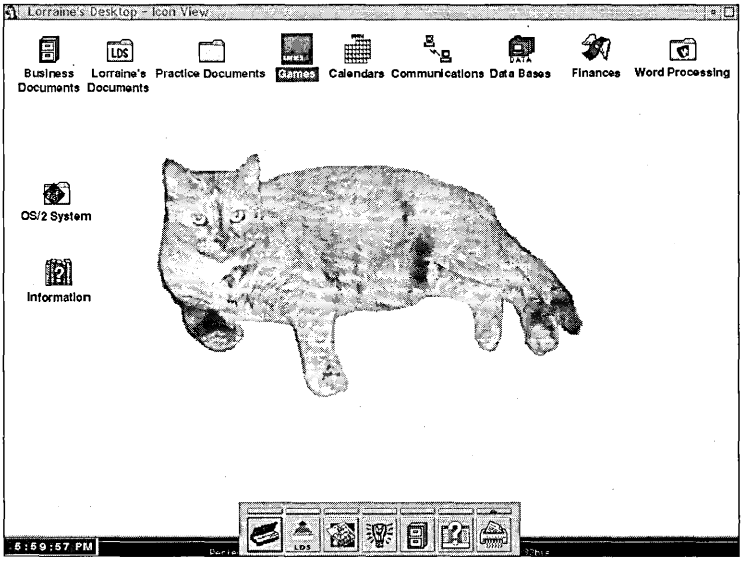
```
Sub MAIN
REM See if Ira or Lorraine's folder is open
On Error Goto NoWho
Open  "c:\flags\who.flg" For Input As #1
Input #1, Who$
If Who$ = "IRA" Then Ira = 1
If Who$ = "LORRAINE" Then Lorraine = 1
NoWho:
Close #1
Err = 0
REM See if any word processing document folder is open
On Error Goto NoWhere
Open "c:\flags\where.flg" For Input As #1
Input #1, Where$
If InStr(Where$, "BUSINESS") > 0 Then Business = 1
If InStr(Where$, "CORRESPO") > 0 Then Corr = 1
NoWhere:
Close #1
Err = 0
REM If the Who$ or Where$ flags are not set, ask the
REM user who he is and whether this is for business or not
   ...
REM Set up the default path for document files
ChDir Where$
SetPrivateProfileString "QLO","Path",Where$,"FILEMAN.INI"
REM Set up the private customizations
If Ira = 1 Then
  ToolsOptionsSave .AutoSave = 1, .SaveInterval = "15",
.SummaryPrompt = 1 \
: WW2_ToolsOptionsGeneral .Units = 0, .Pagination = 1,
.ReplaceSelection = 1 \
: ToolsOptionsUserInfo .Name = "Ira Schneider",
.Initials = "IHS"
Else
  ToolsOptionsSave .AutoSave = 1, .SaveInterval = "15",
.SummaryPrompt = 1 \
: WW2_ToolsOptionsGeneral .Units = 0, .Pagination = 1,
.ReplaceSelection = 1 \
: ToolsOptionsUserInfo .Name = "Lorraine Schneider",
.Initials = "LDS"
End If
End Sub
```

The rest of my Work Area folder is pretty standard. Each of the folders contains Program Reference objects for various programs.

Figure B-15 shows Lorraine's normal Work Area folder. Her Work Area is guarded by our other cat, Cleopatra. She (Lorraine, not the cat) has her own LaunchPad, which contains the objects she uses most. Also, she has a subset of the folders that I have, and each folder only contains the objects that she needs.

Lorraine's nighttime Work Area folder (see Figure B-16) is even more limited. It is essentially a copy of her normal Work Area folder, but some of the folders and objects are replaced with inoperative objects. (Actually, they point to a command file that displays a message saying she can't use that facility after 6 p.m.) It was easy to modify the normal icons to the "no" icons by using the Icon Editor. I just drew a circle (with a pen width of 2 x 2 pixels) around the normal icon, and then drew a diagonal line across it.

Figure B-15. Lorraine's Work Area Folder



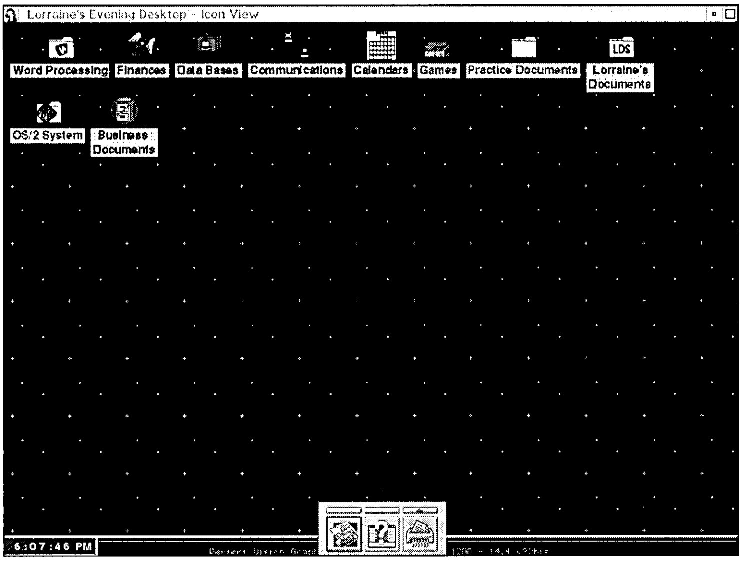


Figure B-16. Lorraine's Nighttime Work Area Folder

WIN-OS/2 Setup

That's about all there is to say about my Desktop setup. But there is one more piece of customization I want to share with you. That is my WIN-OS/2 setup.

I have two distinct WIN-OS/2 environments. For most Windows programs, I want them to run in a common windowed WIN-OS/2 session. This includes programs like Word for Windows, Calendar Creator Plus, my Checkbook program, and my Windows database programs. These programs don't need any sound support, and they run very well in a window.

Most multimedia Windows programs, including many games, reference books, and so on, run best at a VGA display resolution (640 x 480 pixels) using 256 colors. This set of programs also requires sound support. Since the programs' optimum display resolution is different from my Desktop display resolution (1024 x 768 pixels), they must run in a separate full-screen WIN-OS/2 session. This is not much of a problem, because I normally run these programs one at a time anyhow.

To accomplish this, I did several things:

1 I selected the Fast Load option from the WIN-OS/2 Setup object. On my home system, it takes about 20 seconds at startup time to load the shared WIN-OS/2 session. However, this saves me about 30 seconds every time I start a Windows programs. I think this is a reasonable trade-off.

2 I selected a display resolution of 640 x 480 x 256 from the System Object and rebooted the system. I used the Windows Control Panel to install all of the multimedia device drivers that I would need – specifically the audio and video device drivers needed for multimedia programs. I saved the resulting SYSTEM.INI file as SYSTEM.MM.

3 I changed the display resolution back to 1024 x 768 x 256 and rebooted the system. I used the Windows Control Panel to remove all of the multimedia device drivers and saved the resulting SYSTEM.INI file as SYSTEM.OS2.

4 I changed my AUTOEXEC.BAT file to copy either SYSTEM.OS2 or SYSTEM.MM to SYSTEM.INI based on the value of the WINRES environment variable.

5 I built an AUTOVGA.BAT file that sets the WINRES environment variable to VGA and calls AUTOEXEC.BAT. For all multimedia programs, I modified the DOS_AUTOEXEC setting to point to this batch file instead of AUTOEXEC.BAT and made them full-screen WIN-OS/2 sessions.

Some of the techniques I have shown you in this appendix might not be directly useful to you, but they should give you some ideas about the kinds of things you can do with OS/2 Warp if you are creative. You can use REXX scripts as wrappers around DOS and Windows programs, to do things with your programs you might not have thought of before. Also, with the proper use of REXX functions, you can change fixed customizations (such as system sounds, the Desktop background, or even the initial Logo screen) into randomized events.

There are many add-ons available for OS/2 Warp, if you know where to look. Many bulletin board systems have an OS/2 section. If you have access to the Internet, you can look at the FTP site hobbes.nmsu.edu for a wide range of OS/2 programs. For example, one package, called PLOGO100, allows you to replace the OS/2 Warp Logo screen with any bit map you have. Other programs, such as PMVIEW and PMJPEG, allow you to view almost any type of image files under OS/2 and convert between different image file formats. For example, you can convert any GIF or JPEG file you might download from the Internet into an OS/2 Bitmap file that you can use for the Desktop background.

You can also find many packages of Windows wallpaper (which are nothing more than bit map files) in computer stores, on bulletin board systems, and on the Internet. They can all be used by OS/2 Warp as folder or Desktop backgrounds. You can also find many packages of sound effects in .WAV format. These can be used as system sounds. And any Windows icon in a .ICO file can be used as an OS/2 icon — just copy it to the General page of any object.

Several OS/2 screen savers are available. The WipeOut screen saver, by BOCA*Soft*, even allows you to use the popular AfterDark screen savers under OS/2.

There is no end to the things you can do with OS/2 Warp. Live Long and Customize!

Ira H. Schneider is an Advisory Programmer on the Workplace Shell Development Team. He has been instrumental in developing OS/2 Warp, including performance improvements and functional enhancements to the Workplace Shell. He has also worked on the OS/2 Swapper and on OS/2 System Initialization, and he was Lead Architect for OS/2 1.2. Ira has been with IBM since 1969.

Appendix C

System Keyboard Definitions

The table represents the list of valid system keyboard definitions as defined by the IBM Common User Access Guidelines. Individual applications may define other keyboard definitions or redefine the system definition.

System Keyboard Definitions

Key	Definition	Definition with Alt key modifier	Definition with Ctrl key modifier	Definition with Shift key modifier
F1	Displays contextual help.			
F2	In a Help window, displays General help.			In a Help window, displays the Tutorial.

continued

System Keyboard Definitions, continued

Key	Definition	Definition with Alt key modifier	Definition with Ctrl key modifier	Definition with Shift key modifier
F4		Closes window. If it is the last window, then it ends the application.		
F5	Refreshes the contents of the window.	Returns the window size to its restore size, that is, its size when not minimized, maximized, nor hidden.		
F6	Switches focus between window panes of the same window. For example, in a File Manager window.	Switches focus from a window to another associated window. For example, from the LaunchPad to one of its drawers.		
F7		Puts the window in move mode. You can then move the window with the arrow keys or with the mouse.		
F8		Puts the window in size mode. You can then size the window with the arrow keys or with the mouse.		Toggles add mode when in a window enabled for extended selection, such as a Workplace Shell folder in Icon View. When in add mode, the arrow keys move the cursor instead of changing the selection. Use the space bar to toggle the selection state of the cursored item.
F9	In a Help window, displays Keys help.	Minimizes or Hides the window.		

Key	Definition	Definition with Alt key modifier	Definition with Ctrl key modifier	Definition with Shift key modifier
F10	Switches focus to the menu bar.	Maximizes the window.		Displays pop-up menu for the selected objects in the window. If no objects are selected, it displays pop-up menu for the object represented by the window. In a Help window, displays the Using help window.
F11	In a Help window, displays Help index window.			
\			Deselects all objects and items in the window.	
/			Selects all objects and items in the window.	
↓	Moves the cursor down.	Displays the drop down list combo box. Moves focus from notebook tab or from the page arrow buttons back to the notebook page.		Extends selection to the current character position on the line below the current line.
↑	Moves the cursor up.			
←	Moves the cursor left.		Moves to the previous word to the left of the current word.	Extends selection one character or object to the left.
→	Moves the cursor right.		Moves to the next word to the right of the current word.	Extends the selection one character or object to the right.
Backspace	Backspace.	Undoes the last action in the window. This is supported in Workplace Shell Settings Notebooks in addition to applications that have an Undo menu.		

continued

System Keyboard Definitions, continued

Key	Definition	Definition with Alt key modifier	Definition with Ctrl key modifier	Definition with Shift key modifier
Delete	Deletes or clears.			Cuts the selected items, placing them on to the clipboard.
End	Moves cursor to the end of the line.		Moves the cursor to the end of the data.	Extends the selection to the end of the data.
Enter	Causes the default action to be performed. For example, in a Workplace Shell folder the selected objects are opened. In a dialog box, the default push button is activated.	In a Workplace Shell folder, opens the selected object's Settings Notebook.	Carries out default action.	In text, causes a new line. In a Workplace Shell folder, causes the folder to be closed after the selected objects are opened.
Enter on numeric keyboard	Default action. Never performs the new line action.			
Esc	Cancels the current operation or dismisses the current window without taking any action.	Moves focus from current window to next window (including full-screen sessions).	Displays the Window list.	Displays the System menu of the current window.
Home	Moves focus to the beginning of the line.		Moves focus to the beginning of the data.	Extends the selection to the beginning of the data.
Insert	Switches between insert and replace modes In text.	Creates a new object of the same type as the selected object on the clipboard.	Creates a copy of the selected objects on the clipboard.	Pastes the object on the clipboard into the current location of the object with focus.
Page Down	Pages down within the data.	Moves to the next page in the Settings Notebook.	Pages to the right.	Extends the selection by one page full of data downward.

Key	Definition	Definition with Alt key modifier	Definition with Ctrl key modifier	Definition with Shift key modifier
Page Up	Pages up within the data.	Moves to the previous page in the Settings Notebook.	Pages to the left.	Extends the selection by one page full of data upward.
Spacebar	Inserts a space character in text. Pushes or changes the state of a button. Changes the selection state of an object or item.	Displays System menu.	Pushes or changes the state of a button. Changes the selection state of an object or item.	Inserts a space character in text.
Tab	Moves cursor to the next control. In text, inserts a tab character or moves cursor to next tab stop.		Moves cursor to the next control even if cursor is in text.	Moves cursor to the previous control. In text, it moves cursor to the previous tab stop.

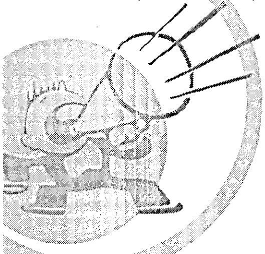
Index

Symbols

& (ampersand), 604, 606–607
* (asterisk), 82, 608
@ (at symbol), 607
\ (backslash), 602
[] (brackets), 84
^ (caret), 606–607
: (colon), 300
, (comma), 300, 303, 608
– (dash), 302
" (double quotes), 323, 339, 607
= (equal sign), 608
/ (forward slash), 302, 602, 608
> (greater-than symbol), 605–606
< (less-than symbol), 606
() (parentheses), 604–605
% (percent sign), 84–85, 607
. (period), 302, 303, 602
? (question mark), 83–84, 116–117, 608
; (semicolon), 302, 339
' (single quote), 302, 323
| (vertical bar), 604, 605

A

Action Button area, 101–103
Add Criteria dialog box, 177–178, 182, 183
Add Font dialog box, 233–234
add mode selection, 90, 92–93
Add New Font dialog box, 233–234
Add Programs object, 396
Add push button, 177, 182, 183
addresses
  Internet, 503–504
  in the PIM Address Book, 489–490
Adjust mark, 411
Advanced Options dialog box, 30–31
Advantis, 502–503, 507
alarm(s)
  in the Alarm Clock, 287
  Personal Information Manager, 492
  in the System Clock, 304, 309–310
Alarm page, 309–310
ALIAS, 561
Alt-F1, 374
Alt-F4, 108
Alt-F6, 108
Alt-F7, 108
ALWAYSSORT=, 340, 346



Ami Pro, 47, 59
ampersand (&), 604, 606–607
animation icons, 136, 174
Animation option, 282, 288
ANSI command, 644
ANSI escape sequences, 630–637
APIs (Application Program Interfaces), 693
APM (Advanced Power Management), 310–314
APPDEFAULT=, 340, 355
Application Consideration, 431
Aptiva computers, 310
Archive and Recover, 473–478
archive attribute, 146, 148, 181–182
Archive page, 193
archives, 473, 475–476
Archives Characteristics property, 187–188
area (marquee) selection, 90–91
ARPAnet, 500–501
Arrow mouse pointer, 261–262, 264
ARROW.PTR, 264
ASCII text editors, 720
ASSOCFILTER=, 340, 350
associations
    basic description of, 47–50
    default, 48–49
    establishing/changing, 49–51
    using Association pages and, 77–79, 547
    using Menu pages and, 79–82
    in the Workplace Shell, 77–82
ASSOCTYPE=, 340, 350
asterisk (*), 82, 608
ATM (Adobe Type Manager) fonts, 24, 296
at symbol (@), 607
ATTRIB, 644–645
AUDIO_ADAPTER_SHARING, 707
Auto-dim function, 191
AUTOEXEC.BAT, 296, 760
    DOS command prompts and, 403, 405
    installation and, 12, 13
    migrating old statements from, into new configuration files, 31
    sound devices and, 296
Auto-rename object check box, 280
Autosave option, 411, 420
AUTOSETUP=, 340, 353, 354
AUTOSTART keyword, 116
Available Allocation Units field, 162
Available Formats list box, 409

B

BACKGROUND=, 340, 346
background color, 219, 223–228, 245–253
    for graphs, 422
    Scheme Palette and, 246–253
    transparent, 237–241
Background page, 186, 191, 242–246
Background property, 172
Background radio button, 307
backslash (\), 602
BACKUP, 646–647
BASEDEV=, 443, 597, 598
Base Multimedia Support, 28
batch files, chained, 609
batteries, 310–314
Battery only radio button, 313
baud rates, 585
Begin drag, assigning a sound for, 287
BEGINLIBPATH, 456–457
bitmaps, 220–223
    on the Productivity Pack CD-ROM, list of, 731–734

Chalkboard feature and, 520–521
optional, installation of, 28
Palette Manager and, 208–209
tiling, 245
transmitting, 506
BLDLEVEL, 647
Bloomfield, Cathy, 598
BonusPak, 339, 479–525, 528, 565
bookmarks, 511
BOOT, 648
Boot time services, 465–478
Boot Manager and, 13–15, 465–473
using the Archive and Recover feature, 473–478
borders, sizing, 253
brackets ([ ]), 84
BREAK=, 444
BUFFERS=, 443
button appearance, 124, 281

C

CACHE, 648–649
CALL, 608–609, 617
CALL=, 444, 462
Call Manager, 519–520
Cancel button, 581
cancelling, pickup, 95
Cancel push button, 183, 278, 402
caret (^), 606–607
cascading windows, 66
CCVIEW=, 340, 344
CD, 609–610
CD Players, 536–538, 558–560
CDPM.EXE, 536–538
CD-ROMs, 1–2, 144, 536–538, 558–560
contents of, list of, 731–744
copying bitmaps from, 245
the Disk object and, 153
drives for, choosing types of, 6
installation and, 6, 7, 10, 547
the Multimedia Viewer and, 548–550
as read-only, 96, 435
REXX files on, 137
the WPDisk class and, 162, 194
cells
horizontal/vertical separation of, 164
number of, 164
width/height of, 164
Centered, 422
chained batch files, 609
Chalkboard, 519–520
Change Criteria dialog box, 177–178, 182, 184
Change push button, 177, 183, 184
charts, 486
CHCP, 610–611
CHDIR, 609–610
Check Disk, 153, 154–157
Check Disk Results Window, 157
Chess, 426–427
CHKDSK, 9, 35, 140, 189, 649–650
the Check Disk dialog box and, 153–157
FAT file system and, 142–143
CIM.INI, 514
CIM-OS/2 (CompuServe Information Manager for OS/2), 480–482, 499, 511–517
Circle, 415–416
class(es). *See also* specific classes
base, basic description of, 139–151
definition of, 120
Class Icon property, 121
Class Style property, 121–122

Class Title property, 121
Clear, 414, 421
clipboard
- copying items to, 400, 410, 414, 421
- data exchange and, 298–299
- marking text/graphics to be copied to, 400

Clipboard Viewer, 408– 409
CLIPOS2, 650–651
clocks
- Alarm Clock, 287
- System Clock, 287, 299, 304–310, 396, 750–751

closing windows, 66, 86
- assigning a sound for, 287
- closing minimized windows, 199

CLS, 611
CLSSTYLE_NEVERTEMPLATE, 197
CMD, 651–652
CMD.EXE, 65, 81, 83, 321–322, 608, 651, 725
CODEPAGE=, 443
colon (:), 300
color. *See also* background color; color palettes
- adding, to the LaunchPad, 115–116
- changing default, 224
- changing foreground, 224–228
- corruption, 207–211
- customizing, 203–254
- in dialog boxes, 216–217
- dithering, 208, 209, 228
- in the edit window, 421
- in folders, 218–219
- in help panels, 215
- Icon Editor and, 413–419
- in menus, 214–215
- of miscellaneous elements, 219
- modes, 206–207
- mouse cursors and, 266–268
- RGB color, 229–230
- for the System Clock, 307–308
- transparent, 237–242, 250, 253
- use of, basic description of, 206–220
- window element, 211–220

Color fill, 419
Color group, 243
color palettes, 223–230, 308
- basic description of, 163
- changing System Clock colors with, 308
- the Workplace Shell and, 56–57

COLORS=, 340, 353
COM_DIRECT_ACCESS, 707
Comet Cursor, 256, 266–268
COMETRUM.EXE, 268
COM_HOLD, 707
comma (,), 300, 303, 608
command line, 474, 599–690. *See also* command prompts
- command operators and, 603–607
- external commands and, 644–690
- file systems and, 600–603
- internal commands and, 608–644

Command menu, 411
Command Processor, 455–456
command prompts, 29. *See also* command line; Command Prompts folder
- DOS, 397, 403–406
- WIN-OS/2, 406–407

Command Prompts folder, 105, 397–407
- DOS command prompts in, 397, 403–406
- OS/2 command prompts in, 397
- OS/2 Window object in, 398–402
- WIN-OS/2 command prompts in, 406–407

Command Reference, 431

Comment property, 146, 182
Common User Access Group, 89
COMP, 652–653
Comparison type list box, 178–179
COM ports, 502, 514–515, 584, 707–708
compressed files, 506
CompuServe, 518, 565
CompuServe Information Manager for OS/2 (CIM-OS/2), 480–482, 499, 511–517
COM_RECEIVE_BUFFER_FLUSH, 708
COM_SELECT, 708
Concurrent View Behavior property, 124
CONFIG.D, 478
CONFIG.SYS, 438, 601, 692–693
  Comet Cursor and, 268
  creating, for different boot environments, 474, 477–478
  disabling the LaunchPad and, 116
  DOS command prompts and, 403
  installation and, 10, 12, 13, 30, 546
  the Menu Items Settings dialog box and, 81
  multimedia setup and, 546
  options that affect the Workplace Shell, 441–463
  printing and, 584, 597
  running programs and, 461–463
  Startup folder and, 196, 427
CONFIG.U, 478
Confirmations pages, 99, 276–279
CONNECTIONS keyword, 438, 450
container object views, 62
Container property, 125
context menus. *See* pop-up menus
Control Drag and Drop Operations, 164–165
Control Panel
  cursor blink rate settings in, 270
  installing fonts and, 24
  mouse settings in, 258
Copy, 410, 414, 421
COPY, 84, 159, 584, 611–615
Copy dialog box, 131–132
copying
  diskettes, 98, 153–154, 157–159, 654–655
  graphics, 400
  highlighted text, 410
  items to the clipboard, 400, 410, 414, 421
  marked text to the cursor, 410
  print jobs, 574
  shadows, 278
  text, 400, 410
Corel, Doris, 524
COUNTRY=, 443
country information, 18–22, 299–304, 396, 443, 666–668
Country object, 299–304, 396
Country page, 300
Create Another dialog box, 129–130, 132, 133
Create a Partition dialog box, 468–469
Create Shadow dialog box, 133–134
Creative Labs, 560
Criteria System Group, 183
Ctrl key, as a drag modifier key, 94
Ctrl-Alt-Del, 32, 39, 189, 192, 442, 451
Ctrl-Alt-NumLock, 189
Ctrl-Break, 158–159, 561
Ctrl-C, 158–159
Ctrl-Esc, 102, 404, 406, 407, 449
Ctrl-Shift, 94, 97
CUA (Common User Access) group, 42–43, 63, 763
Currency format, 303–304

Current Disk Usage area, 157
cursor
- blink rates, timing, 268–270
- comet, 256, 266–268
- copying/moving marked text to, 410

Cut, 414, 421

D

dash (–), 302
data
- compression, 506
- file objects, 49–51, 168–171

databases, 487–488
Data Exchange page, 297–299
Data type property, 146
DATE, 615
Date page, 301–302
Date radio button, 308
date/time. *See also* alarms; clocks; time; timing
- in the Country objects, 299–304
- creation, 145, 181
- last changed/last accessed, 145, 181
- last write, 74, 180–181
- in the Personal Information Manager, 488–495
- print job, 575–576

Date/Time page, 304–305
DBLSPIN directory, 556
DDP (Device Driver Profile) files, 34
Default Icon Position property, 122
Default push button, 137, 139, 264
DEFAULTSORT=, 340, 346
DEFAULTVIEW=, 340, 344, 346, 355
Default View property, 125, 172
Defense Department, 500–501
DEFRAG, 17
DEL, 615–616
Delay until repeat, 273
deleting
- confirmation before, 276–277, 279
- drawer objects, 115
- files, in the file system, 615–616
- folders, 276
- fonts, 234
- LaunchPad objects, 115
- marked text, 411
- Minimized Window Viewer objects, 149
- objects, 97, 115, 126, 128, 134, 149, 166, 276, 581
- printer driver objects, 581
- print jobs, 574
- search criterion, 178

Delete Objects dialog box, 134
Delphi, 502
Desktop
- adding programs to your, 35, 743–762
- archives, 473, 475–477
- Developer's Desktop, 743–762
- directory, archiving, 476–477
- folder, displaying a Tree view starting at, 130
- installation and, 27–28, 35
- Maintenance Desktop, 473, 475
- minimizing windows to, 138
- saving states of, 188, 193

Desktop page, 193–194
DETACH, 461, 617
DETAILSCLASS=, 340, 346
DETAILSFONT=, 340, 346
Details page, 159, 161–163
DETAILSTODISPLAY=, 340, 346
DETAILSVIEW=, 340, 346

Details view, 62, 70, 72–75, 81, 173, 185, 245
Details view page, 240–242
Developer's Desktop, 743–762
Developer's Toolkit, 25, 94–96
device(s)
  names for, 543–544, 601–602
  problems with, troubleshooting, 37–39
DEVICE=, 443, 444
Device Driver Install object, 396
DEVICEHIGH=, 444
Device List dialog box, 418
Device menu, 418
Device Selections and Settings dialog box, 21–22
DEVINFO=, 443
diagnostic aids, installation of, 28
dialog boxes
  creating, 129–130
  what you can color in, 216–217
Digital Auto editor, 538–540
Digital radio button, 306
Digital Video, 545–547
directories. *See also* Shared Directory folder object
  archives and, 476–477
  that contain help, specifying the list of, 458–459
  the DIR command and, 159, 617–620
  file systems and, 600–603
  root directories, 438–439
  shared, drives assigned to, 436, 437
Directory Flag property, 182
disabling
  the Animation option, 282
  the Comet Cursor, 267
  keyboard editing keys, 625–627
  the LaunchPad, 116
  the Print Screen key, 295
  the print spooler, 292–293
  system sound, 535
  the Workplace shell palette aware check box, 275–276
DISKCOMP, 653
DISKCOPY, 157–161, 653–655
diskettes. *See also* CHKDSK; Disk object
  checking, 99, 153–157
  copying, 98, 153–154, 157–161, 654–655
  fixed, 160
  formatting, 99, 661–663
  managing, 98–99
  utilities, creating, 35–36, 396
  write protecting, 158
Disk object, 151–152, 153–165, 348
display adapters, specifying the resolution of, 274–276
dithering, 208, 209, 228
Digital Audio, 546–547
DLLs (Dynamic Link Libraries), 47, 48, 446, 456–457, 462
documentation, installation of, 24
DOS (Disk Operating System), 397, 403–406, 691–730
  advanced options and, 30–31
  associations and, 77, 79
  dual boot support and, 17
  FAT file system and, 142–144, 693
  formatting diskettes and, 160–161
  HPFS file system and, 17, 144
  installation and, 15, 17, 18, 26
  keyboard settings, 696–698
  memory settings, 698–703
  mouse and touch screen settings, 702–703
  partitions, creating, 468–469, 473
  printing and, 568, 578, 703–704, 720–725
  Protected Mode Interface (DPMI), 693, 699, 719

session settings, 86, 694–695
settings, modifying all, 717–718
settings, printing/loading, 720–725
sound devices and, 296–297
support, overview of, 692–693
the System Information Tool and, 522–523
templates and, 45
video settings, 704–706
DOS_AUTOEXEC, 691, 709
DOS_BACKGROUND_EXECUTION, 709–710
DOS_BREAK, 710
DOS_DEVICE, 710–711, 722
DOS_FCBS, 711
DOS_FCBS_KEEP, 711–712
DOS_FILES, 712, 691
DOS_HIGH, 698
DOSKRNL, 692–693, 719
DOS_LASTDRIVE, 712
DOS_RMSIZE, 698
DOS_SHELL, 712–713
DOS_STARTUP_DRIVE, 713
DOS_UMB, 699
DOS_VERSION, 713
double-click settings, 256–257
double quotes ("), 323, 339
DPATH, 621
DPMI (DOS Protected Mode Interface), 693, 699, 719
DPMI_DOS_API, 699
DPMI_MEMORY_LIMIT, 699
DPMI_NETWORK_BUFF_SIZE, 699
dragging, 82, 83
concepts, understanding, 93–94
drag modifier keys and, 94
LaunchPad and, 103–104
mouse settings and, 260
object management and, 93–94
pickup/drop operation and, 94–96, 128–129, 199
standard objects and, 58, 59
templates, 76
the WPObject class and, 126
the WPPalette class and, 164–165
Drawer Access Button area, 102, 105–108
DRAWEROBJECTS=, 340, 348
drawers
adding objects to, 114–115
breakaway, 107–108
closing, 108
deleting objects from, 115
displaying text in, 110
drive(s). *See also* partitions
assigned to shared directories, 436, 437
changing print spool paths to different, 291
compressed, installation on, 9–10
displaying available, 130
the Drives folder and, 70, 104, 407
formatting, 29
Drive Copy Window, 157–161
DRIVENUM=, 340, 348
drivers
port, 568, 584–585
printer, 439, 568–571, 575, 580–581
queue, 568, 576–577, 586–587
Drives folder, 70, 104, 407
DSOM (Distributed System Object Model) environment, 459–461
Dynamic Data Exchange (DDE), 298

E

EAUTIL, 655–657
ECHO, 621

Edit Color dialog box, 230, 236–237, 242, 249, 252, 416
Edit Font dialog box, 232–234, 236, 242
Edit menu, 410–411, 414, 421
editor(s)
ASCII text editors, 720
Digital Auto editor, 538–540
Enhanced Editor, 409–411, 420
Icon Editor, 135–136, 170, 244, 412–419
MLE (multiple line editor), 722, 727
System Editor, 47–48, 77–79, 82, 420–421, 431
Edit Predefined Device Forms dialog box, 418
Edit Scheme dialog box, 248–249
EGA display, 223
Eject Disk, 153, 154
e-mail, 501, 505–506
EMS (Expanded Memory Specification), 699–700, 719
EMS_FRAME_LOCATION, 699
EMS_HIGH_OS_MAP_REGION, 700
EMS_LOW OS_MAP_REGION, 700
EMS_MEMORY_LIMIT, 700
End drag, assigning a sound for, 287
ENDLIBPATH, 456–457
ENDLOCAL, 621
Enhanced Editor, 409–411, 420
environmental variable substitution, 607
Epson printers, 583
equal sign (=), 608
ERASE, 615–616
error(s)
CIM-OS/2 and, 512
display of, assigning a sound for, 287
fatal, during installation, 36
MCI command, 562
off-line printer, 584
escape sequences, 630–637
Event Monitor, 495
execution
multiple threads of, 529
silent, 607
EXENAME=, 340, 350
exiting
closing windows on, 86
the command prompts window, 398
with the EXIT command, 622
saving settings when, 188, 193, 417
EXPLORE.INI, 508–509
extended attributes (EAs), 141, 142–143, 145, 149, 169, 180, 603
EXTPROC, 622

F

Fast load option, 295
FAT (File Allocation Table) File System, 29, 147
basic description of, 142–144, 600–603
DOS settings and, 693
formatting diskettes and, 159–161
installation and, 12, 16–17
the Shared Directory folder and, 435
FaxWorks for OS/2, 480–482, 495–499
FCBS=, 444
FDISK, 14, 16, 466, 657–659
FDISKPM, 466, 468–469, 472–473, 659–660
FILE=, 557
file access
basic description of, 600–603
functions, 368–391

file extensions, 47, 77–78, 279
- BAT, 644
- BMP, 170
- COM, 644
- CMD, 644
- DOC, 176
- DRV, 171
- EXE, 37, 644
- ICO, 170
- PTR, 263
- TXT, 84

File Manager, 42, 53, 89, 98–99
File menu, 409–410, 420
filenames
- physical, of objects, 74
- qualified/fully qualified, 602
- rules that govern, 600–602
- UNC (Universal Naming Convention), 436–437

File page, 146–149
FILES=, 444
File Server folder object, 196, 352
file system objects, 65, 141–149
file systems. *See* FAT (File Allocation Table) File System; HPFS (High Performance File System)
FILE.TXT, 85, 602
file types, 47, 48, 77
Fill, 415, 422
FIND, 660–661
Find action, 108
Find Criteria dialog box, 177–178
Find dialog box, 421
Find Objects dialog box, 175–179, 244
Find utility, 102
flags, 75, 145–146, 181–182
Flip Horizontal, 415
Flip Vertical, 415
Flowed Icon view, 72, 235–236
folder(s)
- allowing the Workplace Shell to open, 449–450
- basic description of, 52–56, 69–76
- color in, 218–219
- deleting, 276
- displaying open, 130
- folder objects and, 172–197

Folder Automatic Close, 186–187, 283–284
Folder Background page, 191
Folder Settings Notebook, 235–246
FOLDERS keyword, 449–450
Folder view, 90
font(s), 203–254. *See also* font palettes
- adding new, 233–234
- ATM (Adobe Type Manager), 24, 296
- basic description of, 220–223
- creating custom, steps for, 232–233
- installation and, 24
- ISO (International Standards Organization), 221, 222, 232
- in the LaunchPad, 115–116
- removing old, 234
- size, setting, 399, 401–402
- for the System Clock, 308
- window elements that support, 221–223

font palettes, 223, 230–234, 240, 242, 396
- basic description of, 163
- the Workplace Shell and, 56–57

FONTS=, 340, 354
FOR, 607, 622–623
FORMAT, 661–663
Format Disk, 153, 154, 159–161
Format Disk dialog box, 159–161
formatting disks, 99, 153, 154, 159–161, 661–663

forward slash (/), 502,602,608
FPOBJECTS=, 340, 349
Frame controls, 411
Freeze screen, 422
FROM=, 558
Full screen push button, 191
Full status radio button, 313

G

Galactic Civilizations, 564
games, 564–566
  the Games folder and, 44, 425–427
  installing, 26, 407, 425
General page, 412
Gerstner, Lou, 507
GIF (Graphic Interchange Format), 506, 512
global environment, setting, 455–459
Gopher, 501, 504, 511
GOTO, 623–624
graphs, the Pulse display and, 422
greater-than symbol (>), 605–606
green question marks, 116–117
Grid, 417
GROUP=, 340, 352
GUI (graphical user interface), basic description of, 46

H

Halt command, 411
handshake, 585
hard disk space, 23, 466–468, 481
hardware
  configuration, 18–22
  requirements for OS/2, 5–6
  support, recommended, 6
Hare, John M., 729
height/width
  of borders, 253
  of cells, 164
help, 57, 411
  accessing, with the HELP command, 663–664
  directories that contain, specifying the list of, 458–459
  panels, what you can color in, 215
  for selected windows, 66
  for settings categories, 695, 696
  for Workplace Shell objects, 57
  the WPObject class and, 127–128
  the WPMinWindow class and, 199
HELP command, 663–664
Help Information property, 124
HELPLIBRARY=, 340, 344
HELPPANEL=, 340
hidden attribute, 145–146, 148, 181–182
HIDEBUTTON=, 341, 344
Hide buttons, 281
hiding. *See also* hidden attribute
  frame controls, 109
  the LaunchPad icon, 201
  programs, 46–47
  push buttons, 138
  text, 250
  windows, 66, 67, 86, 124, 138
hierarchy, class, 120
History property, 146, 149, 182
HMA (High Memory Area), 719
hotspots, 265, 418
Hour and minute hands radio button, 308
Hour marks radio button, 307

HPFS (High Performance File System), 74, 235, 693
- basic description of, 144, 600–603
- formatting diskettes and, 160–162
- installation and, 16–17, 28, 29

HW_NOSOUND, 714
HW_ROM_TO_RAM, 714
HW_TIMER, 714
HyperACCESS Lite, 480–482, 499, 518–519

I

IBM (International Business Machines), 572, 583. *See also* IBM Works
- Aptiva computers, 310
- Internet Connection, 480–482, 499–518
- Internet Customer Services, 501, 504
- Internet Dialer, 501–502, 510
- Information Superhighway, 480–482, 499, 500–518
- PS/1 computers, 310

IBM Works, 480–495
- database and report writer, 487–488
- spreadsheet, 485–486
- word processor, 483–485

icon(s)
- animation, 136, 174
- on the Productivity Pack CD-ROM, 731, 733–735
- editing, with the Icon Editor, 135–136, 170, 244, 412–419
- modifying, 135–136
- positioning, 122, 172

ICONEDIT, 412, 664–666
ICONFILE=, 341, 344
ICONFONT=, 341, 346
ICONNFILE=, 341, 347
ICONNRESOURCE=, 341, 347
ICONPOS=, 341, 344
Icon Positions property, 172
Icon property, 122, 198
ICONRESOURCE=, 341, 344
Icon text group, 236
Icon view, 62, 70–72, 172, 174, 185
ICONVIEW=, 341, 347
Icon View page, 236–237
ICONVIEWPOS=, 341, 347
IDLE_SECONDS, 714
IDLE_SENSITIVITY, 714–715
IFS=, 443
IF statements, 607, 624–625
Illegal mouse pointers, 262, 264
ILLEGAL.PTR, 264
Image group, 243
Include page, 183, 201
Indefinite radio button, 286
Independent Color Form, 418
index, Master Help, 428–430, 458–459
Information folder, 54, 104, 321
- books in, 431
- objects in, 428–431

information messages, 17, 287
Input page, 284–285
installation, 5–40
- adding programs to your Desktop after, 35
- Advanced Installation, 13–17, 35
- BonusPak, 481–482
- Boot Manager, 13–15, 467–468
- on compressed drives, 9–10
- configuring software and, 30
- country information and, 18–22
- creating partitions and, 15–16

creating utility diskettes and, 35–36
of device drivers, 34
of diagnostic aids, 28
of documentation, 24
of DOS, on drive C:, 15
dual boot support and, 17
Easy Installation, 11–13, 31
of fonts, 24
formatting partitions and, 16–17, 29
of games, 26, 407, 425
hardware configuration and, 18–22
hardware requirements for, 5–6
HPFS, 28
information messages and, 17
on large partitions, 8–9
media, choosing the right, 7–8
multimedia, 19, 21–22, 28
of optional bitmaps, 28
of optional system utilities, 25
over the network, 32
phase 1 of, 11–18
phase 2 of, 18–31, 407, 425
preparing for, 8–10
of printers, 20–21, 568–571, 582
Selective Install, 33, 407, 425, 571–572
serviceability and, 28
of tools, 26
troubleshooting, 36–39
uninstalling features, 33–34
using advanced options, 30–31
using third-party devices, 10
of Windows, on drive C:, 15

Install New Printer Driver dialog box, 568–571
instance data, saving/restoring, 127
INT_DURING_IO, 715
interaction paradigms, standard, 63–67
Internet
browsing for games on, 564
history of, 500–501
IBM Information Superhighway and, 480–482, 499, 500–518
IBM Internet Connection and, 480–482, 499–518
IBM Internet Dialer and, 501–502, 510
WebExplorer and, 501, 504, 505–507, 528

Invisible icons, 237
IOPL=, 443
ISO (International Standards Organization) fonts, 221, 222, 232

J

JOBDIALOGBEFOREPRINT=, 341, 355, 587
Job object, 198
JOBPROPERTIES=, 341, 355, 582
Jones, Ron, 539–540
JPEG files, 512

K

KBD_ALTHOME_BYPASS, 696–697
KBD_BUFFER_EXTEND, 697
KBD_CTRL_BYPASS, 697
KBD_RATE_LOCK, 697
KEYB, 666–668

keyboard. *See also* Keyboard objects; keystroke combinations
- ANSI escape sequences and, 630–637
- access techniques for the LaunchPad, 108
- country/region information used to interpret, 666–668
- definitions, system, 763–768
- editing keys, enabling/disabling, 625–627
- selection techniques, 90–93
- settings, DOS, 696–698

Keyboard object, 151–152, 269–273, 396
- input, saving, 284–285
- mappings, 269, 271–272
- special needs and, 269, 272–273
- timing, 269–270, 272–273

Key phrases property, 146, 182

KEYS, 625–627

keystroke combinations
- Alt-F1, 374
- Alt-F4, 108
- Alt-F6, 108
- Alt-F7, 108
- Ctrl-Alt-Del, 32, 39, 189, 192, 442, 451
- Ctrl-Alt-NumLock, 189
- Ctrl-Break, 158–159, 561
- Ctrl-C, 158–159
- Ctrl-Esc, 102, 404, 406, 407, 449
- Ctrl-Shift, 94, 97
- Shift-F9, 272
- Shift-F10, 95, 271

Killian, David, 539–540

Kodak Photo-CDs, 528, 531, 548

Kulkarni, Salil, 39

L

LABEL, 668

LANs (local area networks), 53, 141
- BonusPak and, 482
- file systems and, 144
- Network folder and, 431–437
- Person to Person and, 519

LaserJet printers, 583

laser printers, 589, 597

LASTDRIVE=, 444

LaunchPad, 56, 70, 81, 101–118, 744–745
- adding objects to, 113–114
- copying objects and, 131–132
- customizing, 108–115
- disabling, 116
- dropping colors/fonts on, 115–116
- fast and easy access to, 102
- green question marks on, 116–117
- icon, hiding, 201
- options, setting, 108–113
- removing objects from, 113, 115
- shadow objects and, 97, 439
- views and, 60

LAUNCHPAD keyword, 450

LaunchPad object, 151–152, 348–349

Legato, 484

less-than symbol (<), 606

LIBPATH=, 443, 456–457

libraries, dynamic-link (DLLs), 47, 48, 446, 456–457, 462

Link Object Modules utility, 25

lists-of-lists, 719

Load default palette, 416

Local Printer object, 152, 355–356

Lock Disk, 153, 155
Lockup, 102, 108, 187–188
assigning a sound for, 287
pages, 190–193
Lockup Characteristics property, 187–188
logical device names, 543–544
Logo page, 285–286
logos, 243–244, 285–286
Logout dialog box, 434
Lotus 1-2-3, 47, 50, 59
LPACTIONSTYLE=, 341, 349
LPCLOSEDRAWER=, 341, 349
LPDRAWERTEXT=, 341, 349
LPFLOAT=, 341, 349
LPHIDECTLS=, 341, 349
LPMALLICONS=, 349
LPTEXT=, 341, 349
LPVERTICAL=, 341, 349
LucasArts Entertainment, 564

M

Magid, Peter G., 87, 99, 117, 478, 689
Mahjongg Solitaire, 426
Maintenance Desktop, 32, 473, 475
Maintenance Partition, 18, 32
MAKEINI, 668–669
Malone, Bill, 555
mapping
keyboard actions to functions, 269, 271–272
mouse actions to functions, 256, 259–261
Mappings page, 259–261
marquee (area) selection, 90–91
Master Help Index, 428–430, 458–459
Master Volume Control, 289
maximized windows, 199, 341, 350, 399
MAXIMIZED=, 341, 350
MD, 627
measurement systems, 300
Media Player, 542–547, 557, 729
MEM_EXCLUDE_REGIONS, 700
MEM_INCLUDE_REGIONS, 701
MEMMAN=, 443
memory. *See also* hard disk space
Boot Manager and, 466
BonusPak and, 481–482
FAT file system and, 142–144
formatting diskettes and, 160–161
installation and, 5–6, 16–17
management, DOS, 26, 698–703
RAM (random-access memory), 5–6, 16–17, 692
specification, expanded (EMS), 699–700, 719
specification, extended (XMS), 701, 719
menu(s). *See also* menus (listed by name); menus, pop-up
choice help, 57
the Menu Items Settings dialog box and, 80–81
Menu pages and, 65, 79–82
what you can color in, 214–215
menus (listed by name)
Command menu, 411
Device menu, 418
Edit menu, 410–411, 414, 421
File menu, 409–410, 420
Options menu, 417, 422, 472–473
Original/Locate menu, 105
Palette menu, 416
Search menu, 411
Tools menu, 419
menus, pop-up
associations and, 80
basic description of, 44–45
displaying, 70, 125, 260, 271

menus, pop-up *(continued)*
- LaunchPad and, 103
- mapping functions for, 271
- mouse settings for, 260
- object, 67–69
- Workplace Shell, 58–59

messages
- boxes, displayed at alarm times, 310
- displaying previously displayed, 411

Messy Desk view, 70–71
MIDI (Musical Instrument Digital Interface) files, 527, 542, 545, 546, 728
Minimize button behavior, 281–282
MINIMIZED=, 341, 350
Minimized Window Behavior property, 124
Minimized Window object, 198–200
Minimized Window Viewer, 149–150, 428
Minimized Window Viewer folder object, 194–195
minimizing windows, 66, 67, 124, 138, 198–200, 281–282
MINSTALL.EXE, 19, 533
MINWIN=, 341, 344
MIT Media Lab, 555
Mixed Color Palette object, 396
MKDIR, 627
MLE (multiple line editor), 722, 727
MMPM/2 (Multimedia Presentation Manager/2), 527–528, 533, 543
MMPM.INI, 393
MODE, 402, 669–671
modem(s), 53, 514
- IBM Internet Connection and, 501–505
- Person to Person and, 519
- the PIM Address Book and, 489
- protocols, 518

Modifying Partitions Warning dialog box, 14
monetary values, selecting, 303–304
MORE, 672
mouse. *See also* dragging; Mouse object
- Comet Cursor and, 256, 266–268
- DOS settings for, 702–703
- input, saving, 284–285
- left-handed vs. right-handed, 256, 258–259
- mapping, 256, 259–261
- pointers, 256, 261–264
- selection techniques, 90–93
- timing of (click/tracking speed), 256–257

MOUSE_EXCLUSIVE_ACCESS, 702
Mouse object, 151–152, 256–268, 396
MOVE, 627
Move To Another Folder, 125
Move dialog box, 132–133
movies, digital, 28, 556
moving. *See also* dragging
- marked text to the cursor, 410
- objects, 96–97, 128–129, 132–133

MPEG (Moving Picture Experts Group), 530–531
MPPM.EXE, 545
multimedia, 525–566. *See also* sound
- brief history of, 527–528
- data, challenge of delivering, 525–530
- hardware requirements for, 5
- information about, 431, 565–566
- installation and, 19, 21–22, 28, 533, 546–547
- logical device names and, 543–544
- the Media Player and, 542–547, 557
- MIDI files and, 542, 527, 545, 546, 728
- MMPM/2 and, 527–528, 533, 543
- the Multimedia Data Converter and, 532, 533

the Multimedia Device Support icon and, 21–22
the Multimedia dialog box and, 22
the Multimedia Setup object and, 546–547
Multimedia Software Support and, 12
the Multimedia Viewer and, 480–482, 521, 528, 532–550
on-line information about, 556–557
settings for system sound, 535–537
Sound objects and, 287–290
MVDM (Multiple Virtual DOS Machines), 692, 718

N

Naisbitt, John, 500
names
device, 543–544, 601–602
file, qualified/fully qualified, 602
file, rules that govern, 600–602
object, 98, 280
path, 334, 600, 602
Names page, 547
Negroponte, Nicholas, 555
NETGCOMMENT=, 341, 350
NETID=, 341, 356
Network folder, 431–437
Network Group folder object, 195, 431–434
Network Printer object, 572–574
Network Shadow object, 438
Network Status page, 593–594
New, 409, 413, 420
New Name dialog box, 470
New Figure dialog box, 413
New Printer Driver dialog box, 570
news groups, 509–511
NewsReader/2, 501, 504, 505–506, 509–511
Next, 414
NOAUTOCLOSE=, 341, 350
NOCOPY=, 165–166, 342, 345
NODELETE=, 165–166, 342, 345
NODRAG=, 342, 345
NODROP=, 342, 345
NOLINK=, 165–166, 342, 345
NOMOVE=, 165–166, 342, 345
Non-flowed Icon view, 71–72, 235–236
Non-gridded Icon view, 71, 236
None radio button, 286
NOPRINT=, 342, 345
NORENAME=, 342, 345
Normal, 244
NOSETTINGS=, 342, 345
NOSHADOW=, 165–166, 342, 345
NOSOUND.INI, 296
Notepad (PIM), 42, 493–494
NOTVISIBLE=, 342, 345
Nowhere folder, 438–439
Numbers page, 302–304

O

object(s)
adding/removing, 113–115
creating, 97–98, 126, 128
copying, 96–97, 125–126, 128–129, 131–132, 166
deleting, 97, 115, 126, 128, 134, 149, 166, 276
direct manipulation of, 59
Identifiers, 124, 190, 334–339

object(s) *(continued)*
- management, overview of, 89–100
- moving, 96–97, 128–129, 132–133
- original, 165
- physical filenames of, 74
- sorting, 174
- standard, basic description of, 53–67
- starting programs with, 99
- types, basic description of, 119–202
- views of, 59–62
- in the Workplace Shell, basic description of, 67–69

Object Button area, 102, 103–105
Object Class property, 179
OBJECTID=, 342, 345
Object Launcher, 319
Object List, 317, 319
Object open behavior, 282
Object Style property, 179
Object Title property, 179
Object Tool, 731, 738
Off radio button, 313
OK push button, 425, 433, 437
OMNI, 583
OPEN=, 342, 345, 347
opening windows, assigning a sound for, 287
operators, command, 603–607
Options menu, 417, 422, 472–473
Original/Locate menu, 105
Original Window property, 198
OS2.INI, 56, 140, 141, 508, 669
OS2KRNL, 719
OUTPUTOFILE=, 342, 355, 586
Output page, 583–584
overlapping windows, 63
Overlay mark, 410

P

PaintJet printers, 583
Palette Manager, 207–211, 245, 275
Palette menu, 416
Palette object, 152, 163–165, 352–353
palettes, basic description of, 163–164. *See also* color palettes; Scheme Palette
Palette View, 164
paradigms, 63–67
parameters
- customizing OS/2 Windows and, 402
- the Media Player and, 544
- printing and, 578
- Program object, 82–86
- REXX utility functions and, 322–323

PARAMETERS=, 342, 350
parentheses (( )), 604–605
parity checking, 585
Partial screen push button, 191
partitions, 15–18, 98, 194, 467–469
- Boot Manager, 467–468
- C: drive, 472–473
- DOS, 468–469, 473
- installation on large, 8–9
- Maintenance Partition, 18, 32

PAS.EXE, 463
passwords, 192, 432–433, 513
pasting, items to the clipboard, 400, 410, 414, 421
PATCH, 672
path(s)
- environment variable, 607, 628, 644
- names, 334, 600, 602
- specifying, 130

PAUSE, 629
PCMCIA, 22, 315–319
PDDs (Physical Device Drivers), 693, 719, 722
Pen size, 417
percent sign (%), 84–85, 607
perfect save, philosophy of, 52
Performance Considerations, 431
period (.), 302, 303, 602
Person to Person, 480–482, 519–521
Phone/Address Book (Personal Information Manager), 489–490
Phone Book (FaxWorks), 496–498
Photo-CDs, 528, 531, 548
pickup and drop, 94–96, 128–129, 199
Picture Viewer, 421–422
PICVIEW, 578–579
PIF (Picture Interchange Format) files, 421
PIM (Personal Information Manager), 482–483, 488–495
  Appointment Book, 490–493
  Event Monitor, 495
  Notepad, 493–494
  Phone/Address Book, 489–490
  To-Do List Launcher, 494
  Year Calendar, 494–495
piping, 605–606
PKZIP, 506
Planner (PIM), 493–494
PLAY.CMD, 541, 557–560
plotter devices, 583, 585, 589, 592–593
Plug and Play for PCMCIA objects, 315–319
PMCHKDSK, 672–673
PMFORMAT, 673
PMPLOT, 586–587, 592
PMPRINT, 586–587, 592, 594
PMREXX, 674
PMSHELL=, 447
PMSHELL.EXE, 445, 448
PMSPOOL.EXE, 523
pointers, 54, 731, 736–737. *See also* Pointer Sets; shadows
Pointer Sets, 263–266
  building, 265–266
  hotspots for, 265
  using existing, 264–265
points, measuring fonts in, 221
pop-up menus
  associations and, 80
  basic description of, 44–45
  displaying, 70, 125, 260, 271
  LaunchPad and, 103
  mapping functions for, 271
  mouse settings for, 260
  object, 67–69
  Workplace Shell, 58–59
port(s)
  COM ports, 502, 514–515, 584
  port drivers, 568, 584–585
Portele, Andreas, 565
PORTNAME=, 342, 355, 583
PostScript printers, 583, 589
Power objects, 152, 310–314
Power page, 311–312
preemptive multitasking, 529
Preferences, 411, 417
Prefix radio button, 303
Presentation Manager, 63, 66, 89, 219, 284–285
  CIM-OS/2 and, 511
  CONFIG.SYS options statements and, 445, 446–447
  printing and, 594
  running programs and, 461
Preview window, 237, 240, 242–243
PRINT, 578, 584, 674–675
PRINT01.SYS, 597
PRINT02.SYS, 598
PRINTDRIVER=, 342, 355, 580

printer(s). *See also* printer drivers; printer ports
- data flow, 594
- default, 104, 573
- DOS settings for, 703–704
- installation of, 20–21, 568–571, 582
- pooling, 595–596
- sharing, 438, 586, 595–596
- types of, supported by OS/2 Warp, 583

printer driver(s), 342, 355, 439, 575
- basic description of, 568
- default, 568
- installation of, 568–571
- page, 580–581
- what to do when you can't find the correct, 571

Printer object, 56, 567–598
- creating, 568–572
- Settings Notebook, 573, 579–593
- using, 572–579

printer port(s), 568–569, 573–574
- the Printer Port object and, 200
- redirection and, 585–586, 596
- settings, 583–585
- shared access to, 586, 595–596

Printer Selections dialog box, 22

PRINTERSPECIFICFORMAT=, 342, 356, 587

printing
- current edit window, 410
- data file objects, 50
- different ways of, basic description of, 577–578
- DOS and WIN-OS/2 settings for, 720–725
- errors, assigning a sound for, 287
- FaxWorks and, 496–497
- hardware conflicts and, 584
- information, in the Information folder, 431
- jobs, changing the status of, 572
- marked text, 411
- multiple copies, 574–575
- network printer options and, 592–593
- Nowhere folder objects and, 439
- objects, steps for, 97, 126
- out-of-paper messages and, 584
- polling support and, 597–598
- print job options and, 574–577
- print job types and, 577, 578–579
- Printing options page and, 574–575
- redirecting print jobs, 585–586, 596
- subsystem, introduction to, 567–568

Print Manager, 42, 53, 61

Print options page, 587–588

Print Screen key, 577

Print Screen page, 295

PRINT_SEPARATE_OUTPUT, 703

PRINT_TIMEOUT, 703

PRINTWHILESPOOLING=, 342, 356, 587

PRINTXX.SYS drivers, 597

Pro Audio, 406

processors, 5–6

Productivity folder, 407–425

Program Group folder object, 195

Program object, 82–86

Program Manager, 42, 53, 61, 407

Program Reference object, 136, 152, 350–351

Program Settings Notebook page, 82–85

PROGTYPE=, 342, 351

PROMPT, 629–638

properties, definition of, 119–120

Protected Mode Shell, 444–447, 461–462
PROTECTONLY=, 444
PROTSHELL=, 444, 445–447
PS/1 computers, 310
PSCRIPT.SEP, 587
PSTAT, 675–676
public/private data, 298–299
Pulse, 422

Q

QSTARTIME=, 342, 356, 591
QSTOPTIME=, 342, 356, 591
qualified filenames, 602
question mark (?), 83–84, 116–117, 608
QUEUE=, 342, 356
QUEUEDRIVER=, 342, 356, 587
queue drivers, 568, 576–577, 586–587
QUEUENAME=, 343, 356
Queue Options page, 576–577, 586–587, 592–593
QuickList, 507, 508
Quit, 410
quotes, 302, 323, 339, 607

R

RAM (random-access memory), 5–6, 16–17, 692
range click selection, 90, 92
range swipe (ordered) selection, 90, 91–92
range swipe (unordered) selection, 90, 92
Rapid Resume feature, 310
raw data, 579
RD, 638
read-only attribute, 146, 148, 181–182, 435
README files, 8, 35, 430, 485, 731
Real name property, 180
Rebel Assault, 564
REBOOTONLY, 451
RECORD, 559–560
RECORD.CMD, 541, 557–560
RECOVER, 676
Recovery Choices Screen, 192, 477–478
redirection, 585–586, 596, 605–606
Redo, 421
Reel Magic, 528
refresh feature, 174, 275, 313, 343, 357, 569, 592
REFRESHINTERVAL=, 343, 357
Register Object dialog box, 317
REM, 638
Remote Printer object, 152, 438
REMOVEFONTS=, 343, 347
REN, 638
RENAME, 638
Render dialog box, 409
repeat rates, keystroke, 269–270, 272–273
REPLACE, 676–678
reports, 487–488
Reset options, 417
resolution, of displays, 274–276
RESTARTOBJECTS=, 196, 268, 427
RESTORE, 678–679
restoring
    data, 127, 678–679
    windows, 199, 399

REXX, 24, 120, 137, 321–394, 725–727
- BonusPak and, 482
- command files, running, 674
- copying objects and, 131–132
- the Developer's Desktop and, 746–748
- file access functions, 368–391
- information, in the Information folder, 431
- miscellaneous functions, 391–392
- multimedia and, 527–528, 533, 557–563
- printing and, 580–581, 583, 586, 587, 591
- recreating the System Setup folder and, 190
- the REXX command and, 609
- scripts, on the Productivity Pack CD-ROM, 731, 737–741
- specifying setup strings and, 339–347
- the Startup folder and, 196, 427–428
- the System Editor and, 420
- user interface functions, 325–334
- utility functions, 321–325
- Workplace Shell functions, 334–368

REXXTRY, 680
RGB color, 229–230
RMDIR, 638
RMSIZE=, 444
Rogers, Paul, 566
root directories, 438–439
Root folder object, 195–196
RUN=, 444, 461–463
RUNWORKPLACE=, 446–447
RxMessageBox, 323, 325–327

S

SAMPLE.SEP, 587
Save, 410, 414, 416, 420
Save Settings Option property, 188
saving
- closing at the time of, 410
- current edit window into a new file, 410
- current files, 410, 420
- data, the WPObject class and, 127
- desktop states, 188, 193
- edited images, 414
- Input page settings for, 284–285
- options that you have selected, 411
- prompts for, 417
- settings on exit, 188, 193, 417

scalable fonts, 220–223
SCHEME, 246
SCHEME.CMD, 246
Scheme Palette, 211, 222–223
- basic description of, 164, 246–253
- object, 396
- window areas in, 250–253
- the Workplace Shell and, 56–57

Schneider, Ira H., 201, 254, 319, 394, 440, 463, 761
screen
- refresh feature, 174, 275, 313, 343, 357, 569, 592
- resolution, 274–276
- savers, 761

Screen pages, 210, 274–276
scrolling, 399–400
SCSI adapters, problems with, 37–39
search criteria, 177–184
Search menu, 411
Search push button, 424, 429
Seek and Scan Files, 422–425

Select, 415
Select all, 415, 421
selection techniques
  add mode selection, 90, 92–93
  area (marquee) selection, 90–91
  point selection, 90
  range click selection, 90, 92
  range swipe (ordered) selection, 90, 91–92
  range swipe (unordered) selection, 90, 92
Selective Install icon, 33
Selective Install object, 396
Selective Uninstall dialog box, 33–34
Selective Uninstall object, 396
Select menu item, 174
Select System Default Printer dialog box, 20
Self Close Option property, 174
semicolon (;), 302, 339
Separate session check box, 294
SEPARATOR=, 587
SEPARATORFILE=, 343, 356
separator files, 343, 356, 587–591
Serial Support, 22
Server folder object, 434–435
serviceability, 28
SESSION_PRIORITY, 715–716
Session Settings Notebook page, 85–86
SET, 403, 343, 351, 639
SET=, 443
SET AUTOREFRESHFOLDERS=, 452–453
SET AUTOSTART=, 438, 449–450
SET BOOKSHELF=, 459
SETBOOT, 680
Set Colors dialog box, 421
SET DEFERICONLOAD=, 454
SET DESKTOP=, 448–449
SET DPATH=, 457–458
Set Font Size dialog box, 399, 401–402
SET GLOSSARY=, 459
SET HELP=, 458–459
SETLOCAL, 639–640
SET MENUSTYLE=, 454–455
SET OBJECTSNOOZETIME=, 453–454
SET OS2_SHELL=, 455–456
SET PATH=, 457–458
Set pen shape, 418
Set push button, 470
SET RESTARTOBJECTS=, 450–452
SET RUNWORKPLACE=, 446–447
SET SHELL-, 403
SET SHELLEXCEPTIONHANDLER=, 452
SET SYSTEM_INI=, 448
Settings dialog box, 584–585
Settings Notebook, 51–53
  Country object, 299
  LaunchPad, 108–113
  Mouse object, 256
  Print Job object, 574–575
  Printer object, 573, 579–593
  System object, 274, 275–276
  views and, 60
  Window Page, 137–139
  Work Area folders and, 55
  WPDataFile, 169–170
  WPDesktop, 190–193
  WPDisk, 161–163
  WPDrives, 194
  WPFileSystem, 146
  WPFolder, 183–184
  WPMinWindow, 200
  WPMinWinViewer, 195
  WPObject, 125, 127, 134–138
  WPRootFolder, 195–196
  WPShadow, 167
  WPShredder, 167
  WPSpecialNeeds, 167

Settings Notebook *(continued)*
- WPStartup, 197
- WPTemplates, 197
- WPTransient, 150–151

Setup page, 258–259
setup strings, 337–357
- for all folders, 346–348
- for all palette objects, 352–353
- for color palette objects, 353
- for drive objects, 348
- for LaunchPad objects, 348–349
- for network group folders, 350
- for printer objects, 355–356
- for program group folders, 352
- for program reference and program file objects, 350–351
- for scheme palette objects, 354–355
- for server folders, 352
- for shadow objects, 352

SET USER_INI=, 448
SET WPDSERVERSTART=, 460
SHADOWID=, 343, 352
Shadow object, 152, 165–167, 352
shadows, 62, 54, 97, 172, 439
- copying/moving, 278
- creating, 126, 128, 133–134
- on the LaunchPad, 103, 105

SHAREABLE, 561
Shared Directory folder object, 196, 435–438
SHELL=, 444
SHIFT, 640
Shift key, as a drag modifier key, 94
Shift-F9, 272
Shift-F10, 95, 271
SHOWOBJS=, 343, 357
Shredder, 56, 105, 172, 287, 428
Shredder object, 152, 167
shutdown, system, 103, 108, 188–189, 745–746
SHUTDOWN.WAV, 392
Sigma Designs, 528
silent execution, of commands, 607
single quotes ('), 302, 323
SIZENESW.PTR, 262, 264
SIZENS.PTR, 262, 264
SIZENWSE.PTR, 262, 264
Size property, 180
SIZEWE.PTR, 262, 264
slash (/), 302, 602, 608
Small icons option, 110
Smgxpget, 583
Smooth, 422
SMV (software motion video), 28, 527–528, 530, 550–556, 561, 563
Solid Color Palette object, 396
Solitaire, 426
SORT, 681
SORTBYATTR=, 343, 347
SORTCLASS=, 343, 348
sorting, 174, 184–186, 343, 347–348, 681
Sort page, 184–186
sound
- CALL- and, 462
- MIDI files and, 527, 542, 545, 546, 728
- sound cards and, 295–296, 406, 560
- system, settings for, 535–537

Sound Blaster, 406, 560
Sound objects, 152, 287–290, 396
Sound page, 287–290
space character, 601
Special Needs object, 152, 167–168
Special Needs page, 272–273
SPOOL, 681
Spooler object, 153, 291–293, 396, 568
spoolers, 567–68, 578–579, 586, 593, 681
Spool path, 291–293
spreadsheets, 485–486

SRVCOMMENT=, 343, 352
START, 640–642
START COMETRUN.EXE, 449
STARTUP.CMD, 54, 268, 322, 392, 427, 447, 749–751
 CONFIG.SYS statements and, 449, 450, 451, 463
 specifying START COMETRUN.EXE in, 449
STARTUPDIR=, 343, 351
Startup folder, 54, 427–428
Startup folder object, 196–197
Startup Values dialog box, 471–472
STARTUP.WAV, 392
Sticky Keys input method, 272–273
stop bits, 585
Stop push button, 424
Stretch paste, 415
Style property, 123–124, 165
Subject property, 146, 182
Submission date page, 575–576
Suffix radio button, 303
Suppress warnings, 417
Suspend and Resume, 314
SVGA displays, 6, 31
SWAPPATH, 30, 546
SWAPPATH=, 443
SWAPPER.DAT, 546
swapper files, 30, 546
symbols
 & (ampersand), 604, 606–607
 * (asterisk), 82, 608
 @ (at symbol), 607
 \ (backslash), 602
 [] (brackets), 84
 ^ (caret), 606–607
 : (colon), 300
 , (comma), 300, 303, 608
 – (dash), 302
 " (double quotes), 323, 339, 607
 = (equal sign), 608
 / (forward slash), 302, 602, 608
 > (greater-than symbol), 605–606
 < (less-than symbol), 606
 () (parentheses), 604–605
 % (percent sign), 84–85, 607
 . (period), 302, 303, 602
 ? (question mark), 83–84, 116–117, 608
 ; (semicolon), 302, 339
 ' (single quote), 302, 323
 | (vertical bar), 604, 605
SysCls, 323, 327–328
SysCopyObject, 323, 357–358
SysCreateObject, 323, 358–360
SysCreateShadow, 324, 360–361
SysCurPos, 324, 328–329
SysCurState, 324, 329
SysDeregisterObjectClass, 324, 361
SysDestroyObject, 324, 361–362
SysDriveInfo, 324, 368–369
SysDriveMap, 324, 369–370
SysDropFuncs, 324, 391–392
SysFileDelete, 324, 370–371
SysFileSearch, 324, 371–373
SysFileTree, 324, 373–376, 394
SysGetEA, 324, 377–380
SysGetKey, 324, 329–330
SysGetMessage, 324, 330–332
SysIni, 324, 380–383
SYSLEVEL, 681
SYSLOG, 682
SysMkDir, 324, 383–84
SysMoveObject, 324, 362–363
SysOpenObject, 324, 363–364
SysOS2Ver, 324, 391–392
SysPutEA, 324, 384–385
SysQueryClassList, 324, 364–365
SysRegisterObjectClass, 324, 365–366
SysRmDir, 324, 386–387
SysSaveObject, 324, 366–367
SysSearchPath, 325, 387

SysSetIcon, 325, 387–388
SysSetObjectData, 325, 367–368
SysSleep, 325, 332
system attribute, 146, 148, 181–182
System Clock, 287, 299, 304–310, 396, 750–751
System Clock object, 151–152
System Configuration dialog box, 18–19, 20, 22, 23, 33
System Configuration file, 447–448
System Editor, 47–48, 82, 431
- associations and, 48, 77–79
- basic description of, 420–421

System folder, 54, 137
- the Minimized Window Viewer and, 199
- objects in, 395–426
- the Object Title property and, 179

System Information Tool, 480–482, 522–524
SYSTEM.INI, 296–297, 760
System object, 274–280, 396
SysTempFileName, 325, 389–390
System Setup folder, 31, 33, 35
- confirmation objects and, 99
- Country objects in, 299–304
- Font Palette in, 230–231
- Keyboard object in, 269–273
- Mouse object in, 256
- objects in, list of, 396
- opening, 190
- palette management and, 210
- Plug and Play for PCMCIA objects in, 315–319
- Power objects in, 310–314
- Scheme Palette in, 247
- Sound objects in, 287–290
- Spooler objects in, 291–293
- System Clock objects in, 304–310
- versions of the Color Palette in, 223–224

Systems Pointers area, 263, 266
System Utilities dialog box, 25
SysTextScreenRead, 325, 332–333
SysTextScreenSize, 325, 333–334
SysWaitNamedPipe, 325, 390–391

T

TASKLIST, 449
Technical Support Center, 36
TEDIT, 682
Tele-Communications Inc., 555
TEMPLATE=, 343, 345
templates. *See also* Templates folder
- basic description of, 45–46, 58
- IBM Works, 484
- specify objects as, 137
- in the Workplace Shell, 76–77

Templates folder, 46, 58, 129–130, 197–198
- dragging objects from, 93, 568–571
- Network Printer template in, 572

temporary objects, 149–151
text
- copying, 400, 410
- linking, to documents and charts, 486
- title, editing, 261, 271–272

Text mouse pointer, 261–262
TEXT.PTR, 264
tiling
- bitmaps, 245
- windows, 66

time. *See also* timing
- in Country objects, 299–304
- creation, 145, 181
- displaying current, 643

last changed/last accessed, 145, 181
last write, 74, 180–181
print job, 575–576
TIME command, 643
TIMEFMT=, 558
Time page, 300–301
TIMER.DRV, 728
timing
cursor blink rates, 269–270
how long logo panels stay onscreen, 286
mouse click/tracking speed, 256–257
repeat rates, 269–270, 272–273
Timing page, 256–257
Timout push button, 190
TITLE=, 343, 345
Title page, 279–280
Title property, 122
title text, editing, 261, 271–272
TO=, 558
To-Do List Launcher, 494
toggling, between windows, 405
tools, installation of, 26
Tools/Color fill, 413
Tools menu, 419
TOUCH_EXCLUSIVE_ACCESS, 702–703
touch screen settings, 702–703
Trademarks, 431
transient objects, 198
Transparent bkgnd check box, 237, 239, 240–242
transparent color, 237–242, 250, 253
Transparent Color check box, 250, 253
trash can, 56
TREE, 682
TREEFONT=, 343, 348
Tree view, 62, 70, 75–76, 130, 150–151, 173
TREEVIEW=, 343, 348
Tree view page, 238–240
TUTORIAL, 682
tutorials, 24, 57, 104, 430, 556, 682
TYPE, 643
type page, 169–170

U

UART, 502
Ultimedia Mail/2 'Lite,' 501, 505–506
Ultimotion, 551
UMBs (Upper Memory Blocks), 719
UNC (Universal Naming Convention) filenames, 436–437
UNDELETE, 616, 683
undo actions
restoring mouse settings with, 263
undoing edit window changes, 421
undoing Icon Editor changes, 414
undoing window property changes, 139
Unlock Disk, 153, 155
Unmark, 411
UNPACK, 683–684
updates, displaying, in multiple views, 61
URLs (Uniform Resource Locators), 507
USENET news groups, 509–511
User Configuration file, 447–448
user IDs, 503
user interface functions, 325–334
utilities
diskettes, creating, 35–36, 396
loading REXX, 322–325

V

V86 (Virtual 8086) mode, 692
VDD (Virtual Device Driver), 691, 704–706, 719, 722
VDM (Virtual DOS Machine), 693, 718, 722, 728
VER, 643
VERIFY, 643
versions
  displaying current, 643
  OS/2, basic description of, 2–3, 7
vertical bar (|), 604, 605
VGA displays, 31, 206–207, 223, 473, 474–475
VIDBOUNC.CMD, 561–562
VIDEO_8514A_XGA_IOTRAP, 707
VideoIN, 480–482, 521, 528, 530, 533, 548, 550–556, 565
VIDEO_MODE_RESTRICTION, 705
VIDEO_ONDEMAND_MEMORY, 705
VIDEO_RETRACE_EMULATION, 705
VIDEO_ROM_EMULATION, 705
video settings, 704–706
VIDEO_SWITCH_NOTIFICATION, 705
VIDEO_WINDOW_REFRESH, 706
VIEW, 684–685
View Current Configuration push button, 276
View pages, 183, 304, 305–309, 312–314, 579, 592–593
views. *See also* specific views
  container object, 62
  default, 125
  displaying updates in, 61
  for each type of object, 60
  multiple, 60–61
  of objects, basic description of, 59–62
  of on-line reference books, 684–685
  opening, 127
View Sorting Characteristics property, 172
Visible check box, 237
Visible Text check box, 250, 253
visual paradigms, standard, 63–67
VOL, 159, 644
Volume Label field, 162
Volume Serial Number field, 162

W

Wait mouse pointer, 262, 264
WAIT.PTR, 264
warning messages, 287, 290
WebExplorer, 501, 504, 505–507, 528
WEPM.EXE, 538–540
width
  of borders, 253
  of cells, 164
wildcard characters, 608, 649
Win32s, 26–28
WIN_ATM, 296, 717
WIN_CLIPBAORD, 717
WIN_DDE, 717
Window List, 64–67, 199, 260, 273, 449
  actions performed on, 65–67
  activating, 103, 108
Window pages, 186–187, 280–284
Windows (Microsoft)
  FAT file system and, 143
  formatting diskettes and, 160–161
  installation of, on drive C:, 15

support, overview of, 692–693
templates and, 45
Version 3.1, 7, 26–28
Version 3.11, 7, 26–28
Windows NT (Microsoft), 141, 435
WIN.INI, 406
WIN-OS/2, 26–28, 504, 728, 759–761
command prompts, 406–407
cursor blink rates and, 270
full screen radio button, 293
keyboard settings, 696–698
the Internet and, 504, 506–507
memory settings, 698–703
mouse and touch screen settings, 702–703
printer settings, 703–704
session settings, 86, 694–695
settings, basic description of, 691–730
settings, printing/loading, 720–725
settings categories, 695–718
starting windows as separate sessions with, 727
video settings, 704–706
WIN-OS/2 Setup Object, 153, 293–299, 396
3.1 Session page, 293–297
Data Exchange page, 293, 297–299
WIN_RUN_MODE, 295, 716
WINWORD.CDM, 754–759
Word for Windows, 143, 753–759
word length, 585
word processors, 483–485
word wrap, 421
WORKAREA=, 343, 348
Work Area folders, 55–56, 751–753
Work Area property, 172
Workplace Process, 444–447
Workplace Shell
basic description of, 41–88
characteristics of standard objects, 59–67
CONFIG.SYS options and, 441–464
controls, 53–59
copying/moving objects and, 96–97
customizing, 255–319
folders in, 69–76
functions, 334–368
Getting Started tutorial, 57
help, 57
installation and, 18
model, basic description of, 44–47
objects, referencing, 334–339
object types, 119–202
Program object parameters in, 82–86
setup strings and, 339–347
standard objects in, 53–69
templates in, 76–77
user interface elements, 42–44
Workplace Shell palette aware check box, 210, 211, 243, 275–276
World Wide Web, 506–507, 528. *See also* Internet
WPAbstract, 96, 139–140, 151–168
WPBitmap, 48, 168, 170
WPClock, 151–152
WPColorPalette, 151–152, 353
WPCommandFile, 168, 171
WPCountry, 151–152
WPDataFile, 48, 168–171
WPDesktop, 187–194
WPDisk, 151–152, 153–165, 348
WPDrives, 194
WPDSINIT, 461, 685
WPFileSystem, 139, 141–151, 168
WPFolder, 172–187, 346–348
WPIcon, 48, 168, 170
WPJob, 198

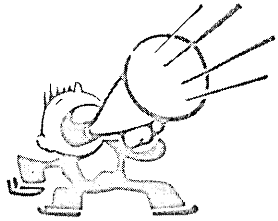
WPKeyboard, 151–152
WPLaunchPad, 151–152, 348–349
WPMet, 168, 171
WPMinWindow, 198–200
WPMinWinViewer, 194–195
WPMouse, 151–152
WPNetGrp, 195, 350
WPNetLink, 152
WPNetwork, 195
WPObject, 121–139, 168, 344–345
WPPalette, 152, 163–165, 352–353
WPPdr, 200
WPPif, 168, 171
WPPointer, 48, 168, 171
WPPort, 200
WPPower, 152
WPPrinter, 152, 355–356
WPPrinterDriver, 171
WPProgram, 152, 350–351
WPProgramFile, 168, 171, 350–351
WPProgramGroup, 195, 352
WPQdr, 200
WPRootFolder, 195–196
WPRPrinter, 152
WPSchemePalette, 152, 354–355
WPServer, 196, 352
WPShadow, 152, 165–167, 352
WPSharedDir, 196
WPShredder, 152, 167
WPSound, 152
WPSpecialNeeds, 152, 167–168
WPSpool, 153
WPStartup, 196–197
WPSystem, 153
WPTemplates, 197–198
WPTransient, 139, 149–151
write protection, 158

X

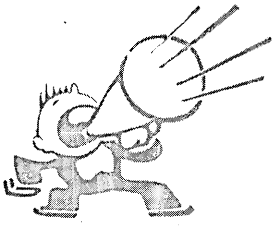
X background, 417
XCELLCOUNT=, 343, 352, 354
XCELLGAP=, 343, 352
XCELLWIDTH=, 343, 352
XCOPY, 685–688
XDFCOPY, 688
XGA/8514 Form, 418
XMS (Extended Memory Specification), 701, 719
XMS_HANDLES, 701
XMS_MEMORY_LIMIT, 701
XMS_MINIMUM_HMA, 701

Y

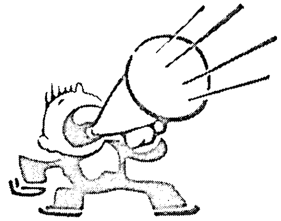
YCELLCOUNT=, 343, 353, 354, 355
YCELLGAP=, 343, 353
YCELLHEIGHT=, 343, 353
Year Calendar, 494–495



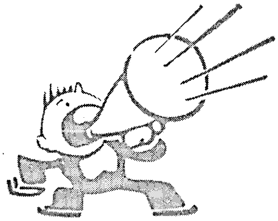
Notes



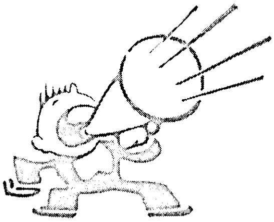
Notes



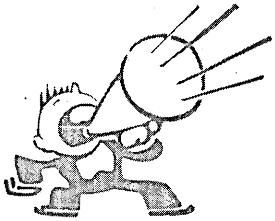
Notes



Notes

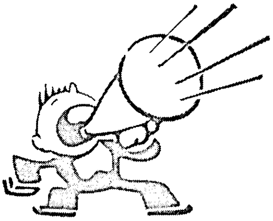


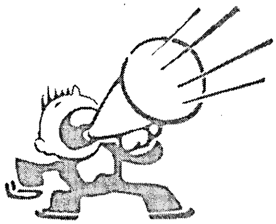
Notes



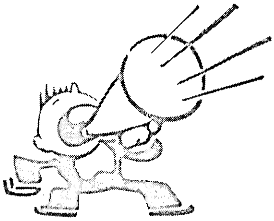
Notes

Notes

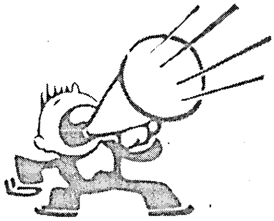




Notes

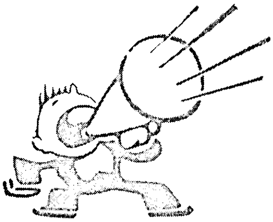


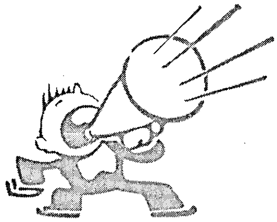
Notes



Notes

Notes





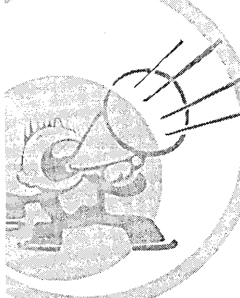
Notes

IDG BOOKS WORLDWIDE LICENSE AGREEMENT

Important — read carefully before opening the software packet. This is a legal agreement between you (either an individual or an entity) and IDG Books Worldwide, Inc. (IDG). By opening the accompanying sealed packet containing the software disc, you acknowledge that you have read and accept the following IDG License Agreement. If you do not agree and do not want to be bound by the terms of this Agreement, promptly return the book and the unopened software packet(s) to the place you obtained them for a full refund.

1. License. This License Agreement (Agreement) permits you to use one copy of the enclosed Software program(s) on a single computer. The Software is in "use" on a computer when it is loaded into temporary memory (i.e., RAM) or installed into permanent memory (e.g., hard disk, CD-ROM, or other storage device) of that computer.

2. Copyright. The entire contents of this disc and the compilation of the Software are copyrighted and protected by both United States copyright laws and international treaty provisions. You may only (a) make one copy of the Software for backup or archival purposes, or (b) transfer the Software to a single hard disk, provided that you keep the original for backup or archival purposes. The individual programs on the disc are copyrighted by the authors of each program respectively. Each program has its own use permissions and limitations. To use each program, you must follow the individual requirements and restrictions detailed for each in Appendix A of this Book. Do not use a program if you do not want to follow its Licensing Agreement. None of the material on this disc or listed in this Book may ever be distributed, in original or modified form, for commercial purposes.



3. Other Restrictions. You may not rent or lease the Software. You may transfer the Software and user documentation on a permanent basis provided you retain no copies and the recipient agrees to the terms of this Agreement. You may not reverse engineer, decompile, or disassemble the Software except to the extent that the foregoing restriction is expressly prohibited by applicable law. If the Software is an update or has been updated, any transfer must include the most recent update and all prior versions. Each shareware program has its own use permissions and limitations. These limitations are contained in the individual license agreements that are on the software discs. The restrictions include a requirement that after using the program for a period of time specified in its text, the user must pay a registration fee or discontinue use. By opening the package that contains the software disc, you will be agreeing to abide by the licenses and restrictions for these programs. Do not open the software package unless you agree to be bound by the license agreements.

4. Limited Warranty. IDG warrants that the Software and disc are free from defects in materials and workmanship for a period of sixty (60) days from the date of purchase of this Book. If IDG receives notification within the warranty period of defects in material or workmanship, IDG will replace the defective disc. IDG's entire liability and your exclusive remedy shall be limited to replacement of the Software, which is returned to IDG with a copy of your receipt. This Limited Warranty is void if failure of the Software has resulted from accident, abuse, or misapplication. Any replacement Software will be warranted for the remainder of the original warranty period or thirty (30) days, whichever is longer.

5. No Other Warranties. To the maximum extent permitted by applicable law, IDG and the author disclaim all other warranties, express or implied, including but not limited to implied warranties of merchantability and fitness for a particular purpose, with respect to the Software, the programs, the source code contained therein and/or the techniques described in this Book. This limited warranty gives you specific legal rights. You may have others which vary from state/jurisdiction to state/jurisdiction.

6. No Liability For Consequential Damages. To the extent permitted by applicable law, in no event shall IDG or the author be liable for any damages whatsoever (including without limitation, damages for loss of business profits, business interruption, loss of business information, or any other pecuniary loss) arising out of the use of or inability to use the Book or the Software, even if IDG has been advised of the possibility of such damages. Because some states/jurisdictions do not allow the exclusion or limitation of liability for consequential or incidental damages, the above limitation may not apply to you.

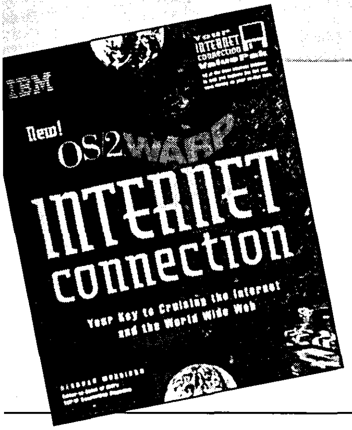
7. U.S.Government Restricted Rights. Use, duplication, or disclosure of the Software by the U.S. Government is subject to restrictions stated in paragraph (c) (1) (ii) of the Rights in Technical Data and Computer Software clause of DFARS 252.227-7013, and in subparagraphs (a) through (d) of the Commercial Computer—Restricted Rights clause at FAR 52.227-19, and in similar clauses in the NASA FAR supplement, when applicable.

Replacement Disc. If a replacement CD-ROM is needed, please write to the following address: IDG Books Disc Fulfillment Center, Attn: *OS/2 Warp Uncensored,* IDG Books Worldwide, 7260 Shadeland Station, Indianapolis, IN 46256, or call 800-762-2974.

IBM PRESS

இப் புத்தகங்கள் மிகவும் நல்லவை

(way cool)



OS/2 Warp Internet Connection: Your Key to Cruising the Internet and the World Wide Web

by Deborah Morrison

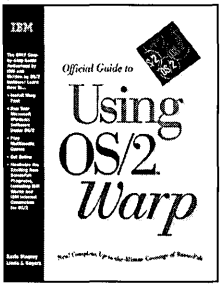
OS/2 users can get warped on the Internet using the OS/2 Warp tips, techniques, and helpful directories found in *OS/2 Warp Internet Connection.* This reference covers OS/2 Warp Internet basics, such as e-mail use and how to access other computers, plus much more! The Internet gets more complex everyday, but for OS/2 Warp users it just got a whole lot easier! Your value-packed disk includes 10 of the best internet utilities to help you explore the Net and save money while you're on-line!

EXPERT AUTHOR PROFILE

Deborah Morrison (Raleigh, NC) is an award-winning IBM writer who specializes in TCP/IP and the Internet. She is currently the editor-in-chief of IBM's *TCP/IP*

ISBN: 1-56884-465-4
$24.99 USA/$34.99 Canada
Includes one 3.5" disk

Available: Now



)fficial Guide to Using)S/2 Warp

y Karla Stagray & Linda S. Rogers

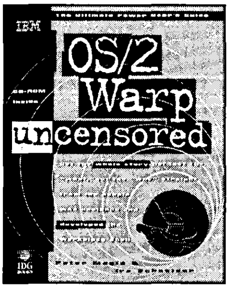
DG Books and IBM have come together o produce the most comprehensive user's ;uide to OS/2 Warp available today. ⁼rom installation to using OS/2 Warp's 3onusPak programs, this book delivers 'aluable help to the reader who needs o get up and running fast. Loaded vith working examples, easy tips, and)perating system concepts, *Official Guide o Using OS/2 Warp* is the only official ıser's guide authorized by IBM.

:XPERT AUTHOR PROFILE

Karla Stagray and Linda Rogers (Boca Raton, ⁼L) both have a unique understanding of omputer software and hardware. As award-vinning IBM writers, Stagray and Rogers ıave received Society of Technical :ommunicators awards for various ndeavors.

;BN: 1-56884-466-2
29.99 USA/$39.99 Canada

Available: Now



OS/2 Warp Uncensored

by Peter G. Magid & Ira H. Schneider

Exploit the power of OS/2 Warp and learn the secrets of object technology for the Workplace Shell. This all new book/ CD-ROM bundle, for power users and intermediate users alike, provides the real inside story—not just the "what," but the "how" and "why" — from the folks who designed and developed the Workplace Shell. Packed with tips and techniques for using IBM's REXX programming language, and the bonus CD includes new bitmaps, icons, mouse pointers, REXX scripts, and an Object Tool!

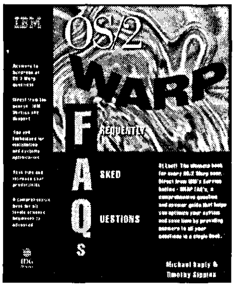
EXPERT AUTHOR PROFILE

Peter G. Magid (Boca Raton, FL) is the User Interface Design Lead for the Workplace Shell and has over 12 years of programming experience at IBM. He is a graduate of Tulane University, and holds a B.S. degree in Computer Science.

Ira H. Schneider (Boca Raton, FL) has focused on enhancements to the Workplace Shell and has over 25 years of experience with IBM. He has held numerous lead programming positions within IBM and graduated from Northeastern University with a B.S. degree in Electrical Engineering.

ISBN: 1-56884-474-3
$39.99 USA/$54.99 Canada
Includes one CD-ROM

Available: Now



OS/2 Warp FAQs™

by Mike Kaply & Timothy F. Sipples

At last, the ultimate answer book for every OS/2 Warp user. Direct from IBM's Service Hotline, *OS/2 Warp FAQs* is a comprehensive question-and-answer guide that helps you optimize your system and save time by putting the answers to all your questions right at your fingertips. CD includes FAQs from the book in an easy-to-search format, plus hard-to-find device drivers for connecting to peripherals, such as printers.

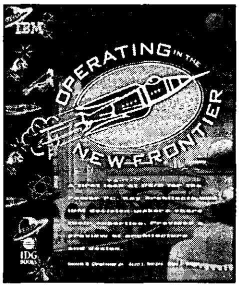
EXPERT AUTHOR PROFILE

Mike Kaply (Boca Raton, FL) is currently on the OS/2 Help Manager Development Team at IBM in Boca Raton, Florida. He holds a B.S. degree in Mathematics and Computer Science from Southern Methodist University.

Timothy F. Sipples (Chicago, IL) is an OS/2 Warp specialist from IBM. He has written for *OS/2 Magazine* and was named "Team OS/2er of the Year" by *OS/2 Professional*.

ISBN: 1-56884-472-7
$29.99 USA/$42.99 Canada
Includes one CD-ROM

Available: September 1995



OS/2 Warp and PowerPC: Operating in the New Frontier

by Ken Christopher, Scott Winters & Mary Pollack Wright

The software makers at IBM unwrap the IBM and OS/2 mystique to share insights and strategies that will take business computing into the 21st century. Readers get a long, hard look at the next-generation of OS/2 Warp for PowerPC.

EXPERT AUTHOR PROFILE

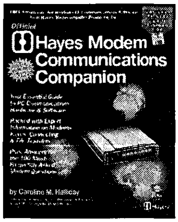
Ken Christopher (Boca Raton, FL) is Program Director of Development for OS/2 for Power PC. He has been a key player in the development on OS/2 Warp.

Scott Winters (Boca Raton, FL) is lead architect of OS/2 for the PowerPC. He has been instrumental in the development on OS/2 Warp on the PowerPC platform.

Mary Pollack Wright (Boca Raton, FL) is currently the technical editor for the OS/2 Techinical Library. She has been part of the OS/2 team since 1985. Her technical articles on OS/2 have been published in the *OS/2 Developer* magazine and *OS/2 Notebooks*.

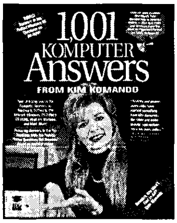
ISBN: 1-56884-458-1
$29.99 USA/$39.99 Canada

Available: September 1995



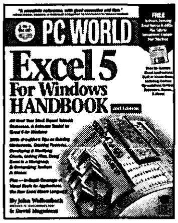
Official Hayes Modem Communications Companion
by Caroline M. Halliday

ISBN: 1-56884-072-1
$29.95 USA/$39.95 Canada
Includes software.



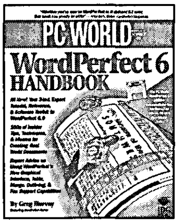
1,001 Komputer Answers from Kim Komando
by Kim Komando

ISBN: 1-56884-460-3
$29.99 USA/$39.99 Canada
Includes software.



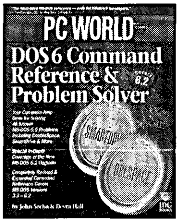
PC World Excel 5 For Windows Handbook, 2nd Edition
by John Walkenbach & Dave Maguiness

ISBN: 1-56884-056-X
$34.95 USA/$44.95 Canada
Includes software



PC World WordPerfect 6 Handbook
by Greg Harvey

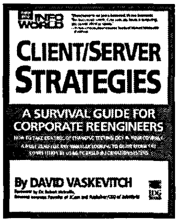
ISBN: 1-878058-80-0
$34.95 USA/$44.95 Canada
Includes software.



PC World DOS 6 Command Reference and Problem Solver
by John Socha & Devra Hall

NATI BESTS

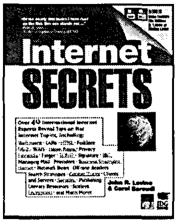
ISBN: 1-56884-055-1
$24.95 USA/$32.95 Canada



Client/Server Strategies: A Survival Guide for Corporate Reengineers
by David Vaskevitch

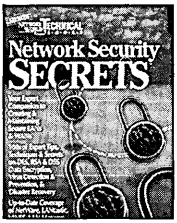
SUPER STAR

ISBN: 1-56884-064-0
$29.95 USA/$39.95 Canada



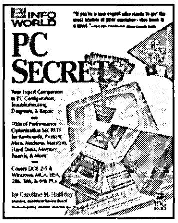
Internet Secrets®
by John Levine & Carol Baroudi

ISBN: 1-56884-452-2
$39.99 USA/$54.99 Canada
Includes Software.



Network Security SECRETS™
by David Stang & Sylvia Moon

ISBN: 1-56884-021-7
Int'l. ISBN: 1-56884-151-5
$49.95 USA/$64.95 Canada
Includes Software.



PC SECRETS™
by Caroline M. Halliday

ISBN: 1-878058-49-5
$39.95 USA/$52.95 Canada
Includes Software.

INFO WORLD TECHNICAL BOOKS

...SECRETS®

IDG BOOKS WORLDWIDE™

Here's a complete listing of PC Press Titles

Title	Author	ISBN #	Price
BBS SECRETS™	by Ray Werner	ISBN: 1-56884-491-3	$39.99 USA/$54.99 Canada
Creating Cool Web Pages with HTML	by Dave Taylor	ISBN: 1-56884-454-9	$19.99 USA/$26.99 Canada
DOS 6 SECRETS™	by Robert D. Ainsbury	ISBN: 1-878058-70-3	$39.95 USA/$52.95 Canada
Excel 5 For Windows Power Programming Techniques	by John Walkenbach	ISBN: 1-56884-303-8	$39.95 USA/$52.95 Canada
Hard Disk SECRETS™	by John M. Goodman, Ph.D.	ISBN: 1-878058-64-9	$39.95 USA/$52.95 Canada
Internet Gizmos™ For Windows	by Joel Diamond, Howard Sobel, & Valda Hilley	ISBN: 1-56884-451-4	$39.99 USA/$54.99 Canada
Making Multimedia Work	by Michael Goodwin	ISBN: 1-56884-468-9	$19.99 USA/$26.99 Canada
MORE Windows 3.1 SECRETS™	by Brian Livingston	ISBN: 1-56884-019-5	$39.95 USA/$52.95 Canada
Official XTree Companion 3rd Edition	by Beth Slick	ISBN: 1-878058-57-6	$19.95 USA/$26.95 Canada
Paradox 4 Power Programming SECRETS,™ 2nd Edition	by Gregory B. Salcedo & Martin W. Rudy	ISBN: 1-878058-54-1	$44.95 USA/$59.95 Canada
Paradox 5 For Windows Power Programming SECRETS™	by Gregory B. Salcedo & Martin W. Rudy	ISBN: 1-56884-085-3	$44.95 USA/$59.95 Canada
PC World DOS 6 Handbook, 2nd Edition	by John Socha, Clint Hicks & Devra Hall	ISBN: 1-878058-79-7	$34.95 USA/$44.95 Canada
PC World Microsoft Access 2 Bible, 2nd Edition	by Cary N. Prague & Michael R. Irwin	ISBN: 1-56884-086-1	$39.95 USA/$52.95 Canada
PC World Word For Windows 6 Handbook	by Brent Heslop & David Angell	ISBN: 1-56884-054-3	$34.95 USA/$44.95 Canada
QuarkXPress For Windows Designer Handbook	by Barbara Assadi & Galen Gruman	ISBN: 1-878058-45-2	$29.95 USA/$39.95 Canada
Windows 3.1 Connectivity SECRETS	by Runnoe Connally, David Rorabaugh & Sheldon Hall	ISBN: 1-56884-030-6	$49.95 USA/$64.95 Canada
Windows 3.1 SECRETS™	by Brian Livingston	ISBN: 1-878058-43-6	$39.95 USA/$52.95 Canada
Windows™ 95 A.S.A.P.	by Dan Gookin	ISBN: 1-56884-483-2	$24.99 USA/$34.99 Canada
Windows™ 95 Bible	by Alan Simpson	ISBN: 1-56884-074-8	$29.99 USA/$39.99 Canada
Windows 95 SECRETS™	by Brian Livingston	ISBN: 1-56884-453-0	$39.99 USA/$54.99 Canada
Windows GIZMOS™	by Brian Livingston & Margie Livingston	ISBN: 1-878058-66-5	$39.95 USA/$52.95 Canada
Windows 3.1 Configuration SECRETS™	by Valda Hilley & James Blakely	ISBN: 1-56884-026-8	$49.95 USA/$64.95 Canada
WordPerfect 6 For Windows Tips & Techniques Revealed	by David A. Holzgang & Roger C. Parker	ISBN: 1-56884-202-3	$39.95 USA/$52.95 Canada
WordPerfect 6 SECRETS™	by Roger C. Parker & David A. Holzgang	ISBN: 1-56884-040-3	$39.95 USA/$52.95 Canada

"A must-have OS/2 Warp book. In just five minutes, I learned enough secrets to dramatically improve my skills."
– Edwin Black, Editor-In-Chief, *OS/2 Professional*

New!

OS/2® Warp uncensored

Straight from two lead developers of OS/2 Warp:

The ultimate unabridged guide to power and productivity in the OS/2 Warp integrated computing environment. Learn to customize the Workplace Shell, streamline your work with the BonusPak applications, and protect your system from crashes with the expert techniques inside *OS/2 Warp Uncensored.*

Only in *OS/2 Warp Uncensored:*

- **Tips for finding and downloading great stuff on the Net with IBM Internet Connection**
- **Little-known CONFIG.SYS tricks to make your system run hassle-free**
- **Essential Archive and Recover secrets to protect yourself from system failures**
- **Timesaving techniques to install OS/2 Warp and configure your hardware for maximum performance**
- **Workplace Shell customization secrets to speed up and simplify your work**
- **Key techniques for writing REXX scripts so you can control your Workplace Shell objects**
- **Optimal settings to use for running DOS and Windows programs under OS/2 Warp**

Uncensored, Uncut! Available Only from IBM Press.

Also available from IBM Press:
Official Guide to Using OS/2 Warp
OS/2 Warp Internet Connection

$39.99 USA
$54.99 Canada
£38.99 Inc. VAT UK

Level
Intermediate to Advanced
Computer Book Shelving Category
PCs/Operating Systems/OS/2 Warp

IBM®

Straight From the Development Team, Only IBM Press Can Provide the Valuable OS/2 Scripts and Tools Enclosed

OS/2 Warp uncensored
Productivity Pack on CD

REXX Scripts that allow you to rebuild your Workplace Shell Objects, plus reusable REXX samples from the book, so there's no retyping!

Bit maps to create cool desktop backgrounds — includes full-screen 256-color bit Maps!

Mouse Pointer Sets to replace the standard arrows

Object Tool — a Workplace Shell object that allows users to view and change all properties of another Workplace Shell object

And Many More Exciting, Productivity-Boosting Resources

Let these Icons Guide You!

Uncensored Key program features that only the developers know

Developer's Desktop Explains how the authors have their own systems configured to maximize OS/2 Warp

CD-ROM Points out key programs and files you'll find on the **Uncensored Productivity Pack CD-ROM!**

About the Authors

Peter G. Magid is the User Interface Design Lead for the OS/2 Workplace Shell and an Advisory Programmer on the Workplace Shell Development Team.

Ira H. Schneider is an Advisory Programmer on the Workplace Shell Development Team. A 26-year IBM veteran, he has been instrumental in developing OS/2 Warp.

IDG
An IBM Press Book
Published by IDG Books Worldwide, Inc.